

Framework

AWS Well-Architected Framework



AWS Well-Architected Framework: Framework

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Die Handelsmarken und Handelsaufmachung von Amazon dürfen nicht in einer Weise in Verbindung mit nicht von Amazon stammenden Produkten oder Services verwendet werden, durch die Kunden irregeführt werden könnten oder Amazon in schlechtem Licht dargestellt oder diskreditiert werden könnte. Alle anderen Marken, die nicht im Besitz von Amazon sind, gehören den jeweiligen Besitzern, die möglicherweise mit Amazon verbunden sind oder von Amazon gesponsert werden.

Table of Contents

Zusammenfassung und Einführung	1
Einführung	1
Definitionen	2
Architekturüberlegungen	5
Allgemeine Designprinzipien	7
Die Säulen des Frameworks	9
Operative Exzellenz	9
Designprinzipien	10
Definition	11
Best Practices	12
Ressourcen	23
Sicherheit	23
Designprinzipien	24
Definition	25
Best Practices	25
Ressourcen	36
Zuverlässigkeit	37
Designprinzipien	37
Definition	38
Bewährte Methoden	39
Ressourcen	45
Leistungseffizienz	45
Designprinzipien	45
Definition	46
Bewährte Methoden	47
Ressourcen	52
Kostenoptimierung	53
Designprinzipien	54
Definition	54
Best Practices	55
Ressourcen	62
Nachhaltigkeit	63
Designprinzipien	63
Definition	64

Bewährte Methoden	65
Ressourcen	72
Der Überprüfungsprozess	74
Schlussfolgerung	77
Mitwirkende	78
Weitere Informationen	79
Dokumentversionen	80
Anhang: Fragen und bewährte Methoden	84
Operative Exzellenz	84
Organisation	84
Vorbereitung	147
Betrieb	223
Weiterentwicklung	270
Sicherheit	291
Sicherheitsgrundlagen	291
Identity and Access Management	318
Erkennung	384
Schutz der Infrastruktur	401
Datenschutz	430
Vorfallreaktion	469
Anwendungssicherheit	494
Zuverlässigkeit	520
Grundlagen	521
Workload-Architektur	565
Änderungsmanagement	620
Fehlerverwaltung	672
Leistungseffizienz	786
Auswahl der Architektur	787
Computer und Hardware	803
Datenverwaltung	821
Netzwerk und Bereitstellung von Inhalten	849
Prozess und Kultur	881
Kostenoptimierung	899
Praxis für Cloud-Finanzmanagement	900
Ausgabenerkennung und Nutzungsbewusstsein	926
Kostengünstige Ressourcen	973

Verwaltung von Nachfrage und Bereitstellung von Ressourcen	1019
Optimierung im Laufe der Zeit	1033
Nachhaltigkeit	1042
Auswahl der Region	1043
Ausrichtung am Bedarf	1045
Software und Architektur	1061
Daten	1074
Hardware und Services	1095
Prozess und Kultur	1105
Hinweise	1118
AWS Glossar	1119

AWS Well-Architected Framework

Veröffentlichungsdatum: 6. November 2024 ([Dokumentversionen](#))

Das AWS Well-Architected-Framework unterstützt Sie dabei, die Vor- und Nachteile der Entscheidungen nachzuvollziehen, die Sie beim Aufbau von Systemen in AWS treffen. Das Framework hilft Ihnen, bewährte Architekturmethoden für die Entwicklung und den Betrieb zuverlässiger, sicherer, effizienter, kostengünstiger und nachhaltiger Systeme in der Cloud zu ermitteln.

Einführung

Das AWS Well-Architected-Framework unterstützt Sie dabei, die Vor- und Nachteile der Entscheidungen nachzuvollziehen, die Sie beim Aufbau von Systemen in AWS treffen. Das Framework hilft Ihnen, bewährte Architekturmethoden für den Entwurf und Betrieb zuverlässiger, sicherer, effizienter, kostengünstiger und nachhaltiger Workloads in der AWS Cloud zu ermitteln. Es bietet Ihnen die Möglichkeit, Ihre Architekturen konsequent an bewährten Methoden zu messen und Verbesserungspotenzial zu identifizieren. Der Prozess zur Überprüfung einer Architektur ist ein konstruktives Gespräch über Architekturentscheidungen und kein Auditmechanismus. Wir sind davon überzeugt, dass eine durchdachte Systemarchitektur maßgeblich zu Ihrem künftigen geschäftlichen Erfolg beiträgt.

AWS Solutions Architects entwerfen seit vielen Jahren Lösungen für unterschiedlichste Branchen und Anwendungsfälle. Wir waren am Design und an der Überprüfung Tausender Kundenarchitekturen in AWS beteiligt. Daher kennen wir die bewährten Methoden und Kernstrategien für erfolgreiche Systemarchitekturen in der Cloud.

Das AWS Well-Architected-Framework dokumentiert grundlegende Fragen, die Ihnen dabei helfen zu klären, ob eine Architektur einwandfrei mit bewährten Methoden für die Cloud vereinbar ist. Über das Framework erhalten Sie eine einheitliche Herangehensweise zur Bewertung der Eigenschaften, die Sie von modernen Cloud-basierten Systemen erwarten, sowie Vorschläge zur Realisierung dieser Eigenschaften. AWS entwickelt sich ständig weiter, und auch wir lernen durch die Arbeit mit unseren Kunden ständig dazu. Mit diesem wachsenden Wissen können wir immer noch genauer definieren, wodurch sich eine gute architektonische Struktur auszeichnet.

Dieses Framework richtet sich an Technologiefachleute, z. B. Chief Technology Officers (CTO), Architekten, Entwickler und Mitglieder des Betriebsteams. Die darin enthaltenen bewährten Methoden und Strategien für AWS kommen bei der Gestaltung und Nutzung von Cloud-Workloads zum

Einsatz. Die Links verweisen auf weitere Implementierungsdetails und Architekturmodelle. Weitere Informationen finden Sie auf der [Startseite des AWS Well-Architected Framework](#).

AWS bietet auch einen Service zur kostenfreien Prüfung Ihrer Workloads an. Das [AWS Well-Architected Tool](#) (AWS WA Tool) ist ein Service in der Cloud, der einen einheitlichen Prozess zum Überprüfen und Messen Ihrer Architektur mit dem AWS Well-Architected Framework bietet. Das AWS-WA-Tool stellt Empfehlungen dazu bereit, wie Sie Ihre Workloads zuverlässiger, sicherer, effizienter und kostengünstiger gestalten können.

Um Sie bei der Anwendung von bewährten Methoden zu unterstützen, haben wir [AWS Well-Architected Labs](#) entwickelt. Diese stellen Ihnen ein Repository mit Code und Dokumentation zur Verfügung, damit Sie praktische Erfahrungen mit der Implementierung von bewährten Methoden sammeln können. Außerdem haben wir uns mit ausgewählten Partnern aus dem AWS Partner Network (APN) zusammengetan, die Mitglieder des [Partnerprogramms von AWS Well-Architected](#). Diese AWS-Partner sind bestens mit AWS vertraut und können Sie beim Überprüfen und Verbessern Ihrer Workloads unterstützen.

Definitionen

Die Experten von AWS unterstützen Kunden tagtäglich beim Entwerfen von Systemarchitekturen, die ihnen eine optimale Nutzung bewährter Methoden in der Cloud ermöglichen. Während wir zusammen mit Ihnen die Architektur entwerfen, wägen wir die Anforderungen ab und treffen die richtigen Kompromisse. Wenn Sie die Systeme dann in Live-Umgebungen bereitstellen, beobachten wir, wie gut diese Systeme laufen und welche Auswirkungen die Kompromisse haben.

Unsere bisherigen Erkenntnisse sind die Grundlage des AWS Well-Architected Framework. Das Framework enthält einheitlich zusammengestellte bewährte Methoden, mit denen Kunden und Partner Architekturen bewerten. Anhand verschiedener Fragen können sie beurteilen, wie gut eine Architektur auf die bewährten Methoden von AWS ausgerichtet ist.

Das AWS Well-Architected-Framework basiert auf sechs Säulen: operative Exzellenz, Sicherheit, Zuverlässigkeit, Leistungseffizienz, Kostenoptimierung und Nachhaltigkeit.

Tabelle 1. Die Säulen des AWS Well-Architected Framework

Name	Beschreibung
Operative Exzellenz	Die Fähigkeit, die Entwicklung zu unterstützen und Workloads effektiv auszuführen,

Name	Beschreibung
	Einblicke in die Betriebsabläufe zu erhalten und für geschäftlichen Mehrwert unterstützende Prozesse und Verfahren fortlaufend zu verbessern.
Sicherheit	Die Sicherheitssäule beschreibt, wie Sie Cloud-Technologien nutzen können, um Daten, Systeme und Komponenten so zu schützen, dass Ihre Sicherheitslage verbessert werden kann.
Zuverlässigkeit	Die Säule „Zuverlässigkeit“ umfasst die Fähigkeit einer Workload, die beabsichtigte Funktion erwartungsgemäß korrekt und konsistent auszuführen. Dies umfasst die Möglichkeit, die Workload während des gesamten Lebenszyklus zu betreiben und zu testen. Dieses Dokument bietet umfassende Informationen mit bewährten Methoden für die Implementierung zuverlässiger Workloads in AWS.
Leistungseffizienz	Die Fähigkeit, Computerressourcen effizient zu nutzen, um Systemanforderungen zu erfüllen und diese Effizienz bei der Entwicklung von Nachfrageänderungen und Technologien aufrechtzuerhalten.
Kostenoptimierung	Die Fähigkeit, Systeme zu betreiben, um Unternehmenswert zum niedrigsten Preis zu liefern.

Name	Beschreibung
Nachhaltigkeit	Die Fähigkeit, die Auswirkungen auf die Nachhaltigkeit kontinuierlich zu verbessern, indem der Energieverbrauch reduziert und die Effizienz aller Komponenten einer Workload erhöht wird, indem der Nutzen der bereitgestellten Ressourcen maximiert und die insgesamt erforderlichen Ressourcen minimiert werden.

In Zusammenhang mit dem AWS Well-Architected-Framework verwenden wir diese Bezeichnungen:

- Eine Komponente besteht aus dem Code, der Konfiguration und den AWS-Ressourcen, die für eine Anforderung bereitgestellt werden. Eine Komponente ist häufig die Einheit technischen Eigentums und von anderen Komponenten losgelöst.
- Der Begriff Workload wird für zusammengehörige Komponenten, die geschäftlichen Mehrwert darstellen, verwendet. Eine Workload ist in vielen Fällen der Detaillierungsgrad, von dem Führungskräfte aus Wirtschaft und Technik häufig sprechen.
- Wir betrachten Architektur als das Zusammenwirken von Komponenten in einer Workload. Wie Komponenten kommunizieren und interagieren, ist häufig der Schwerpunkt von Architektordiagrammen.
- Meilensteine markieren wichtige Änderungen im Laufe der Entwicklung einer Architektur im Produktlebenszyklus (Entwurf, Implementierung, Tests, Inbetriebnahme und Produktionsbetrieb).
- In Zusammenhang mit einer Organisation ist das Technologieportfolio die Sammlung an Workloads, die für den Geschäftsbetrieb erforderlich sind.
- Der Grad des Aufwands bezeichnet die Zeitspanne, den Aufwand und die Komplexität, die für die Implementierung einer Aufgabe benötigt werden. Jede Organisation muss die Größe und das Fachwissen des Teams sowie die Komplexität der Workload berücksichtigen, um den Grad des Aufwands für die Organisation richtig einzuordnen.
 - Hoch: Die Arbeit dauert möglicherweise mehrere Wochen oder Monate. Sie könnte in mehrere Abschnitte, Veröffentlichungen und Aufgaben aufgeteilt werden.
 - Mittel: Die Arbeit dauert möglicherweise mehrere Tage oder Wochen. Sie könnte in mehrere Veröffentlichungen und Aufgaben aufgeteilt werden.

- **Niedrig:** Die Arbeit dauert möglicherweise mehrere Stunden oder Tage. Sie könnte in mehrere Aufgaben aufgeteilt werden.

Beim Entwerfen von Workloads stellen Sie eine Kosten-Nutzen-Abwägung zwischen Säulen abhängig von Ihrem Geschäftskontext an. Diese Geschäftsentscheidungen können Ihre technischen Prioritäten beeinflussen. In Entwicklungsumgebungen könnten Sie im Hinblick auf eine Verbesserung der Nachhaltigkeitswirkung und eine Verringerung der Kosten zulasten der Zuverlässigkeit optimieren. Bei unternehmenskritischen Lösungen könnten Sie dagegen die Zuverlässigkeit optimieren und dafür höhere Kosten und stärkere Auswirkungen auf die Nachhaltigkeit in Kauf nehmen. Bei E-Commerce-Lösungen kann sich die Leistung auf die Einnahmen und die Kauflust der Kunden auswirken. Sicherheit und operative Exzellenz werden in der Regel nicht gegen die anderen Säulen abgewogen.

Architekturüberlegungen

In On-Premises-Umgebungen setzen Kunden oft ein zentrales Technologiearchitektur-Team ein. Dies dient als Überlagerung für Produkt- oder Feature-Teams, um zu verifizieren, dass diese nach bewährten Methoden arbeiten. Technologiearchitektur-Teams setzen sich üblicherweise aus Fachleuten mit unterschiedlichen Aufgabengebieten zusammen, z. B.: Technical Architect (Infrastruktur), Solutions Architect (Software), Data Architect, Networking Architect und Security Architect. Diese Teams arbeiten oft nach dem [TOGAF](#)-Modell oder dem [Zachman-Framework](#) – als Teil eines Kompetenzbereichs für Enterprise-Architektur.

Bei AWS werden die Fähigkeiten lieber auf einzelne Teams verteilt, als sie in einem Zentralteam zu konzentrieren. Wenn die Entscheidungsbefugnis auf mehrere Teams verteilt wird, geht das mit Risiken einher. So muss beispielsweise bestätigt sein, dass die Teams internen Standards gerecht werden. Um diese Risiken aufzufangen, verwenden wir zwei Methoden. Zum einen arbeiten wir mit Methoden (Vorgehensweisen, Prozessen, Standards und gemeinhin anerkannte Normen), die darauf abzielen, jedes Team mit dieser Fähigkeit auszustatten. Dazu setzen wir Experten ein, die dafür sorgen, dass die Teams die vorgegebenen Standards übertreffen. Zweitens implementieren wir Mechanismen, die automatisch kontrollieren, ob Standards eingehalten werden.

 „Gut gemeinte Absichten funktionieren nicht. Wer etwas erreichen will, braucht gute Mechanismen“ – Jeff Bezos.

Das bedeutet konkret, dass wir das Bestmögliche, das Menschen leisten können, durch (oftmals automatisierte) Mechanismen ersetzen, die kontrollieren, ob Regeln oder Prozesse eingehalten werden. Hinter diesem breit aufgestellten Ansatz stehen die [Führungsprinzipien von Amazon](#). Diese stellen sicher, dass in allen Aufgabenbereichen eine Kultur verankert wird, die vom Kunden aus denkt. Vom Kunden aus zu denken, ist ein grundlegender Bestandteil unseres Innovationsprozesses. Unsere Arbeit richtet sich ganz nach dem Kunden und dessen Wünschen. Kundenfixierte Teams richten die Produktentwicklung auf Kundenwünsche aus.

In Zusammenhang mit Architekturen bedeutet das: Wir erwarten von jedem Team, dass es Architekturen erstellen und nach bewährten Methoden arbeiten kann. Um neuen Teams zu diesen Fähigkeiten zu verhelfen bzw. um bestehende Teams leistungsfähiger zu machen, nehmen wir sie in eine virtuelle Community auf, in der Principal Engineers ihre Entwürfe begutachten und sie an die bewährten Methoden von AWS heranführen. Die Community der Principal Engineers hat die Aufgabe, bewährte Methoden sichtbar und verständlich zu machen. Dies geschieht beispielsweise durch Mittagsvorträge, in denen es um die Anwendung bewährter Methoden an praktischen Beispielen geht. Die Vorträge werden aufgezeichnet und können für das Onboarding neuer Teammitglieder eingesetzt werden.

Wir haben bislang mehrere Tausende internetähnliche Systeme eingerichtet und dabei einen Erfahrungsschatz aufgebaut, aus dem sich die bewährten Methoden von AWS herauskristallisiert haben. Wir bevorzugen, bewährte Methoden mit Hilfe von Daten zu definieren. Wir setzen dafür aber auch Fachexperten (z. B. Principal Engineers) ein. Principal Engineers sind direkt dabei, wenn sich neue bewährte Methoden abzeichnen. Als Community können sie bestätigen, dass die Teams danach arbeiten. Im Laufe der Zeit werden diese bewährten Methoden in unsere internen Prüfprozesse sowie in Compliance-Mechanismen aufgenommen. Das Well-Architected Framework ist die kundenseitige Implementierung unseres internen Prüfprozesses. Darin ist die Denkweise der Principal Engineers für Zuständigkeitsbereiche vor Ort (z. B. Solutions Architecture, interne Engineering-Teams) festgeschrieben. Das Well-Architected Framework ist ein skalierbarer Mechanismus, mit dem Sie von diesen Erkenntnissen profitieren können.

Wenn so vorgegangen wird wie in einer Community aus Principal Engineers (mit verteilten Architekturzuständigkeiten), kann unserer Ansicht nach eine Well-Architected Enterprise-Architektur zustande kommen, die auf die Kundenwünsche ausgerichtet ist. Technologievordenker (z. B. CTO oder Entwicklungsleiter), die all Ihre Workloads nach den Prinzipien des Well-Architected-Ansatzes prüfen, können die Risiken Ihres Technologieportfolios aufzeigen. Sie identifizieren teamübergreifende Themen, die Ihre Organisation mit Hilfe von Mechanismen, Training oder Mittagsvorträgen angehen könnte. Allesamt Gelegenheiten für Ihre Principal Engineers, ihr Wissen zu bestimmten Themen an mehrere Teams weiterzugeben.

Allgemeine Designprinzipien

Das Well-Architected Framework fasst allgemeine konzeptionelle Grundsätze zusammen, die gutes Design in der Cloud fördern:

- **Keine Ungewissheit mehr über die Kapazität:** Wenn Sie bei der Bereitstellung einer Workload eine schlechte Entscheidung zur Kapazität treffen, sitzen Sie anschließend möglicherweise auf nicht genutzten Ressourcen oder haben zu wenig Kapazität und müssen sich mit mangelnder Performance herumschlagen. Beim Cloud Computing gibt es diese Probleme nicht. Sie arbeiten mit so viel oder so wenig Kapazität wie nötig. Das System wird automatisch auf- und abskaliert.
- **Systeme auf Produktionsbetrieb testen:** Sie können in der Cloud bei Bedarf eine Testumgebung in Produktionsgröße einrichten, Ihre Tests abschließen und die Ressourcen dann wieder außer Betrieb nehmen. Weil Sie für die Testumgebung nur dann zahlen, wenn sie genutzt wird, können Sie Ihre Live-Umgebung zu einem Bruchteil der Kosten testen, die Sie an einem On-Premises-Standort hätten.
- **Automatisieren unter Berücksichtigung architektonischer Experimente:** Wenn Sie automatisieren, können Sie Ihre Workloads kostengünstig erstellen und replizieren und vermeiden manuellen Aufwand. Sie können an der Automatisierung vorgenommene Änderungen nachverfolgen, die Auswirkungen nachprüfen und ggf. auf die vorherigen Parameter zurücksetzen.
- **Evolutionäre Architekturen berücksichtigen:** In herkömmlichen Umgebungen sind architekturelevante Entscheidungen oft als statische, einmalig auftretende Ereignisse implementiert. Dementsprechend gibt es während der Lebensdauer des Systems einige wenige Hauptversionen. Geschäftsvoraussetzungen und ihr Kontext entwickeln sich stetig weiter. Diese anfangs getroffenen Entscheidungen könnten die Fähigkeit des Systems beeinträchtigen, sich auf neue Geschäftsvoraussetzungen einzustellen. In der Cloud können Sie jederzeit automatisieren und testen. Dadurch wird weniger wahrscheinlich, dass sich Änderungen am Design negativ auswirken. Systeme können sich somit im Laufe der Zeit weiterentwickeln. Unternehmen können dann wie selbstverständlich Innovationen für sich nutzen.
- **Mit Daten Architekturen weiterentwickeln:** Sie können in der Cloud Daten zu der Frage sammeln, wie Ihre architekturelevanten Entscheidungen auf das Verhalten Ihrer Workload durchschlagen. Sie können also mit faktenbasierten Entscheidungen Ihre Workload verbessern. Ihre Cloud-Infrastruktur ist Code. Das bedeutet, dass Sie diese Daten im Laufe der Zeit in architekturelevante Entscheidungen und Verbesserungsmaßnahmen einfließen lassen können.
- **Verbesserung anhand von Gamedays:** Simulieren Sie an regelmäßigen Gamedays Vorfälle in der Produktion, um das Verhalten Ihrer Architektur und Prozesse zu simulieren. So können Sie

nachvollziehen, wo nachgebessert werden kann. Zudem üben Sie dabei ein, wie Ihre Organisation mit Ereignissen umgeht.

Die Säulen des Frameworks

Wenn Sie ein Softwaresystem bauen, gehen Sie ähnlich vor wie beim Hausbau. Wenn das Fundament nicht trägt, können Risse auftreten und das Gebäude unbrauchbar machen. Wenn Sie die Architektur einer Technologielösung planen und die sechs Säulen „Operative Exzellenz“, „Sicherheit“, „Zuverlässigkeit“, „Leistungseffizienz“, „Kostenoptimierung“ und „Nachhaltigkeit“ vernachlässigen, kann es schwer werden, ein System zu schaffen, das Ihre Erwartungen und Anforderungen erfüllt. Berücksichtigen Sie aber diese Säulen in Ihrer Architektur, steht am Ende ein stabiles, effizientes System. Und das gibt Ihnen Freiraum, um sich auf andere Designaspekte (z. B. funktionale Anforderungen) zu konzentrieren.

Säulen

- [Operative Exzellenz](#)
- [Sicherheit](#)
- [Zuverlässigkeit](#)
- [Leistungseffizienz](#)
- [Kostenoptimierung](#)
- [Nachhaltigkeit](#)

Operative Exzellenz

Operative Exzellenz (OE) beinhaltet die Verpflichtung, Software korrekt zu entwickeln und dabei durchgehend ein hervorragendes Kundenerlebnis zu bieten. Dies umfasst bewährte Methoden für die Organisation Ihres Teams, die Gestaltung Ihres Workloads, den Betrieb in großem Maßstab und die Weiterentwicklung im Laufe der Zeit.

Die Säule „Betriebliche Exzellenz“ gibt einen Überblick über konzeptionelle Grundsätze, bewährte Methoden und Fragen. Verbindliche Anleitungen zur Implementierung finden Sie im [Whitepaper „Säule der betrieblichen Exzellenz“](#).

Themen

- [Designprinzipien](#)
- [Definition](#)
- [Best Practices](#)

- [Ressourcen](#)

Designprinzipien

Nachfolgend finden Sie die konzeptionellen Grundsätze für die betriebliche Exzellenz in der Cloud:

- **Organisieren von Teams nach Geschäftsergebnissen:** Die Fähigkeit eines Teams, Geschäftsergebnisse zu erzielen, hängt von der Vision der Führung, effektiven Abläufen und einem geschäftsorientierten Betriebsmodell ab. Die Führungskräfte sollten sich voll und ganz für eine CloudOps-Transformation mit einem geeigneten Cloud-Betriebsmodell einsetzen, das die Teams dazu anregt, möglichst effizient zu arbeiten und Geschäftsergebnisse zu erzielen. Ein geeignetes Betriebsmodell nutzt Personal-, Prozess- und Technologiekapazitäten, um zu skalieren, die Produktivität zu optimieren und durch Agilität, Reaktionsfähigkeit und Anpassung einen Wettbewerbsvorteil zu erlangen. Die langfristige Vision der Organisation wird in Ziele umgesetzt, die Stakeholdern und Verbrauchern Ihrer Cloud-Services unternehmensweit vermittelt werden. Ziele und operative KPIs sind auf allen Ebenen aufeinander abgestimmt. Diese Vorgehensweise sorgt dafür, dass der langfristige Mehrwert, der sich aus der Umsetzung der folgenden Gestaltungsprinzipien ergibt, dauerhaft gewährleistet ist.
- **Implementieren von Beobachtbarkeit für umsetzbare Erkenntnisse:** Verschaffen Sie sich einen umfassenden Überblick über das Verhalten, die Leistung, die Zuverlässigkeit, die Kosten und den Zustand von Workloads. Legen Sie wichtige Key Performance Indicators (KPIs, Leistungskennzahlen) fest und nutzen Sie die Telemetrie der Beobachtbarkeit, um fundierte Entscheidungen zu treffen und sofort einzugreifen, wenn die Geschäftsergebnisse gefährdet sind. Verbessern Sie proaktiv Leistung, Zuverlässigkeit und Kosten auf der Grundlage von verwertbaren Daten zur Beobachtbarkeit.
- **Sichere Automatisierung wo möglich:** In der Cloud können Sie die gleichen technischen Vorgehensweisen wie beim Anwendungscode in Ihrer gesamten Umgebung anwenden. Sie können Ihre gesamte Workload und deren Betrieb (Anwendungen, Infrastruktur, Konfiguration und Verfahren) als Code definieren und aktualisieren. Anschließend können Sie den Betrieb Ihrer Workloads automatisieren, indem Sie sie als Reaktion auf Ereignisse initiieren. In der Cloud können Sie Automatisierungssicherheit einsetzen, indem Sie einen Integritätsschutz wie Ratenkontrolle, Fehlerschwellenwerte und Genehmigungen einrichten. Durch eine effektive Automatisierung können Sie konsistente Reaktionen auf Ereignisse durchsetzen, menschliche Fehler begrenzen und den Arbeitsaufwand der Mitarbeiter reduzieren.
- **Vornehmen kleiner, häufiger und umkehrbarer Änderungen:** Entwerfen Sie skalierbare und lose verkoppelte Workloads, sodass Komponenten regelmäßig aktualisiert werden können.

Automatisierte Bereitstellungstechniken in Verbindung mit kleineren, inkrementellen Änderungen verringern den „Blast Radius“ und ermöglichen eine schnellere Umkehrung bei Fehlern. Dadurch erhöht sich das Vertrauen, vorteilhafte Änderungen an Ihrer Workload vornehmen zu können, während die Qualität erhalten bleibt und Sie sich schnell an veränderte Marktbedingungen anpassen können.

- Betriebliche Verfahren regelmäßig nachbessern: Wenn Sie Ihre Workloads weiterentwickeln, passen Sie auch Ihre Betriebsabläufe entsprechend an. Suchen Sie beim Einsatz betrieblicher Verfahren nach Möglichkeiten, diese zu verbessern. Führen Sie regelmäßige Überprüfungen durch und vergewissern Sie sich, dass alle Verfahren effektiv sind und dass die Teams mit ihnen vertraut sind. Wenn Lücken festgestellt werden, aktualisieren Sie die Verfahren entsprechend. Informieren Sie alle Beteiligten und Teams über Aktualisierungen der Verfahren. Gamifizieren Sie Ihren Betrieb zum Weitergeben von bewährten Methoden und zur Schulung von Teams.
- Fehlern vorbeugen: Maximieren Sie den betrieblichen Erfolg, indem Sie Ausfallszenarien entwickeln, um das Risikoprofil der Workload und ihre Auswirkungen auf Ihre Geschäftsergebnisse zu verstehen. Testen Sie die Wirksamkeit Ihrer Verfahren und die Reaktion Ihres Teams auf diese simulierten Fehler. Treffen Sie fundierte Entscheidungen, um offene Risiken zu auszuräumen, die anhand Ihrer Tests identifiziert wurden.
- Aus allen betrieblichen Ereignissen und Metriken lernen: Ziehen Sie aus allen betrieblichen Zwischenfällen und Ausfällen entsprechende Lehren und treiben Sie geeignete Verbesserungen voran. Geben Sie Ihre Erkenntnisse an alle Teams in Ihrer gesamten Organisation weiter. Die Erkenntnisse sollten Daten und Anekdoten enthalten, wie die Betriebsabläufe zu den Geschäftsergebnissen beitragen.
- Verwaltete Services verwenden: Verringern Sie die operative Belastung, indem Sie verwaltete AWS-Services nutzen, wo immer dies möglich ist. Erstellen Sie operative Verfahren für die Interaktion mit diesen Services.

Definition

Die bewährten Methoden für betriebliche Exzellenz in der Cloud lassen sich in vier Bereiche einteilen:

- Organisation
- Vorbereitung
- Betrieb
- Weiterentwicklung

Die Leitung Ihrer Organisation definiert Geschäftsziele. Anforderungen und Prioritäten müssen in Ihrer Organisation bekannt sein, damit Aufgaben entsprechend organisiert und durchgeführt und die Geschäftsergebnisse erreicht werden können. Ihre Workload muss die Informationen ausgeben, die für ihre Unterstützung erforderlich sind. Die Implementierung von Services zur Integration, Bereitstellung und Lieferung Ihrer Workload schafft einen erhöhten Fluss nützlicher Änderungen in die Produktion, indem wiederkehrende Prozesse automatisiert werden.

Es kann Risiken im Zusammenhang mit dem Betrieb Ihrer Workload geben. Verstehen Sie diese Risiken und treffen Sie eine fundierte Entscheidung dazu, ob der Übergang in die Produktion vollzogen werden sollte. Ihre Teams müssen in der Lage sein, Ihre Workload zu unterstützen. Geschäfts- und Betriebsmetriken, die von den gewünschten Geschäftsergebnissen abgeleitet werden, erlauben es Ihnen, den Zustand Ihrer Workload und Ihrer Betriebsaktivitäten nachzuvollziehen und auf Vorfälle zu reagieren. Ihre Prioritäten ändern sich, wenn sich Ihre geschäftlichen Anforderungen und die geschäftliche Umgebung ändern. Verwenden Sie diese als Feedback-Schleife, um Ihre Organisation und den Betrieb Ihrer Workload kontinuierlich zu verbessern.

Best Practices

Note

Alle Fragen zur betrieblichen Exzellenz haben das OPS-Präfix als Abkürzung für die Säule.

Themen

- [Organisation](#)
- [Vorbereitung](#)
- [Betrieb](#)
- [Weiterentwicklung](#)

Organisation

Um die Prioritäten festlegen zu können, die den geschäftlichen Erfolg ermöglichen, müssen Ihre Teams gemeinsam in Erfahrung bringen, wie sämtliche Workloads aussehen, welche Rolle die einzelnen Teams dabei spielen und was für geschäftliche Ziele damit erreicht werden sollen. Mit gut definierten Prioritäten erzielen Ihre Bemühungen den größtmöglichen Nutzen. Bewerten Sie die Bedürfnisse interner und externer Kunden. Binden Sie dabei alle wichtigen Beteiligten

ein, einschließlich der Geschäfts-, Entwicklungs- und Betriebsteams, um zu bestimmen, auf welche Bereiche die Anstrengungen konzentriert werden sollten. Durch das Bewerten von Kundenbedürfnissen wird sichergestellt, dass Sie den Support, der für die Erzielung der gewünschten geschäftlichen Ergebnisse erforderlich ist, genau kennen und verstehen. Vergewissern Sie sich, dass Sie sich der Richtlinien oder Verpflichtungen bewusst sind, die von der Führung Ihres Unternehmens definiert wurden. Bewerten Sie externe Faktoren, z. B. gesetzliche Compliance-Anforderungen und Branchenstandards, die einen bestimmten Fokus erfordern oder verstärken können. Überprüfen Sie, ob Sie Mechanismen haben, um Änderungen an internen Governance- und externen Compliance-Anforderungen zu identifizieren. Wenn keine Anforderungen festgestellt werden, stellen Sie sicher, dass diese Prüfung sorgfältig durchgeführt wurde. Überprüfen Sie Ihre Prioritäten regelmäßig, damit sie bei Bedarf aktualisiert werden können.

Bewerten Sie Bedrohungen für das Unternehmen (z. B. Geschäftsrisiken und -verpflichtungen und Bedrohungen der Informationssicherheit) und pflegen Sie diese Informationen in einem Risikoregister. Bewerten Sie die Auswirkungen von Risiken und Kompromissen zwischen konkurrierenden Interessen oder alternativen Ansätzen. Beispielsweise kann eine beschleunigte Markteinführung neuer Features vor der Kostenoptimierung Vorrang haben, oder Sie können eine relationale Datenbank für nicht relationale Daten wählen, um die Migration eines Systems ohne Faktorwechsel zu vereinfachen. Wägen Sie die Vorteile und Risiken ab, um fundierte Entscheidungen zu treffen, wenn es darum geht, auf welche Bereiche die Anstrengungen konzentriert werden sollen. Einige Risiken oder Entscheidungen können eine bestimmte Zeit lang akzeptabel sein. Es gibt ggf. die Möglichkeit, die damit verbundenen Risiken zu minimieren, oder es ist zu einem bestimmten Zeitpunkt nicht mehr akzeptabel, dass ein Risiko weiterhin bestehen bleibt. In diesem Fall ergreifen Sie Maßnahmen, um das Risiko zu beheben.

Ihre Teams müssen ihre Rolle beim Erreichen von Geschäftsergebnissen verstehen. Teams müssen ihre Rolle für den Erfolg anderer Teams und die Rolle anderer Teams für ihren Erfolg verstehen und gemeinsame Ziele haben. Indem sie die Konzepte Verantwortlichkeit und Zuständigkeit verstehen und wissen, wie Entscheidung getroffen werden und wer dazu berechtigt ist, können ihre Anstrengungen fokussiert und der Nutzen Ihrer Teams maximiert werden. Die Anforderungen eines Teams werden durch den unterstützten Kunden, die Organisation, die Zusammensetzung des Teams und die Merkmale der jeweiligen Workloads beeinflusst. Es ist nicht sinnvoll, davon auszugehen, dass ein einziges Betriebsmodell alle Teams und Workloads in Ihrer Organisation unterstützen kann.

Stellen Sie sicher, dass für jede Anwendung, jede Workload, jede Plattform und jede Infrastrukturkomponente zuständige Besitzer vorhanden sind und dass jeder Prozess und jedes Verfahren einen festen Besitzer hat, der für die Definition verantwortlich ist, und Besitzer, die für die Leistung verantwortlich sind.

Durch das Verständnis für den geschäftlichen Nutzen der einzelnen Komponenten, Prozesse und Verfahren sowie dafür, weshalb diese Ressourcen vorhanden sind oder Aktivitäten ausgeführt werden und warum diese Zuständigkeit besteht, basieren die Aktionen Ihrer Teammitglieder auf fundierten Informationen. Definieren Sie eindeutig die Verantwortlichkeiten der Teammitglieder, damit sie entsprechend handeln und Mechanismen zur Identifizierung von Verantwortlichkeit und Zuständigkeit besitzen. Nutzen Sie entsprechende Mechanismen zum Anfordern von Ergänzungen, Änderungen und Ausnahmen, damit Sie die Innovation nicht einschränken. Definieren Sie Vereinbarungen zwischen Teams, die beschreiben, wie sie für die gegenseitige und die Unterstützung der Geschäftsergebnisse zusammenarbeiten.

Unterstützen Sie Ihre Teammitglieder, damit sie effektiver handeln und positiv zu Ihrem Geschäftsergebnis beitragen können. Die beteiligten Führungskräfte sollten Erwartungen festlegen und den Erfolg messen. Die Geschäftsführung sollte Sponsor, Fürsprecher und treibende Kraft für die Übernahme bewährter Methoden und die Weiterentwicklung der Organisation sein. Lassen Sie die Teammitglieder Maßnahmen ergreifen, wenn Ergebnisse gefährdet sind, um Auswirkungen zu minimieren. Sie müssen dazu ermutigt werden, Entscheidungsträger und Interessenvertreter über ermittelte Risiken zu informieren, damit diese angegangen und Vorfälle vermieden werden können. Kommunizieren Sie bekannte Risiken und geplante Ereignisse zeitnah, klar und umsetzbar, damit Teammitglieder rechtzeitig entsprechende Maßnahmen ergreifen können.

Ermutigen Sie das Ausprobieren neuer Ansätze, damit schneller Erkenntnisse erreicht werden, und sorgen Sie dafür, dass Teammitglieder interessiert und motiviert bleiben. Teams müssen ihre Fähigkeiten erweitern, um neue Technologien einzuführen und Änderungen bei Bedarf und Zuständigkeiten zu unterstützen. Dies sollten sie durch spezielle, strukturierte Lernzeiten unterstützen und ermutigen. Stellen Sie sicher, dass Ihre Teams über die nötigen Ressourcen verfügen (Tools und Teammitglieder), um positiv zu Ihren Geschäftsergebnissen beitragen zu können. Profitieren Sie von der Diversität in der gesamten Organisation, um verschiedene einzigartige Standpunkte zu erfahren. Nutzen Sie diese Perspektive, um Innovation zu fördern, Ihre Annahmen in Frage zu stellen und das Risiko einer Verzerrung durch automatische Bestätigung zu reduzieren. Stärken Sie die Inklusion, Diversität und Zugänglichkeit innerhalb Ihrer Teams, um nützliche Perspektiven zu gewinnen.

Wenn es externe gesetzliche Vorschriften oder Compliance-Anforderungen gibt, die für Ihre Organisation gelten, sollten Sie Ihre Teams mithilfe der von [AWS Cloud Compliance](#) bereitgestellten Ressourcen darin schulen, welche Auswirkungen es bei Ihren Prioritäten zu berücksichtigen gilt. Das Well-Architected Framework legt den Schwerpunkt auf Lernen, Messen und Verbessern. Es bietet einen konsistenten Ansatz, mit dem Sie Architekturen bewerten und Designs implementieren können, die sich im Laufe der Zeit skalieren lassen. AWS stellt das AWS Well-Architected Tool bereit, mit dem Sie Ihren Ansatz vor der Entwicklung, den Status Ihrer Workloads vor der Produktion und

den Status Ihrer Workloads in der Produktion überprüfen können. Sie können Workloads mit den neuesten bewährten Methoden für die AWS-Architektur vergleichen, ihren Gesamtstatus überwachen und Einblicke in potenzielle Risiken erhalten. AWS Trusted Advisor bietet als Tool Zugriff auf verschiedene wichtige Prüfungen, die Optimierungsempfehlungen ausgeben. Diese Informationen können Ihnen beim Festlegen Ihrer Prioritäten helfen. Kunden mit Business und Enterprise Support erhalten Zugriff auf weitere Prüfungen in den Bereichen Sicherheit, Zuverlässigkeit, Leistung, Kostenoptimierung und Nachhaltigkeit, die beim Festlegen von Prioritäten noch hilfreicher sind.

AWS kann Ihnen helfen, Ihre Teams über AWS und die verfügbaren Services zu schulen, sodass alle Mitarbeiter wissen, welche Auswirkungen ihre Entscheidungen auf Ihre Workload haben können. Nutzen Sie bei der Schulung Ihrer Teams die vom AWS Support (AWS Knowledge Center, AWS-Diskussionsforen und AWS Support Center) bereitgestellten Ressourcen und AWS-Dokumente. Wenn Sie eine Frage zu AWS haben, können Sie sich über das AWS Support Center an den AWS Support wenden. AWS stellt in der Amazon Builders' Library auch bewährte Methoden und Muster vor, die wir durch den Betrieb von AWS gelernt haben. Eine Vielzahl weiterer nützlicher Informationen finden Sie im AWS-Blog und im offiziellen AWS-Podcast. AWS Training and Certification bietet einige Schulungen durch digitale Kurse im Selbststudium zu den Grundlagen von AWS. Sie können sich auch für eine Schulung registrieren, die von Dozenten geleitet wird, um die AWS-Fähigkeiten Ihres Teams auszubauen.

Verwenden Sie die Tools oder Services, mit denen Sie Ihre Umgebungen kontenübergreifend verwalten können, z. B. AWS Organizations. Das unterstützt Sie bei der Verwaltung Ihrer Betriebsmodelle. Services wie AWS Control Tower erweitern diese Verwaltungsfunktion, sodass Sie Vorlagen (die Ihre Betriebsmodelle unterstützen) für die Einrichtung von Konten definieren, laufende Governance mit AWS Organizations anwenden und die Bereitstellung neuer Konten automatisieren können. Anbieter von verwalteten Services wie AWS Managed Services, AWS Managed Services-Partner oder Anbieter von verwalteten Services im AWS-Partnernetzwerk stellen Fachwissen zur Implementierung von Cloud-Umgebungen bereit und unterstützen Ihre Sicherheits- und Compliance-Anforderungen und Geschäftsziele. Durch die Erweiterung Ihres Betriebsmodells um verwaltete Services können Sie Zeit und Ressourcen sparen, Ihre internen Teams klein halten und sich auf strategische Ergebnisse konzentrieren, die Ihr Unternehmen auszeichnen, anstatt neue Fähigkeiten und Kompetenzen zu entwickeln.

In den folgenden Fragen geht es um Überlegungen zur betrieblichen Exzellenz. (Eine Liste der Fragen und bewährten Methoden zur betrieblichen Exzellenz finden Sie im [Anhang](#).)

OPS 1: Wie können Sie Ihre Prioritäten bestimmen?

Jeder muss verstehen, welchen Beitrag er zum Geschäftserfolg leistet. Setzen Sie sich gemeinsame Ziele, damit Sie die Prioritäten für Ressourcen festlegen können. Dadurch erzielen Ihre Bemühungen den größtmöglichen Nutzen.

OPS 2: Wie strukturieren Sie Ihr Unternehmen, um die gewünschten Geschäftsergebnisse zu erzielen?

Ihre Teams müssen ihre Rolle beim Erreichen von Geschäftsergebnissen verstehen. Teams müssen ihre Rolle für den Erfolg anderer Teams und die Rolle anderer Teams für ihren Erfolg verstehen und gemeinsame Ziele haben. Indem sie die Konzepte Verantwortlichkeit und Zuständigkeit verstehen und wissen, wie Entscheidung getroffen werden und wer dazu berechtigt ist, können ihre Anstrengungen fokussiert und der Nutzen Ihrer Teams maximiert werden.

OPS 3: Wie unterstützt Ihre Organisationskultur Ihre Geschäftsergebnisse?

Lassen Sie Ihren Teammitgliedern Unterstützung zukommen, damit sie effektiver handeln und Ihr Geschäftsergebnis unterstützen können.

Manchmal kann es vorkommen, dass das Augenmerk zu stark auf eine kleine Auswahl von operativen Prioritäten gerichtet wird. Gehen Sie langfristig gut ausgewogen vor, um sicherzustellen, dass erforderliche Fähigkeiten entwickelt und Risiken verwaltet werden. Überprüfen Sie die Prioritäten regelmäßig und passen Sie sie an geänderte Anforderungen an. Wenn Verantwortlichkeit und Zuständigkeit undefiniert oder unbekannt sind, besteht das Risiko, dass erforderliche Aktionen nicht rechtzeitig ausgeführt werden und redundante und potenziell widersprüchliche Anstrengungen unternommen werden, um diese Anforderungen zu erfüllen. Die Organisationskultur wirkt sich direkt auf die Zufriedenheit und Bindung der Teammitglieder aus. Ermöglichen Sie die Interaktion und aktivieren Sie die Fähigkeiten Ihrer Teammitglieder für den Erfolg Ihres Unternehmens. Durch Experimente werden Innovationen möglich und Ideen zu Ergebnissen. Sie sollten anerkennen, dass unerwünschte Ergebnisse erfolgreiche Experimente sein können, durch die ein Pfad aufgezeigt wurde, der nicht zum Erfolg führt.

Vorbereitung

Zur Vorbereitung auf die betriebliche Exzellenz müssen Sie in Erfahrung bringen, mit welchen Workloads zu rechnen ist und wie diese wahrscheinlich ausfallen werden. Dann können Sie diese so gestalten, dass Sie Einblick in deren Status erhalten und entsprechende Verfahren zu deren Unterstützung entwerfen.

Gestalten Sie Ihre Workload so, dass sie die Informationen bereitstellt, die Sie benötigen, um den internen Status (z. B. Metriken, Protokolle, Ereignisse und Ablaufverfolgungen) über alle Komponenten hinweg zu verstehen. Dies erhöht die Beobachtbarkeit und erleichtert die Untersuchung von Problemen. Beobachtbarkeit geht über die einfache Überwachung hinaus und bietet ein umfassendes Verständnis der internen Funktionsweise eines Systems auf der Grundlage seiner externen Ergebnisse. Beobachtbarkeit basiert auf Metriken, Protokollen und Ablaufverfolgungen und liefert tiefgreifende Erkenntnisse zum Verhalten und zur Dynamik von Systemen. Mit effektiver Beobachtbarkeit können Teams Muster, Anomalien und Trends erkennen, sodass sie potenzielle Probleme proaktiv angehen und einen optimalen Systemzustand aufrechterhalten können. Die Identifizierung von wichtigen Leistungskennzahlen (Key Performance Indicators, KPIs) ist entscheidend, um sicherzustellen, dass die Überwachungsaktivitäten und die Geschäftsziele aufeinander abgestimmt sind. Diese Abstimmung stellt sicher, dass Teams datengestützte Entscheidungen anhand von Metriken treffen, die wirklich wichtig sind, wodurch sowohl die Systemleistung als auch die Geschäftsergebnisse optimiert werden. Darüber hinaus ermöglicht Beobachtbarkeit Unternehmen, proaktiv statt reaktiv zu handeln. Teams können die Ursache-Wirkung-Beziehungen innerhalb ihrer Systeme verstehen und Probleme vorhersagen und verhindern, anstatt nur auf sie zu reagieren. Da sich Workloads weiterentwickeln, ist es wichtig, die Beobachtbarkeitsstrategie immer wieder neu aufzugreifen und zu verfeinern, um sicherzustellen, dass sie relevant und effektiv bleibt.

Verwenden Sie Strategien, die die Übertragung von Änderungen auf die Produktionsumgebung verbessern und einen Faktorwechsel, schnelles Feedback zur Qualität sowie eine schnelle Fehlerbehebung erreichen. Dadurch fließen nützliche Änderungen schneller in die Produktion ein und es treten bei der Bereitstellung weniger Probleme auf. Zudem können Probleme, die durch Bereitstellungsaktivitäten verursacht oder in Ihren Umgebungen erkannt werden, schnell aufgespürt und gelöst werden.

Verwenden Sie Ansätze, die schnelles Feedback zur Qualität liefern und eine schnelle Wiederherstellung bei Änderungen ermöglichen, die nicht zu den gewünschten Ergebnissen führen. Mit diesen Verfahren können Sie die Auswirkung von Problemen eindämmen, die durch Änderungen entstehen. Kalkulieren Sie nicht erfolgreiche Änderungen ein, damit Sie bei Bedarf

schneller reagieren und die vorgenommenen Änderungen testen und validieren können. Achten Sie auf geplante Aktivitäten in Ihren Umgebungen, damit Sie mit dem Risiko von Änderungen umgehen können, die sich auf geplante Aktivitäten auswirken. Nehmen Sie häufige, kleine und umkehrbare Änderungen vor, um den Umfang der Änderungen einzuschränken. Dies beschleunigt die Fehlersuche und ermöglicht eine schnellere Korrektur, da die Möglichkeit besteht, eine Änderung zurückzusetzen. Dies bedeutet auch, dass Sie häufiger von den Vorteilen wertvoller Änderungen profitieren.

Bewerten Sie die operative Bereitschaft Ihrer Workloads, der Prozesse und Verfahren sowie Ihrer Mitarbeiter, damit Sie die operativen Risiken im Zusammenhang mit Ihrer Workload genau kennen. Wenden Sie einen konsistenten Prozess (inklusive manueller und automatisierter Checklisten) an, damit Sie wissen, wann Sie bereit sind, Ihre Workload oder eine Änderung live zu schalten. Auf diese Weise können Sie auch alle Bereiche finden, um die Sie sich kümmern müssen. Ihre routinemäßigen Aktivitäten sollten in Runbooks notiert werden, und Playbooks helfen Ihnen bei der Lösung von Problemen. Machen Sie sich mit den Vorteilen und Risiken vertraut, um fundierte Entscheidungen treffen und Änderungen für die Produktion ermöglichen zu können.

Mit AWS können Sie sämtliche Workloads (Anwendungen, Infrastruktur, Richtlinien, Governance und Betrieb) als Code aufrufen. Das bedeutet, dass Sie für jedes Element Ihres Stacks dieselbe technische Vorgehensweise anwenden können, die Sie für Anwendungscode nutzen. Diese können Sie über Teams oder Organisationen hinweg teilen und damit die Auswirkung der Entwicklungsbemühungen verstärken. Verwenden Sie Operations-as-Code in der Cloud und nutzen Sie die Möglichkeit, sicher zu experimentieren, Ihre Workload und betriebliche Verfahren zu entwickeln und Ausfälle zu üben. Durch den Einsatz von CloudFormation verfügen Sie über konsistente, auf Vorlagen basierende und in einer Sandbox befindliche Entwicklungs-, Test- und Produktionsumgebungen mit steigender betrieblicher Kontrolle.

In den folgenden Fragen geht es um Überlegungen zur betrieblichen Exzellenz.

OPS 4: Wie implementieren Sie die Beobachtbarkeit in Ihre Workload?

Implementieren Sie Beobachtbarkeit in Ihre Workload, damit Sie deren Zustand verstehen und datengesteuerte Entscheidungen auf der Grundlage von Geschäftsanforderungen treffen können.

OPS 5: Wie können Sie Fehler reduzieren, die Fehlerbehebung erleichtern und den Ablauf bis zur Produktion verbessern?

Verwenden Sie Ansätze, die den Fluss von Änderungen in die Produktion verbessern, die einen Faktorwechsel ermöglichen, schnelles Feedback zur Qualität geben und Fehler beheben. Dadurch fließen nützliche Änderungen schneller in die Produktion ein und es treten bei der Bereitstellung weniger Probleme auf. Zudem können Probleme, die durch Bereitstellungsaktivitäten verursacht werden, schnell aufgespürt und gelöst werden.

OPS 6: Wie können Sie Bereitstellungsrisiken eindämmen?

Verwenden Sie Ansätze, die schnelles Feedback zur Qualität liefern und eine schnelle Wiederherstellung bei Änderungen ermöglichen, die nicht zu den gewünschten Ergebnissen führen. Mit diesen Verfahren können Sie die Auswirkung von Problemen eindämmen, die durch Änderungen entstehen.

OPS 7: Wie bringen Sie in Erfahrung, ob Sie für die Unterstützung einer Workload bereit sind?

Bewerten Sie die Betriebsbereitschaft Ihrer Workloads, von Prozessen und Verfahren sowie Ihrer Mitarbeiter, damit Sie die betrieblichen Risiken im Zusammenhang mit Ihrer Workload genau kennen.

Investieren Sie in die Implementierung von Betriebsabläufen als Code, um die Produktivität von Betriebsmitarbeitern zu maximieren, Fehlerraten zu minimieren und automatisierte Reaktionen zu erreichen. Beugen Sie Fehlern nach Möglichkeit vor und stellen Sie entsprechende Abläufe auf. Wenden Sie Metadaten mithilfe von Ressourcen-Tags und AWS -Ressourcengruppen nach einer konsistenten Markierungsstrategie an, um die Identifizierung Ihrer Ressourcen zu erreichen. Versehen Sie Ihre Ressourcen mit Tags für Organisation, Kostenkalkulation, Zugriffssteuerung und Zielrichtung der Ausführung von automatisierten Betriebsaktivitäten. Übernehmen Sie Bereitstellungsmethoden, die die Elastizität der Cloud ausnutzen, um Entwicklungsaktivitäten, die Vorabbereitstellung von Systemen und damit schnellere Implementierungen zu ermöglichen. Wenn Sie an Checklisten, mit denen Sie Ihre Workloads beurteilen, Änderungen vornehmen, bedenken Sie auch, was mit live geschalteten Systemen geschehen soll, die mit den Änderungen nicht mehr kompatibel sind.

Betrieb

Beobachtbarkeit ermöglicht es Ihnen, sich auf aussagekräftige Daten zu konzentrieren und die Interaktionen und Ergebnisse Ihrer Workload zu verstehen. Indem Sie sich auf wichtige Erkenntnisse konzentrieren und unnötige Daten eliminieren, behalten Sie einen einfachen Ansatz zum Verständnis der Workload-Leistung bei. Es ist wichtig, Daten nicht nur zu erfassen, sondern sie auch richtig zu interpretieren. Definieren Sie klare Ausgangswerte, legen Sie geeignete Alarmschwellenwerte fest und überwachen Sie aktiv, ob Abweichungen vorliegen. Wenn eine wichtige Metrik abweicht, insbesondere wenn sie mit anderen Daten korreliert, kann dies spezifische Problembereiche aufzeigen. Mit Beobachtbarkeit sind Sie besser in der Lage, potenzielle Herausforderungen vorherzusehen und zu bewältigen sowie sicherzustellen, dass Ihre Workload reibungslos funktioniert und den Geschäftsanforderungen entspricht.

Der erfolgreiche Betrieb einer Workload wird daran gemessen, ob geschäftliche Ergebnisse erreicht und Kundenanforderungen erfüllt werden. Definieren Sie zu erwartende Ergebnisse, legen Sie fest, wie der Erfolg gemessen wird, und geben Sie an, welche Metriken in Berechnungen verwendet werden sollen, mit denen festgestellt wird, ob Workload und Betrieb erfolgreich sind. Der betriebliche Status beinhaltet sowohl den Status der Workload als auch den Status und Erfolg der betrieblichen Vorgänge, die zur Unterstützung der Workload ausgeführt werden (z. B. Bereitstellung und Vorfallreaktion). Legen Sie Metrikausgangswerte für die Verbesserung, Untersuchung und Intervention fest. Erfassen und analysieren Sie Ihre Metriken und prüfen Sie dann nach, wie weit diese mit ihrem Verständnis von betrieblichen Erfolgen übereinstimmen und welche Änderungen es im zeitlichen Verlauf gibt. Finden Sie anhand gesammelter Metriken heraus, ob kundenseitige und geschäftliche Anforderungen erfüllt werden, und stellen Sie fest, wo noch etwas verbessert werden kann.

Um betriebliche Exzellenz zu erreichen, ist eine effiziente und effektive Verwaltung betrieblicher Ereignisse erforderlich. Dies gilt sowohl für geplante als auch für ungeplante betriebliche Ereignisse. Greifen Sie bei bekannten Ereignissen auf vorab aufgestellte Runbooks zurück. Lassen Sie sich bei der Untersuchung und Behebung von Problemen von Playbooks helfen. Priorisieren Sie Ihre Reaktionen auf Ereignisse anhand der Beeinträchtigungen, die das jeweilige Ereignis für den Geschäftsbetrieb und die Kunden mit sich bringt. Stellen Sie sicher, dass für einen Alarm, der bei einem bestimmten Ereignis ausgelöst werden soll, auch ein auszuführendes Verfahren inklusive eines zuständigen Besitzers festgelegt ist. Legen Sie vorab fest, welche Mitarbeiter für die Behebung eines Ereignisses zuständig sein sollen. Dazu gehören auch Prozesse für einen Eskalationsprozess, über den im Notfall auf der Grundlage der Dringlichkeit und Auswirkungen weitere Mitarbeiter herangezogen werden sollen. Für den Fall, dass eine nicht vorab festgelegte Vorfallreaktion

erforderlich ist, die möglicherweise den geschäftlichen Betrieb beeinträchtigen kann, legen Sie Personen fest, die über die nötige Autorität für Entscheidungen verfügen.

Geben Sie Informationen zum betrieblichen Status von Workloads über Dashboards und Mitteilungen weiter, die auf die Zielgruppe (z. B. Kunde, Unternehmen, Entwickler, Betriebsteam) zugeschnitten sind, damit die jeweiligen Personen geeignete Maßnahmen durchführen können und wissen, wann der normale Betrieb wieder weitergeht.

In AWS können Sie Dashboard-Ansichten Ihrer Metriken generieren, die aus Workloads erfasst wurden oder nativ aus AWS stammen. Sie können CloudWatch oder Anwendungen von Drittanbietern verwenden, um Ansichten von betrieblichen Aktivitäten auf geschäftlicher, Workload-bezogener und betrieblicher Ebene zusammenzustellen und anzuzeigen. AWS stellt über seine Protokollierungsfähigkeiten (wie AWS X-Ray, CloudWatch, CloudTrail und VPC Flow Logs) Einblicke in Workloads bereit. So können Workload-Probleme identifiziert werden, was bei der Ursachenanalyse und Behebung von Fehlern hilft.

In den folgenden Fragen geht es um Überlegungen zur betrieblichen Exzellenz.

OPS 8: Wie nutzen Sie die Beobachtbarkeit von Workloads in Ihrer Organisation?

Sorgen Sie für einen optimalen Zustand der Workload, indem Sie die Beobachtbarkeit nutzen. Nutzen Sie relevante Metriken, Protokolle und Ablaufverfolgungen, um sich einen umfassenden Überblick über die Leistung Ihrer Workload zu verschaffen und Probleme effizient zu beheben.

OPS 9: Wie können Sie den Zustand Ihrer Operationen beurteilen?

Definieren, erfassen und analysieren Sie Metriken für Operationen, um einen Einblick in Ereignisse rund um Ihre Betriebsabläufe zu erhalten. Dies ist wichtig, damit Sie bei Bedarf entsprechende Maßnahmen ergreifen können.

OPS 10: Wie bewältigen Sie Workload- und operationsspezifische Ereignisse?

Erarbeiten und prüfen Sie Verfahren für die Reaktion auf Ereignisse, um Beeinträchtigungen für Ihre Workload zu minimieren.

Alle von Ihnen erfassten Metriken sollten an die geschäftlichen Anforderungen und Ergebnisse angepasst werden, die sie unterstützen. Entwickeln Sie skriptbasierte Antworten auf bekannte Ereignisse und automatisieren Sie deren Leistung als Reaktion auf die Ereigniserkennung.

Weiterentwicklung

Lernen Sie dazu und streben Sie kontinuierliche Verbesserungen an, um nachhaltige betriebliche Exzellenz zu erreichen. Planen Sie Arbeitszyklen ein, um nahezu kontinuierlich kleinere Verbesserungen vorzunehmen. Analysieren Sie nach einem Vorfall alle Ereignisse, die sich auf den Kunden auswirken. Identifizieren Sie die beitragenden Faktoren und Präventivmaßnahmen, um Wiederholungen zu begrenzen oder zu verhindern. Teilen Sie den betroffenen Communitys die beitragenden Faktoren nach Bedarf mit. Beurteilen und priorisieren Sie in regelmäßigen Abständen Möglichkeiten für Verbesserungen (z. B. Anfragen nach Features, Behebung von Problemen, Compliance-Anforderungen), inklusive Workload- und Betriebsverfahren.

Nehmen Sie Feedback-Schleifen in Ihre Verfahren auf, um Verbesserungsmöglichkeiten schnell zu erfahren und Rückmeldungen aus dem Praxisbetrieb zu dokumentieren.

Geben Sie die Dinge, die Sie erfahren, an andere Teams weiter, damit alle davon profitieren. Untersuchen Sie, ob Ihre neuen Erkenntnisse vielleicht Trends aufzeigen, und führen Sie nachträglich teamübergreifende Analysen von operativen Metriken durch, um Verbesserungsmöglichkeiten und -methoden festzustellen. Implementieren Sie Änderungen, die zu Verbesserungen führen sollen, und beurteilen Sie deren Ergebnisse.

In AWS können Sie Ihre Protokolldaten zu Amazon S3 exportieren oder Protokolle zur langfristigen Speicherung direkt an Amazon S3 senden. Mit AWS Glue können Sie Ihre Protokolldaten in Amazon S3 zwecks Analyse erkunden und vorbereiten und die zugehörigen Metadaten im AWS Glue Data Catalog speichern. Amazon Athena kann anschließend durch eine native Integration mit AWS Glue zum Analysieren Ihrer Protokolldaten und Abfragen mit Standard-SQL verwendet werden. Mit einem Business-Intelligence-Tool wie Amazon Quick können Sie Ihre Daten visualisieren, untersuchen und analysieren. Erkennen von Trends und Ereignissen, die zu einer Verbesserung führen können.

In der folgenden Frage geht es um Überlegungen zur betrieblichen Exzellenz.

OPS 11: Wie können Sie Arbeitsvorgänge weiterentwickeln?

Widmen Sie nahezu kontinuierlichen inkrementellen Verbesserungen Zeit und Ressourcen, um die Effektivität und Effizienz Ihrer Betriebsabläufe weiterzuentwickeln.

Die Voraussetzung für eine erfolgreiche Weiterentwicklung des Betriebs sind kontinuierliche kleinere Verbesserungen, das Bereitstellen sicherer Umgebungen und Zeitfenster zum Experimentieren, das Entwickeln und Testen von Verbesserungen sowie die Schaffung eines Umfeldes, in dem alle ermutigt werden, aus Fehlern zu lernen. Die operative Unterstützung für Sandbox-, Entwicklungs-, Test- und Produktionsumgebungen, mit steigenden Levels von operativer Kontrolle erleichtert die Entwicklung und steigert die Kalkulierbarkeit, dass Änderungen zu erfolgreichen Ergebnissen führen.

Ressourcen

Weitere Informationen zu bewährten Methoden für betriebliche Exzellenz finden Sie in den folgenden Ressourcen.

Dokumentation

- [DevOps und AWS](#)

Whitepaper

- [Säule „Betriebliche Exzellenz“](#)

Video

- [DevOps bei Amazon](#)

Sicherheit

In der Säule der Sicherheit wird beschrieben, wie Sie Daten, Systeme und Komponenten so schützen, dass Sie Cloud-Technologien nutzen können, um Ihre Sicherheitslage zu verbessern.

Die Säule „Sicherheit“ gibt einen Überblick über konzeptionelle Grundsätze, bewährte Methoden und Fragen. Verbindliche Anleitungen zur Implementierung finden Sie im [Whitepaper „Säule der Sicherheit“](#).

Themen

- [Designprinzipien](#)
- [Definition](#)
- [Best Practices](#)

- [Ressourcen](#)

Designprinzipien

Die Cloud bietet zahlreiche Möglichkeiten zur Verbesserung Ihrer Workload-Sicherheit:

- Implementieren einer starken Identitätsgrundlage: Etablieren Sie das Prinzip der geringsten Berechtigung und erzwingen Sie die Trennung von Pflichten durch eine entsprechende Autorisierung für jede Interaktion mit Ihren AWS-Ressourcen. Zentralisieren Sie die Identitätsverwaltung und vermeiden Sie die Abhängigkeit von langfristigen statischen Anmeldeinformationen.
- Sicherstellen der Nachverfolgbarkeit: Überwachen, melden und prüfen Sie Aktionen und Änderungen in Ihrer Umgebung in Echtzeit. Integrieren Sie die Protokoll- und Metrikerfassung in Systeme, um automatisch zu untersuchen und Maßnahmen zu ergreifen.
- Sicherheit auf allen Ebenen: Etablieren Sie eine Abwehrstrategie mit mehreren Sicherheitsmechanismen. Wenden Sie diesen auf allen Ebenen an (z. B. Netzwerkgrenzen, VPC, Lastverteilung, alle Instances und Datenverarbeitungsservices, Betriebssystem, Anwendung und Code).
- Automatisieren bewährter Sicherheitsverfahren: Mithilfe automatisierter softwarebasierter Sicherheitsmechanismen können Sie Ihr System sicher, schnell und kosteneffektiv skalieren. Erstellen Sie sichere Architekturen, einschließlich implementierter Kontrollen, die als Code in versionsgesteuerten Vorlagen definiert und verwaltet werden.
- Schutz von Daten während der Übertragung und im Ruhezustand: Klassifizieren Sie Ihre Daten nach Sensibilität und nutzen sie Mechanismen wie Verschlüsselung, Tokenisierung der Daten und Zugriffskontrolle.
- Trennen von Benutzern und Daten: Nutzen Sie Mechanismen und Tools, um den direkten Zugriff auf Daten zu minimieren/verhindern und das manuelle Verarbeiten Ihrer Daten zu reduzieren. Sie reduzieren dadurch das Risiko, dass sensible Daten verloren gehen, geändert werden oder anderweitigen Benutzerfehlern unterliegen.
- Vorbereitung auf Sicherheitsereignisse: Seien Sie auf Vorfälle vorbereitet. Richten Sie entsprechend Ihren organisatorischen Anforderungen ein Verfahren zum Vorfalldmanagement sowie Richtlinien für die Überprüfung ein. Simulieren Sie Vorfalldreaktionen und nutzen Sie automatisierbare Tools, um die Erkennung, Untersuchung und Wiederherstellung zu beschleunigen.

Definition

Es gibt sieben bewährte Methoden für die Sicherheit in der Cloud:

- Sicherheitsgrundlagen
- Identity and Access Management
- Erkennung
- Schutz der Infrastruktur
- Datenschutz
- Vorfallreaktion
- Anwendungssicherheit

Vor der Entwicklung von Workloads ist es wichtig, geeignete Sicherheitsverfahren festzulegen. Sie müssen die einzelnen Prozesse steuern können. Wichtig ist auch, dass Sie Sicherheitsvorfälle erkennen, Ihre Systeme und Services schützen und die Vertraulichkeit und Integrität von Daten durch entsprechende Schutzmaßnahmen wahren können. Richten Sie ein gut definiertes und geübtes Verfahren ein, das es Ihnen ermöglicht, auf Sicherheitsvorfälle zu reagieren. Derartige Tools und Techniken sind unabdinglich, um Ihr Unternehmen vor finanziellen Verlusten zu schützen und gesetzliche Vorgaben zu erfüllen.

Das AWS-Modell der gemeinsamen Verantwortlichkeit hilft Organisationen dabei, durch die Migration zur Cloud ihre Sicherheits- und Compliance-Ziele zu erfüllen. Dadurch, dass sich AWS um den physischen Schutz der Infrastruktur unserer Cloud-Services kümmert, können Sie sich als AWS-Kunden darauf konzentrieren, mithilfe unserer Services Ihre Ziele zu erreichen. Sie haben in der AWS Cloud auch einen verbesserten Zugriff auf Sicherheitsdaten und können automatisch auf Sicherheitsereignisse reagieren.

Best Practices

Themen

- [Sicherheitsgrundlagen](#)
- [Identity and Access Management](#)
- [Erkennung](#)
- [Schutz der Infrastruktur](#)
- [Datenschutz](#)

- [Vorfallreaktion](#)
- [Anwendungssicherheit](#)

Sicherheitsgrundlagen

In der folgenden Frage geht es um Überlegungen zur Sicherheit. (Eine Liste der Fragen und bewährten Methoden zur Sicherheit finden Sie im [Anhang](#).).

SICH 1: Wie können Sie Ihre Workload sicher betreiben?

Um Ihre Workload sicher zu betreiben, müssen Sie in allen Sicherheitsbereichen übergreifende bewährte Methoden anwenden. Wenden Sie die Anforderungen und Prozesse, die Sie im Bereich Operational Excellence auf Organisations- und Workload-Ebene definiert haben, auf alle Bereiche an.

Bleiben Sie auf dem Laufenden mit Empfehlungen von AWS, branchenspezifischen Quellen sowie Informationsquellen zu Bedrohungen, um Ihr Bedrohungsmodell und Ihre Kontrollziele weiterzuentwickeln. Durch die Automatisierung von Sicherheitsprozessen, Tests und Validierungen können Sie Ihre Sicherheitsabläufe skalieren.

In AWS empfehlen wir die Trennung verschiedener Workloads nach Konto basierend auf ihrer Funktion und Compliance- oder der Datensensibilität.

Identity and Access Management

Das Identity and Access Management ist ein wichtiger Bestandteil eines Informationssicherheitsprogramms. Es stellt sicher, dass nur autorisierte und authentifizierte Benutzer in dem von Ihnen gewünschten Umfang auf Ihre Ressourcen zugreifen können. Definieren Sie beispielsweise Prinzipien (d. h. Konten, Benutzer, Rollen und Services, die Aktionen in Ihrem Konto durchführen), erstellen Sie entsprechende Richtlinien, und implementieren Sie eine strenge Verwaltung von Anmeldeinformationen. Diese Elemente der Rechteverwaltung bilden die Grundlage der Authentifizierung und Autorisierung.

In AWS erfolgt die Rechteverwaltung primär durch den AWS Identity and Access Management (IAM)-Service. Damit können Sie sowohl den Benutzerzugriff als auch den programmgesteuerten Zugriff auf AWS-Services und -Ressourcen steuern. Wenden Sie detaillierte Richtlinien an, um Benutzern, Gruppen, Rollen oder Ressourcen Berechtigungen zuzuweisen. Darüber hinaus können

Sie die Verwendung starker Kennwörter erzwingen. Sie können deren Komplexität vorgeben, Wiederverwendungen vermeiden und Multi-Faktor-Authentifizierung (MFA) nutzen. Sie haben die Möglichkeit, die Rechteverwaltung mit Ihrem Verzeichnisdienst zu verbinden. Wenn Sie Workloads haben, die Zugriff auf AWS erfordern, ermöglicht IAM diesen sicher durch Rollen, Instance-Profile, Identitätsverbund und temporäre Anmeldeinformationen.

In den folgenden Fragen geht es um Überlegungen zur Sicherheit.

SICH 2: Wie verwalten Sie Identitäten für Personen und Maschinen?

Es gibt zwei Arten von Identitäten, die Sie für den Betrieb von sicheren AWS-Workloads verwalten müssen. Wenn Sie verstehen, welche Arten von Identitäten Sie verwalten und Zugriff gewähren müssen, können Sie sicherstellen, dass die richtigen Identitäten unter den richtigen Bedingungen Zugriff auf die richtigen Ressourcen haben.

Menschliche Identitäten: Ihre Administratoren, Entwickler, Mitarbeiter und Endbenutzer benötigen eine Identität, um auf Ihre AWS-Umgebungen und Anwendungen zugreifen zu können. Dies sind Mitglieder Ihrer Organisation oder externe Benutzer, mit denen Sie zusammenarbeiten und die über einen Webbrowser, eine Client-Anwendung oder interaktive Befehlszeilentools mit Ihren AWS-Ressourcen interagieren.

Maschinenidentitäten: Ihre Serviceanwendungen, Betriebstools und Workloads erfordern eine Identität, um Anforderungen an AWS-Services zu senden, z. B. um Daten zu lesen. Zu diesen Identitäten gehören Maschinen, die in Ihrer AWS-Umgebung ausgeführt werden, wie z. B. Amazon-EC2-Instances oder AWS Lambda-Funktionen. Sie können auch Maschinenidentitäten für externe Parteien verwalten, die Zugriff benötigen. Darüber hinaus könnten Sie Maschinen außerhalb von AWS haben, die Zugang zu Ihrer AWS-Umgebung benötigen.

SICH 3: Wie verwalten Sie Berechtigungen für Personen und Maschinen?

Verwalten Sie Berechtigungen zum Steuern des Zugriffs auf Personen- und Maschinenidentitäten, die Zugriff auf AWS und Ihre Workload benötigen. Berechtigungen steuern, wer worauf und unter welchen Bedingungen zugreifen kann.

Anmeldeinformationen dürfen nicht zwischen Benutzern oder Systemen weitergegeben werden. Der Benutzerzugriff sollten nach dem Prinzip der geringsten Berechtigung erfolgen, passwortgeschützt

sein und nur mittels MFA möglich sein. Der programmgesteuerte Zugriff etwa durch API-Aufrufe von AWS-Services sollte mit temporären Anmeldeinformationen mit eingeschränkten Berechtigungen erfolgen, die beispielsweise durch den AWS -Security-Token-Service ausgegeben werden.

Benutzer benötigen programmgesteuerten Zugriff, wenn sie außerhalb der AWS-Managementkonsole mit AWS interagieren möchten. Die Vorgehensweise, um programmgesteuerten Zugriff zu gewähren, hängt davon ab, welcher Benutzertyp auf zugreift AWS.

Um Benutzern programmgesteuerten Zugriff zu gewähren, wählen Sie eine der folgenden Optionen.

Welcher Benutzer benötigt programmgesteuerten Zugriff?	Bis	Von
IAM	(Empfohlen) Verwenden Sie Konsolen-Anmeldeinformationen als temporäre Anmeldeinformationen, um programmgesteuerte Anforderungen an die AWS CLI, AWS-SDKs oder AWS-APIs zu signieren.	Befolgen Sie die Anweisungen für die Schnittstelle, die Sie verwenden möchten. • Informationen zur AWS CLI finden Sie unter Anmeldung für lokale AWS-Entwicklung im AWS Command Line Interface-Benutzerhandbuch. • Informationen zu AWS-SDKs finden Sie unter Anmeldung für lokale AWS-Entwicklung im Referenzhandbuch zu AWS-SDKs und -Tools.
Mitarbeiteridentität (Benutzer, die in IAM Identity Center verwaltet werden)	Verwenden Sie temporäre Anmeldeinformationen, um programmgesteuerte Anforderungen an die AWS CLI, AWS SDKs oder AWS-APIs zu signieren.	Befolgen Sie die Anweisungen für die Schnittstelle, die Sie verwenden möchten. • Informationen zur AWS CLI finden Sie unter Konfigurieren der AWS CLI für die Verwendung von AWS IAM

Welcher Benutzer benötigt programmgesteuerten Zugriff?	Bis	Von
		Identity Center im AWS Command Line Interface-Benutzerhandbuch. • Informationen zu AWS-SDKs, Tools und AWS-APIs finden Sie unter IAM-Identity-Center-Authentifizierung im Referenzhandbuch zu AWS-SDKs und Tools.
IAM	Verwenden Sie temporäre Anmeldeinformationen, um programmgesteuerte Anforderungen an die AWS CLI, AWS-SDKs oder AWS-APIs zu signieren.	Folgen Sie den Anweisungen unter Verwenden temporärer Anmeldeinformationen mit AWS-Ressourcen im IAM-Benutzerhandbuch.

Welcher Benutzer benötigt programmgesteuerten Zugriff?	Bis	Von
IAM	(Nicht empfohlen) Verwenden Sie langfristige Anmeldeinformationen, um programmgesteuerte Anforderungen an die AWS CLI, AWS-SDKs oder AWS-APIs zu signieren.	Befolgen Sie die Anweisungen für die Schnittstelle, die Sie verwenden möchten. • Informationen zur AWS CLI finden Sie unter Authentifizierung mit IAM-Benutzer-Anmeldeinformationen im AWS Command Line Interface-Benutzerhandbuch. • Informationen zu AWS SDKs und Tools finden Sie unter Authentifizierung mit langfristigen Anmeldeinformationen im Referenzhandbuch zu AWS SDKs und Tools. • Informationen zu AWS-APIs finden Sie unter Verwalten von Zugriffsschlüsseln für IAM-Benutzer im IAM-Benutzerhandbuch.

AWS bietet Ressourcen, die Ihnen das Identity and Access Management erleichtern. Mehr zu den bewährten Methoden erfahren Sie in unseren praktischen Übungen zu den Themen [Verwaltung von Anmeldeinformationen und Authentifizierung](#), [Steuerung des Benutzerzugriffs](#) und [Steuerung des programmgesteuerten Zugriffs](#).

Erkennung

Aufdeckende Kontrollen bieten Ihnen die Möglichkeit, potenzielle Sicherheitsbedrohungen oder -vorfälle zu erkennen. Die Kontrollmechanismen sind ein wesentlicher Bestandteil von

Governance-Frameworks. Sie können zur Unterstützung von Qualitätssicherungsverfahren, zur Einhaltung gesetzlicher Vorgaben und Pflichten sowie zur Erkennung und Abwehr von Bedrohungen genutzt werden. Es gibt unterschiedliche Arten aufdeckender Kontrollen. Eine Bestandserfassung der Ressourcen und ihrer detaillierten Attribute trägt beispielsweise zu einer effektiveren Entscheidungsfindung (und Lebenszyklussteuerung) bei, wenn es darum geht, operative Ausgangswerte festzulegen. Sie können auch durch eine interne Prüfung der mit Informationssystemen verbundenen Steuerelemente sicherstellen, dass Ihre Verfahren den Richtlinien und Anforderungen entsprechen. Basierend auf definierten Bedingungen sind passende automatisierte Benachrichtigungen möglich. Diese Steuerelemente sind wichtige reaktive Faktoren, die es Ihrer Organisation ermöglichen, den Umfang anomaler Aktivitäten zu ermitteln und zu verstehen.

In AWS können Sie aufdeckende Kontrollen durch Verarbeitungsprotokolle, Ereignisse und Überwachungsfunktionen implementieren, die eine Prüfung, automatisierte Analyse und Benachrichtigung ermöglichen. Mit CloudTrail-Protokollen, AWS API-Aufrufen und CloudWatch können Sie Kennzahlen überwachen und Benachrichtigungen senden. Der Konfigurationsverlauf ist mit AWS Config einsehbar. Amazon GuardDuty ist ein verwalteter Service zur Bedrohungserkennung, der Ihre AWS-Konten und -Workloads zu deren Schutz fortlaufend auf böswillige oder unbefugte Verhaltensweisen überwacht. Mit Serviceprotokollen – etwa von Amazon Simple Storage Service (Amazon S3) – können Sie Zugriffsanfragen protokollieren.

In der folgenden Frage geht es um Überlegungen zur Sicherheit.

SICH 4: Wie erkennen und untersuchen Sie Sicherheitsereignisse?

Erfassen und analysieren Sie Ereignisse anhand von Protokollen und Metriken, um mehr Transparenz zu erhalten. Reagieren Sie auf Sicherheitsereignisse und potenzielle Bedrohungen, um Ihre Workload zu schützen.

Die Protokollverwaltung ist für eine Well-Architected-Workload wichtig, um so vielfältige Bereiche wie Sicherheit, Forensik sowie die Einhaltung gesetzlicher Vorgaben abzudecken. Um potenzielle Sicherheitsvorfälle ermitteln zu können, müssen diese Protokolle analysiert und bei Bedarf entsprechende Maßnahmen ergriffen werden. AWS bietet Funktionen, die die Protokollverwaltung erleichtern. Sie können damit einen Lebenszyklus für die Datenaufbewahrung festlegen oder angeben, wo Daten gespeichert, archiviert oder schließlich gelöscht werden. Dies vereinfacht die vorhersehbare und zuverlässige Datenverarbeitung und senkt die Kosten.

Schutz der Infrastruktur

Zum Schutz der Infrastruktur sind Steuermethoden wie etwa eine tiefgreifende Abwehr erforderlich, um Best Practices sowie organisatorische und gesetzliche Verpflichtungen zu erfüllen. Die Nutzung dieser Methoden ist für erfolgreiche, kontinuierliche Betriebsabläufe sowohl in der Cloud als auch On-Premises ausschlaggebend.

AWS ermöglicht die Überprüfung zustandsbehafteter und zustandsloser Pakete. Sie können dafür wahlweise AWS-native Technologien oder im AWS Marketplace angebotene Partnerprodukte und -services nutzen. Amazon Virtual Private Cloud (Amazon VPC) wird empfohlen, um eine private, sichere und skalierbare Umgebung zu erstellen, in der Sie Ihre Topologie, einschließlich Gateways, Routing-Tabellen sowie öffentlichen und privaten Subnetzen definieren können.

In den folgenden Fragen geht es um Überlegungen zur Sicherheit.

SICH 5: Wie schützen Sie Ihre Netzwerkressourcen?

Alle Workloads, die über eine Art Netzwerkverbindung verfügen, unabhängig davon, ob es sich um das Internet oder ein privates Netzwerk handelt, erfordern mehrere Abwehrebenen, um Schutz vor externen und internen Netzwerkbedrohungen sicherzustellen.

SICH 6: Wie schützen Sie Ihre Datenverarbeitungsressourcen?

Die Rechenressourcen in Ihrer Workload erfordern mehrere Verteidigungsebenen, um sie vor externen und internen Bedrohungen zu schützen. Zu den Rechenressourcen gehören u. a. EC2-Instances, Container, AWS Lambda-Funktionen, Datenbankservices und IoT-Geräte.

Ungeachtet der Umgebung sollten mehrere Abwehrebenen vorhanden sein. Was den Schutz der Infrastruktur anbelangt, gelten viele der Konzepte und Methoden für Cloud- und lokale Modelle gleichermaßen. Das Erzwingen des Grenzschatzes, die Überwachung von Ein- und Ausgangspunkten sowie die umfassende Protokollierung, Überwachung und Benachrichtigung sind für einen effektiven Informationssicherheitsplan wichtig.

AWS-Kunden können die Konfiguration der Amazon Elastic Compute Cloud (Amazon EC2) sowie von Amazon Elastic Container Service (Amazon ECS)-Containern (Amazon ECS) und AWS Elastic Beanstalk-Instances anpassen oder härten und in einem unveränderlichen Amazon Machine Image (AMI) speichern. Dadurch erhalten alle neuen virtuellen Server (Instances), die mit diesem AMI

gestartet werden, diese gehärtete Konfiguration. Dabei spielt es keine Rolle, ob sie durch Auto Scaling oder manuell gestartet wurden.

Datenschutz

Vor der Entwicklung eines Systems sollten grundlegende Sicherheitspraktiken implementiert werden. Mittels Datenklassifizierung lassen sich beispielsweise organisatorische Daten nach Sensitivität kategorisieren. Die Verschlüsselung macht sie zudem für unbefugte Benutzer unleserlich. Derartige Tools und Techniken sind unabdinglich, um Ihr Unternehmen vor finanziellen Verlusten zu schützen und gesetzliche Vorgaben zu erfüllen.

In AWS können Sie Daten mit folgenden Maßnahmen schützen:

- Als AWS-Kunden behalten Sie die vollständige Kontrolle über Ihre Daten.
- AWS erleichtert Ihnen die Datenverschlüsselung und die Schlüsselverwaltung, einschließlich einer regulären Schlüsselrotation. Sie können diese auf einfache Weise selbst verwalten oder von AWS automatisieren lassen.
- Sie haben Zugriff auf detaillierte Protokolle mit wichtigen Angaben etwa zu Dateizugriffen und -änderungen.
- Die Speichersysteme von AWS zeichnen sich durch eine exzeptionelle Ausfallsicherheit aus. Amazon S3 Standard, S3 Standard-IA, S3 One Zone-IA und Amazon Glacier bieten beispielsweise eine einjährige Objektlanglebigkeit von 99,999999999 %. Dies entspricht einem jährlichen erwarteten Verlust von 0,000000001 % der Objekte.
- Die Versionsverwaltung, die in ein umfassenderes Verfahren zur Datenlebenszyklusverwaltung eingebunden sein kann, bietet Schutz vor versehentlichen Überschreibungen, Löschungen und ähnlichen Gefahren.
- AWS veranlasst niemals eine Verschiebung von Daten zwischen Regionen. Die in einer Region platzierten Inhalte bleiben in dieser Region, sofern Sie eine Verschiebung nicht ausdrücklich mithilfe eines Features oder Services durchführen.

In den folgenden Fragen geht es um Überlegungen zur Sicherheit.

SICH 7: Wie klassifizieren Sie Ihre Daten?

Die Datenklassifizierung bietet eine Möglichkeit, Daten basierend auf Wichtigkeit und Sensibilität zu kategorisieren, um Ihnen dabei zu helfen, angemessene Schutz- und Aufbewahrungskontrollen zu bestimmen.

SICH 8: Wie schützen Sie Ihre Daten im Ruhezustand?

Schützen Sie Ihre Daten im Ruhezustand, indem Sie mehrere Kontrollen implementieren und so das Risiko eines unbefugten Zugriffs oder einer falschen Handhabung verringern.

SICH 9: Wie schützen Sie Ihre Daten während der Übertragung?

Schützen Sie Ihre Daten während der Übertragung, indem Sie mehrere Kontrollen implementieren und so das Risiko eines unbefugten Zugriffs oder eines Datenverlusts verringern.

AWS bietet mehrere Möglichkeiten zur Verschlüsselung von Daten im Ruhezustand und während der Übertragung. Unsere Services enthalten Funktionen, die die Verschlüsselung Ihrer Daten erleichtern. Wir haben beispielsweise in Amazon S3 eine serverseitige Verschlüsselung (SSE) implementiert, die die Speicherung Ihrer Daten in verschlüsselter Form vereinfacht. Sie können auch das komplette Ver- und -Entschlüsselungsverfahren mit HTTPS (generell als SSL-Terminierung bekannt) mit Elastic Load Balancing (ELB) arrangieren.

Vorfallreaktion

Obwohl die präventiven und aufdeckenden Kontrollen mittlerweile extrem ausgereift sind, sollte Ihr Unternehmen dennoch Verfahren etablieren, um auf Sicherheitsvorfälle reagieren und mögliche Auswirkungen mindern zu können. Wie effektiv Ihre Teams bei einem Vorfall reagieren können, um Systeme zu isolieren oder zu bergen und Betriebsabläufe in einem bekanntermaßen funktionierenden Zustand wiederherzustellen, hängt stark von der Architektur des Workloads ab. Indem Sie sich mit entsprechenden Tools und Zugriffsmöglichkeiten auf Sicherheitsvorfälle vorbereiten und die Vorfallreaktion regelmäßig im Rahmen von Gamedays üben, stellen Sie eine zeitnahe Untersuchung und Wiederherstellung sicher.

In AWS ermöglichen die folgenden Praktiken eine effektive Vorfallreaktion:

- Eine detaillierte Protokollierung wichtiger Informationen etwa zu Dateizugriffen und -änderungen.
- Ereignisse können automatisch verarbeitet werden und Tools starten, die Reaktionen über AWS APIs automatisieren.
- Sie können vorab mit AWS CloudFormation entsprechende Tools und einen „Clean Room“ bereitstellen. Sie erhalten dadurch eine sichere, isolierte Umgebung für forensische Untersuchungen.

In der folgenden Frage geht es um Überlegungen zur Sicherheit.

SICH 10: Wie können Sie Vorfälle voraussagen, darauf reagieren und diese beheben?

Die Vorbereitung ist entscheidend für eine rechtzeitige und effektive Untersuchung, Reaktion auf und Wiederherstellung nach Sicherheitsvorfällen, um Unterbrechungen der Geschäftsabläufe zu minimieren.

Wichtig ist, dass Sie eine Möglichkeit haben, Ihrem Sicherheitsteam für forensische Zwecke schnell Zugriff zu gewähren. Automatisieren Sie sowohl die Isolation von Instances als auch die Erfassung von Daten und Zuständen.

Anwendungssicherheit

Anwendungssicherheit (Application Security, AppSec) beschreibt den gesamten Prozess des Entwurfs, der Erstellung und Tests der Sicherheitseigenschaften von Workloads, die Sie entwickeln. Sie sollten die Personen in Ihrer Organisation entsprechend geschult haben und die Sicherheitseigenschaften Ihrer Entwicklung und der Infrastruktur Ihrer Softwareveröffentlichung verstehen. Sie sollten auch Automatisierung zum Identifizieren von Sicherheitsproblemen einsetzen.

Das Einführen von Anwendungssicherheitstests als Teil Ihres Softwareentwicklungs-Lebenszyklus (SDLC) sowie des Prozesses nach der Veröffentlichung hilft Ihnen dabei, sicherzustellen, dass Sie über einen strukturierten Mechanismus zum Identifizieren, Lösen und Verhindern von Anwendungssicherheitsproblemen verfügen, die sich in Ihre Produktionsumgebung einschleichen könnten.

Ihre Methodologie zur Anwendungsentwicklung sollte Sicherheitskontrollen enthalten, während Sie Ihre Workloads entwerfen, entwickeln, bereitstellen und ausführen. Während Sie das machen, passen Sie den Prozess für kontinuierliche Fehlerrückmeldung und Minimierung von technischen Schulden an. Das Verwenden von Bedrohungsmodellierung in der Designphase hilft Ihnen beispielsweise dabei, Designfehler früh aufzudecken, wodurch sie einfacher und günstiger behoben werden können – im Gegensatz dazu, wenn Sie warten und die Fehler später beseitigen.

Je früher Fehler im Softwareentwicklungs-Lebenszyklus behoben werden, desto geringer sind die Kosten und die Komplexität. Die einfachste Weise, Probleme zu lösen, ist keine zu haben. Daher hilft Ihnen ein Bedrohungsmodell, sich bereits in der Designphase auf die richtigen Ergebnisse zu konzentrieren. Während Ihr AppSec-Programm reift, können Sie mithilfe von Automatisierung die

Anzahl an durchgeführten Tests erhöhen, die Genauigkeit des Feedbacks für Entwickler verbessern und die für Sicherheitsüberprüfungen aufgewendete Zeit verringern. All diese Aktionen erhöhen die Qualität der Software, die Sie entwickeln, und beschleunigen das Ausliefern von Funktionen in die Produktion.

Bei diesen Implementierungsrichtlinien gibt es vier Schwerpunktbereiche: Organisation und Kultur, Sicherheit der Pipeline, Sicherheit in der Pipeline und Verwaltung von Abhängigkeiten. Jeder Bereich bietet einen Satz an Prinzipien, die Sie implementieren können, und eine bietet umfassende Sicht darauf, wie Sie Ihre Workloads entwerfen, entwickeln, aufbauen, bereitstellen und ausführen.

In AWS gibt es eine Reihe von Ansätzen, die Sie in Zusammenhang mit Ihrem Anwendungssicherheitsprogramm verwenden können. Einige dieser Ansätze basieren auf Technologie, während sich andere auf die menschlichen und betrieblichen Aspekte Ihres Anwendungssicherheitsprogramms konzentrieren.

In der folgenden Frage geht es um Überlegungen zur Anwendungssicherheit.

SICH 11: Wie beziehen Sie die Sicherheitseigenschaften von Anwendungen während des gesamten Entwurfs-, Entwicklungs- und Bereitstellungslebenszyklus ein und validieren sie?

Das Schulen von Personen, das Testen mithilfe von Automatisierung, ein Verständnis der Abhängigkeiten und die Validierung der Sicherheitseigenschaften von Tools und Anwendungen helfen dabei, die Wahrscheinlichkeit eines Sicherheitsproblems bei Produktions-Workloads zu verringern.

Ressourcen

Werfen Sie einen Blick auf die folgenden Ressourcen, um mehr über unsere bewährten Methoden für die Sicherheit zu erfahren.

Dokumentation

- [AWS Cloudsicherheit](#)
- [AWS Compliance](#)
- [AWS Blog zum Thema Sicherheit](#)
- [AWS Modell des Sicherheitsreifegrads](#)

Whitepaper

- [Säule der Sicherheit](#)
- [AWS Übersicht über die Sicherheit in](#)
- [AWS Risiko und Compliance](#)

Video

- [AWS Lage der -Sicherheit](#)
- [Übersicht über die gemeinsame Verantwortlichkeit](#)

Zuverlässigkeit

Die Säule „Zuverlässigkeit“ umfasst die Fähigkeit einer Workload, die beabsichtigte Funktion erwartungsgemäß korrekt und konsistent auszuführen. Dies umfasst die Möglichkeit, die Workload während des gesamten Lebenszyklus zu betreiben und zu testen. Dieses Dokument bietet umfassende Informationen mit bewährten Methoden für die Implementierung zuverlässiger Workloads in AWS.

Die Säule „Zuverlässigkeit“ gibt einen Überblick über konzeptionelle Grundsätze, bewährte Methoden und Fragen. Verbindliche Anleitungen zur Implementierung finden Sie im [Whitepaper „Säule der Zuverlässigkeit“](#).

Themen

- [Designprinzipien](#)
- [Definition](#)
- [Bewährte Methoden](#)
- [Ressourcen](#)

Designprinzipien

Es gibt fünf Gestaltungsprinzipien für die Zuverlässigkeit in der Cloud:

- Automatische Wiederherstellung nach einem Fehler: Durch die Überwachung wichtiger Leistungskennzahlen (KPIs) einer Workload können Sie die Automatisierung starten, sobald ein Schwellenwert überschritten wurde. Diese KPIs sollten als Kennzahlen für den Geschäftswert

und nicht als technische Aspekte für den Betrieb des Service betrachtet werden. Dies bietet eine automatische Benachrichtigung bei und Verfolgung von Fehlern sowie die Einleitung einer automatisierten Wiederherstellung, die eine Fehlerumgehung bietet oder den Fehler behebt. Bei einer ausgefeilteren Automatisierung ist es möglich, Fehler vor ihrem eigentlichen Auftreten zu antizipieren und zu beheben.

- **Testen von Wiederherstellungsverfahren:** In einer On-Premises-Umgebung werden häufig Tests durchgeführt, um nachzuweisen, dass die Workload in einem bestimmten Szenario funktioniert. Mit den Tests werden in der Regel keine Wiederherstellungsstrategien validiert. In der Cloud können Sie testen, in welchen Situationen die Workload Fehler produziert, und Sie können die Wiederherstellungsverfahren validieren. Mit der Automatisierung können Sie verschiedene Fehler simulieren oder Szenarien reproduzieren, die zuvor zu Fehlern geführt haben. Dieser Ansatz stellt Fehlerpfade bereit, die Sie testen und beheben können, bevor ein echtes Fehlerszenario auftritt, und reduziert so das Risiko.
- **Horizontale Skalierung zur Erhöhung der aggregierten Workload-Verfügbarkeit:** Ersetzen Sie eine große Ressource durch mehrere kleine Ressourcen, um die Auswirkung eines einzigen Fehlers auf die Gesamt-Workload zu reduzieren. Verteilen Sie Anfragen auf mehrere kleinere Ressourcen, um sicherzustellen, dass sie keine gemeinsame Fehlerquelle haben.
- **Genaue Analyse der verfügbaren Kapazität:** Eine häufige Fehlerursache bei On-Premises-Workloads ist die Ressourcensättigung. Ein solches Szenario liegt vor, wenn die Anforderungen an die Workload die Kapazität dieser Workload überschreiten (dies ist häufig das Ziel von Denial-of-Service-Angriffen). In der Cloud können Sie die Nachfrage und die Workload-Auslastung überwachen und das Hinzufügen oder Entfernen von Ressourcen automatisieren, um den Bedarf ohne Über- oder Unterbereitstellung stets effizient zu erfüllen. Es gibt weiterhin Grenzen, aber einige Kontingente können gesteuert und andere verwaltet werden (siehe Verwaltung von Service Quotas und Einschränkungen).
- **Änderungsmanagement per Automatisierung:** Änderungen an Ihrer Infrastruktur sollten über eine Automatisierung vorgenommen werden. Zu den Änderungen, die verwaltet werden müssen, gehören Änderungen an der Automatisierung, die anschließend nachverfolgt und überprüft werden können.

Definition

Die bewährten Methoden für Zuverlässigkeit in der Cloud lassen sich in vier Bereiche einteilen:

- Grundlagen
- Workload-Architektur

- Änderungsmanagement
- Fehlerverwaltung

Um Zuverlässigkeit zu erreichen, müssen Sie mit den Grundlagen beginnen – einer Umgebung, in der Service Quotas und Netzwerktopologie der Workload entsprechen. Die Workload-Architektur des verteilten Systems muss so ausgelegt sein, dass Ausfälle verhindert und minimiert werden. Die Workload muss Änderungen in Bezug auf den Bedarf oder die Anforderungen verarbeiten und so konzipiert sein, dass sie Fehler erkennt und sie automatisch selbst behebt.

Bewährte Methoden

Themen

- [Grundlagen](#)
- [Workload-Architektur](#)
- [Änderungsmanagement](#)
- [Fehlerverwaltung](#)

Grundlagen

Grundlegende Anforderungen sind diejenigen, deren Umfang über eine einzelne Workload oder ein einzelnes Projekt hinausgeht. Vor dem Aufbau der Architektur eines System sollten grundlegende Anforderungen, die sich auf die Zuverlässigkeit auswirken, implementiert werden. So müssen Sie beispielsweise Ihre Rechenzentren mit einer ausreichenden Netzwerkbandbreite versorgen.

Bei AWS sind die meisten dieser grundlegenden Anforderungen jedoch bereits berücksichtigt oder können nach Bedarf adressiert werden. Die Cloud ist vom Konzept her nahezu unbegrenzt. Daher liegt es in der Verantwortung von AWS, die Anforderungen an ausreichende Netzwerk- und Datenverarbeitungskapazität zu erfüllen, sodass Sie die Größe und Zuweisungen von Ressourcen bedarfsgerecht ändern können.

In den folgenden Fragen geht es um Überlegungen zur Zuverlässigkeit. (Eine Liste der Fragen und bewährten Methoden zur Zuverlässigkeit finden Sie im [Anhang](#).)

ZUV 1: Wie werden Service Quotas und Serviceeinschränkungen verwaltet?

Für cloudbasierte Workload-Architekturen gibt es Service Quotas (auch als „Servicebeschränkungen“ bezeichnet). Mit diesen Kontingenten soll verhindert werden, dass versehentlich mehr

ZUV 1: Wie werden Service Quotas und Serviceeinschränkungen verwaltet?

Ressourcen bereitgestellt werden, als Sie benötigen, und es sollen die Anforderungsraten bei API-Vorgängen begrenzt werden, um Services vor Missbrauch zu schützen. Es gibt auch Einschränkungen für Ressourcen, z. B. im Bezug auf die Rate, mit der Bits über ein Glasfaserkabel übertragen werden können, oder die Menge an Speicherplatz auf einer physischen Festplatte.

ZUV 2: Was ist bei der Planung der Netzwerktopologie zu beachten?

Workloads existieren oft in mehreren Umgebungen. Dazu gehören mehrere Cloud-Umgebungen (sowohl öffentlich zugänglich als auch privat) und möglicherweise Ihre bestehende Rechenzentrumsinfrastruktur. Die Pläne müssen Netzwerkaspekte wie systeminterne und systemübergreifende Konnektivität, Verwaltung öffentlicher IP-Adressen, Verwaltung privater IP-Adressen und Auflösung von Domainnamen beinhalten.

Workload-Architektur

Ausgangspunkt für eine zuverlässige Workload sind vorab getroffene Designentscheidungen für Software und Infrastruktur. Ihre Auswahl in puncto Architektur wirkt sich in allen Well-Architected-Säulen auf das Verhalten der Workload aus. Zur Gewährleistung von Zuverlässigkeit sind bestimmte Muster zu befolgen.

Bei AWS haben Entwickler von Workloads die Wahl zwischen verschiedenen Sprachen und Technologien. AWS SDKs vereinfachen die Codierung durch die Bereitstellung sprachspezifischer APIs für AWS-Services. Diese SDKs und die Auswahl an Sprachen ermöglichen es Entwicklern, die hier aufgeführten bewährten Methoden zur Gewährleistung von Zuverlässigkeit zu implementieren. Entwickler können sich auch in der [Amazon Builders' Library](#) darüber informieren, wie Software von Amazon erstellt und betrieben wird.

In den folgenden Fragen geht es um Überlegungen zur Zuverlässigkeit.

ZUV 3: Wie entwerfen Sie Ihre Workload-Service-Architektur?

Erstellen Sie hoch skalierbare und zuverlässige Workloads mithilfe einer serviceorientierten Architektur (SOA) oder einer Microservices-Architektur. Eine serviceorientierte Architektur (SOA)

ZUV 3: Wie entwerfen Sie Ihre Workload-Service-Architektur?

hat zum Ziel, Softwarekomponenten über Service-Schnittstellen wiederverwendbar zu machen. Die Microservices-Architektur geht noch weiter, um Komponenten kleiner und einfacher zu machen.

ZUV 4: Wie lassen sich Interaktionen in einem verteilten System so gestalten, dass Ausfälle vermieden werden?

Verteilte Systeme sind auf Kommunikationsnetzwerke angewiesen, um Komponenten wie Server oder Services miteinander zu verbinden. Ihre Workload muss trotz Datenverlust oder Latenz in diesen Netzwerken zuverlässig funktionieren. Die Komponenten des verteilten Systems müssen so funktionieren, dass sie sich nicht negativ auf andere Komponenten oder die Workload auswirken. Diese bewährten Methoden verhindern Ausfälle und verbessern die mittlere Betriebsdauer zwischen Ausfällen (MTBF).

ZUV 5: Wie lassen sich Interaktionen in einem verteilten System so gestalten, dass Ausfälle abgemildert oder bewältigt werden?

Verteilte Systeme nutzen Kommunikationsnetzwerke, um Komponenten (wie Server oder Services) miteinander zu verbinden. Ihre Workload muss trotz Datenverlust oder höherer Latenz in diesen Netzwerken zuverlässig ausgeführt werden. Die Komponenten des verteilten Systems müssen so funktionieren, dass sie sich nicht negativ auf andere Komponenten oder die Workload auswirken. Diese bewährten Methoden sorgen dafür, dass Workloads Belastungen oder Fehlern standhalten, sich schneller davon erholen und die Auswirkungen solcher Beeinträchtigungen abgeschwächt werden. Das Ergebnis ist eine verbesserte mittlere Reparaturzeit (MTTR).

Änderungsmanagement

Änderungen an Ihrer Workload oder der Umgebung müssen vorausgesehen und berücksichtigt werden, um einen zuverlässigen Betrieb der Workload zu erreichen. Zu diesen Änderungen gehören z. B. Bedarfsspitzen sowie interne Änderungen wie Featurebereitstellungen und Sicherheitspatches, die sich auf Ihre Workloads auswirken.

Mit AWS können Sie das Verhalten einer Workload überwachen und die Reaktion auf KPIs automatisieren. Beispielsweise kann die Workload bei einer zunehmenden Zahl von Benutzern zusätzliche Server hinzufügen. Sie können kontrollieren und steuern, welche Benutzer Änderungen an der Workload vornehmen dürfen, und die Historie dieser Änderungen überprüfen.

In den folgenden Fragen geht es um Überlegungen zur Zuverlässigkeit.

ZUV 6: Was ist bei der Überwachung von Workload-Ressourcen zu beachten?

Protokolle und Metriken sind leistungsstarke Tools, mit denen Sie sich einen Überblick über den Zustand Ihrer Workload verschaffen können. Sie können Ihre Workload so konfigurieren, dass Protokolle und Metriken überwacht und Benachrichtigungen gesendet werden, wenn Schwellenwerte überschritten werden oder wichtige Ereignisse auftreten. Dank der Überwachung kann die Workload erkennen, wenn Schwellenwerte für eine niedrige Leistung unterschritten werden oder Ausfälle auftreten, sodass als Reaktion drauf eine automatische Wiederherstellung erfolgen kann.

ZUV 7: Wie lässt sich die Workload so gestalten, dass sie sich an Bedarfsänderungen anpasst?

Eine skalierbare Workload bietet Elastizität, sodass Ressourcen automatisch hinzugefügt oder entfernt werden können, damit sie dem aktuellen Bedarf zu einem bestimmten Zeitpunkt genau entsprechen.

ZUV 8: Wie werden Änderungen implementiert?

Kontrollierte Änderungen sind erforderlich, um neue Funktionen bereitzustellen und um sicherzustellen, dass die Workloads und die Betriebsumgebung bekannte Software ausführen und auf vorhersagbare Weise durch Patches aktualisiert oder ersetzt werden können. Wenn solche Änderungen nicht kontrolliert sind, ist es schwierig, die Auswirkungen der Änderungen vorherzusagen oder Probleme zu lösen, die sich aus ihnen ergeben.

Wenn Sie eine Workload so gestalten, dass Ressourcen als Reaktion auf Bedarfsänderungen automatisch hinzugefügt und entfernt werden, erhöht das nicht nur die Zuverlässigkeit. Vielmehr sorgt diese Vorgehensweise auch dafür, dass geschäftlicher Erfolg nicht zu einer Belastung wird. Bei einer vorhandenen Überwachung wird Ihr Team automatisch benachrichtigt, wenn KPIs von erwarteten

Normen abweichen. Mit dem automatischen Protokollieren von Änderungen an Ihrer Umgebung können Sie auf Aktionen prüfen, die sich möglicherweise auf die Zuverlässigkeit ausgewirkt haben, und diese schnell identifizieren. Mit der Kontrolle und Steuerung des Änderungsmanagements können Sie die Regeln durchsetzen, die für die benötigte Zuverlässigkeit sorgen.

Fehlerverwaltung

In Systemen mit großer Komplexität ist es wahrscheinlich, dass Fehler auftreten. Zur Gewährleistung von Zuverlässigkeit muss Ihre Workload auftretende Fehler erkennen und Maßnahmen ergreifen, um Auswirkungen auf die Verfügbarkeit zu vermeiden. Workloads müssen Ausfälle verkraften sowie Probleme automatisch beheben können.

Mit AWS können Sie automatisch auf überwachte Daten reagieren. Wenn eine bestimmte Kennzahl beispielsweise einen Schwellenwert überschreitet, können Sie eine automatische Maßnahme zur Behebung dieses Problems starten. Statt also zu versuchen, eine fehlerhafte Ressource, die Teil Ihrer Produktionsumgebung ist, zu diagnostizieren und zu reparieren, können Sie sie durch eine neue Ressource ersetzen und die Analyse der fehlerhaften Ressource extern vornehmen. Da Sie in der Cloud temporäre Versionen eines gesamten Systems zu geringen Kosten aufstellen können, können Sie automatisiertes Testen verwenden, um vollständige Wiederherstellungsprozesse zu überprüfen.

In den folgenden Fragen geht es um Überlegungen zur Zuverlässigkeit.

ZUV 9: Was ist bei der Sicherung von Daten zu beachten?

Sichern Sie Daten, Anwendungen und Konfigurationen, um Ihre Anforderungen an Recovery Time Objectives (RTO) und Recovery Point Objectives (RPO) zu erfüllen.

ZUV 10: Wie schützen Sie Ihre Workload mithilfe der Fehlerisolierung?

Die Fehlerisolierung begrenzt die Auswirkungen eines Komponenten- oder Systemausfalls auf eine definierte Grenze. Bei ordnungsgemäßer Isolierung sind Komponenten außerhalb der Grenze nicht vom Ausfall betroffen. Wenn Sie Ihren Workload über mehrere Fehlerisolierungsgrenzen hinweg ausführen, kann er anfälliger für Ausfälle werden.

ZUV 11: Wie lassen sich Workloads so gestalten, dass sie Komponentenausfälle verkraften?

Workloads, für die eine hohe Verfügbarkeit und eine niedrige mittlere Reparaturzeit (MTTR) erforderlich ist, müssen auf Ausfallsicherheit ausgelegt sein.

ZUV 12: Wie lässt sich die Zuverlässigkeit testen?

Nachdem Sie Ihre Workload so konzipiert haben, dass sie den Belastungen der Produktion standhält, sind Tests die einzige Möglichkeit, sie auf die erwartete Funktionalität und Ausfallsicherheit hin zu testen.

ZUV 13: Was ist bei der Planung der Notfallwiederherstellung zu beachten?

Sicherungen und redundante Workload-Komponenten sind der Ausgangspunkt Ihrer Strategie für die Notfallwiederherstellung. [RTO und RPO sind Ihre Ziele](#) für die Wiederherstellung Ihrer Workload. Legen Sie diese entsprechend den geschäftlichen Anforderungen fest. Implementieren Sie eine Strategie, um diese Ziele zu erreichen. Berücksichtigen Sie dabei Standorte und Funktionen von Workload-Ressourcen und -Daten. Die Wahrscheinlichkeit von Unterbrechungen und die Kosten von Wiederherstellungen sind ebenfalls wichtige Faktoren bei der Ermittlung des Unternehmenswerts, den Notfallwiederherstellungen von Workloads bieten.

Sichern Sie Ihre Daten regelmäßig und stellen Sie anhand von Tests der Sicherungsdateien sicher, dass Sie nach logischen und physischen Fehlern eine Wiederherstellung durchführen können. Ein Schlüssel zur Verwaltung von Fehlern ist das regelmäßige und automatisierte Testen von Workloads, um Ausfälle hervorzurufen, und das anschließende Beobachten des Wiederherstellungsverhaltens. Führen Sie diese Tests regelmäßig durch, auch nach größeren Workload-Änderungen. Verfolgen Sie KPIs und das Recovery Time Objective (RTO, Wiederherstellungsdauer) sowie das Recovery Point Objective (RPO, Wiederherstellungszeitpunkt) aktiv, um die Ausfallsicherheit einer Workload (insbesondere unter Fehlertestszenarios) zu bewerten. Die Verfolgung von KPIs unterstützt Sie bei der Identifizierung und Milderung einzelner Fehlerquellen. Hierbei geht es darum, Ihre Prozesse zur Wiederherstellung von Workloads gründlich zu testen, damit Sie darauf vertrauen können, dass Sie alle Daten wiederherstellen und Ihren Service Ihren Kunden unterbrechungsfrei anbieten können – selbst bei länger anhaltenden Problemen. Mit Ihren Wiederherstellungsprozessen sollten Sie sich genauso vertraut machen wie mit Ihren normalen Produktionsprozessen.

Ressourcen

Werfen Sie einen Blick auf die folgenden Ressourcen, um mehr über unsere bewährten Methoden für die Zuverlässigkeit zu erfahren.

Dokumentation

- [AWS-Dokumentation:](#)
- [Globale AWS-Infrastruktur](#)
- [AWS Auto Scaling: Funktionsweise von Skalierungsplänen](#)
- [Was ist AWS Backup?](#)

Whitepaper

- [Säule „Zuverlässigkeit“: AWS Well-Architected](#)
- [Implementieren von Microservices in AWS](#)

Leistungseffizienz

Die Säule der Leistungseffizienz betrifft die Fähigkeit zur effizienten Nutzung von Cloud-Ressourcen, um die Leistungsanforderungen zu erfüllen, sowie die Möglichkeit zur Aufrechterhaltung dieser Effizienz bei Nachfrageänderungen und einer Weiterentwicklung der Technologien.

Die Säule „Leistungsexzellenz“ gibt einen Überblick über konzeptionelle Grundsätze, bewährte Methoden und Fragen. Verbindliche Anleitungen zur Implementierung finden Sie im [Whitepaper „Säule der Leistungseffizienz“](#).

Themen

- [Designprinzipien](#)
- [Definition](#)
- [Bewährte Methoden](#)
- [Ressourcen](#)

Designprinzipien

Es gibt fünf Gestaltungsprinzipien für die Leistungseffizienz in der Cloud:

- **Demokratisieren fortschrittlicher Technologien:** Vereinfachen Sie die Implementierung fortschrittlicher Technologien für Ihr Team, indem Sie komplexe Aufgaben an Ihren Cloud-Anbieter delegieren. Statt Ihr IT-Team aufzufordern, sich näher über das Hosten und Ausführen einer neuen Technologie zu informieren, sollten Sie die Technologie als Service nutzen. Zum Beispiel sind keine SQL Datenbanken, Medientranscodierung und maschinelles Lernen alles Technologien, für die spezielles Fachwissen erforderlich ist. In der Cloud kann Ihr Team diese Technologien als Service nutzen und sich auf die Produktentwicklung konzentrieren, ohne sich um die Bereitstellung und Verwaltung von Ressourcen kümmern zu müssen.
- **Werden Sie innerhalb weniger Minuten global:** Durch die Bereitstellung Ihrer Workloads in mehreren AWS Regionen auf der ganzen Welt können Sie Ihren Kunden bei minimalen Kosten eine geringere Latenz und ein besseres Erlebnis bieten.
- **Nutzung von Serverless-Architekturen:** Aufgrund der in der Cloud verwendeten Serverless-Architekturen brauchen Sie für herkömmliche Datenverarbeitungsaktivitäten keine physischen Server mehr auszuführen und zu verwalten. Serverless-Speicherservices können beispielsweise als statische Websites genutzt werden, wodurch sich Webserver erübrigen. Ihren Code können Sie von Ereignisservices hosten lassen. Auf diese Weise entfällt nicht nur die Verwaltung physischer Server, sondern auch die Transaktionskosten sinken, da verwaltete Services in der Cloud-Umgebung ausgeführt werden.
- **Vermehrtes Experimentieren:** Mit virtuellen und automatisierbaren Ressourcen können Sie schnell unterschiedliche Konfigurationen, Instance- oder Speichertypen miteinander vergleichen.
- **Berücksichtigen des technischen Verständnisses:** Befassen Sie sich mit der Verwendungsweise von Cloud-Services und nutzen Sie stets den Technologieansatz, der für Ihre Workload-Ziele geeignet ist. Berücksichtigen Sie bei der Auswahl des passenden Datenbank- oder Speicherkonzepts beispielsweise die Datenzugriffsmuster.

Definition

Es gibt fünf bewährte Methoden für die Leistungseffizienz in der Cloud:

- Auswahl der Architektur
- Computer und Hardware
- Datenverwaltung
- Netzwerk und Bereitstellung von Inhalten
- Prozess und Kultur

Um eine leistungsstarke Architektur sicherzustellen, empfiehlt sich für deren Entwicklung ein datenbasierter Ansatz. Sammeln Sie zu allen Aspekten der Architektur Daten, angefangen vom allgemeinen Design bis hin zur Auswahl und Konfiguration der Ressourcentypen.

Wenn Sie Ihre Optionen regelmäßig überprüfen, wird bestätigt, dass Sie die Vorteile der sich ständig weiterentwickelnden AWS Cloud nutzen. Durch Überwachung erkennen Sie Abweichungen von der erwarteten Leistung. Zur Leistungssteigerung der Architektur können Sie auch Kompromisse eingehen, beispielsweise durch Komprimierung oder Caching, oder indem Sie hinsichtlich der Konsistenz mehr Toleranz einräumen.

Bewährte Methoden

Themen

- [Auswahl der Architektur](#)
- [Computer und Hardware](#)
- [Datenverwaltung](#)
- [Netzwerk und Bereitstellung von Inhalten](#)
- [Prozess und Kultur](#)

Auswahl der Architektur

Die optimale Lösung für eine bestimmte Workload variiert und Lösungen sind häufig eine Kombination mehrerer Ansätze. Well-Architected-Workloads nutzen mehrere Lösungen und ermöglichen verschiedene Funktionen zur Verbesserung der Leistung.

AWS Ressourcen sind in vielen Typen und Konfigurationen verfügbar, was es einfacher macht, einen Ansatz zu finden, der Ihren Bedürfnissen am besten entspricht. Sie können zudem Optionen nutzen, die sich in Ihrer On-Premises-Infrastruktur nicht ohne Weiteres umsetzen ließen. Ein verwalteter Service wie Amazon DynamoDB bietet beispielsweise eine vollständig verwaltete SQL Nein-Datenbank mit einer Latenz im einstelligen Millisekundenbereich in jeder Größenordnung.

In der folgenden Frage geht es um Überlegungen zur Leistungseffizienz. (Eine Liste der Fragen und bewährten Methoden zur Leistungseffizienz finden Sie im [Anhang](#).).

PERF1: Wie wählen Sie geeignete Cloud-Ressourcen und Architekturmuster für Ihren Workload aus?

Oft sind mehrere Ansätze erforderlich, um eine effektivere Leistung für eine Workload zu erzielen. Well-Architected-Systeme nutzen mehrere Lösungen und Funktionen zur Verbesserung der Leistung.

Computer und Hardware

Die optimale Datenverarbeitungsoption für eine bestimmte Workload kann sich je nach Anwendungsdesign, Nutzungsmustern und Konfigurationseinstellungen unterscheiden. Architekturen können verschiedene Datenverarbeitungsoptionen für verschiedene Komponenten verwenden und verschiedene Funktionen zur Verbesserung der Leistung bieten. Die Wahl der falschen Datenverarbeitungslösung für eine Architektur kann die Leistungseffizienz schmälern.

In AWS, Compute ist in drei Formen verfügbar: Instanzen, Container und Funktionen:

- Instanzen sind virtualisierte Server, die es Ihnen ermöglichen, ihre Funktionen mit einer Taste oder einem API Aufruf zu ändern. Da Ressourcenentscheidungen in der Cloud flexibel sind, können Sie mit verschiedenen Servertypen experimentieren. Bei AWS diesen virtuellen Serverinstanzen gibt es verschiedene Familien und Größen und sie bieten eine Vielzahl von Funktionen, darunter Solid-State-Laufwerke (SSDs) und Grafikprozessoren (GPUs).
- Container sind eine Methode der Betriebssystemvirtualisierung, mit der Sie eine Anwendung und ihre Abhängigkeiten in ressourcenisolierten Prozessen ausführen können. AWS Fargate ist serverloses Computing für Container oder Amazon EC2 kann verwendet werden, wenn Sie Kontrolle über die Installation, Konfiguration und Verwaltung Ihrer Rechenumgebung benötigen. Sie können auch aus mehreren Plattformen für die Container-Orchestrierung wählen: Amazon Elastic Container Service (ECS) oder Amazon Elastic Kubernetes Service (EKS).
- Funktionen abstrahieren die Ausführungsumgebung vom anzuwendenden Code. AWS Lambda ermöglicht es Ihnen beispielsweise, Code auszuführen, ohne eine Instance auszuführen.

In der folgenden Frage geht es um Überlegungen zur Leistungseffizienz.

PERF2: Wie wählen und verwenden Sie Rechenressourcen in Ihrem Workload?

Welche Datenverarbeitungslösung für eine Workload effizienter ist, ist vom Anwendungsdesign sowie von Nutzungsmustern und Konfigurationseinstellungen abhängig. Architekturen können unterschiedliche Datenverarbeitungslösungen für verschiedene Komponenten verwenden und unterschiedliche Funktionen zur Leistungsverbesserung bieten. Die Wahl der falschen Datenverarbeitungslösung für eine Architektur kann die Leistungseffizienz schmälern.

Datenverwaltung

Die optimale Datenverwaltungslösung für ein bestimmtes System hängt von der Art des Datentyps (Block, Datei oder Objekt), den Zugriffsmustern (zufällig oder sequentiell), dem erforderlichen Durchsatz, der Häufigkeit des Zugriffs (online, offline, archiviert), der Aktualisierungshäufigkeit (WORMdynamisch) sowie von Verfügbarkeits- und Haltbarkeitsbeschränkungen ab. Well-Architected-Workloads verwenden zweckgebundene Datenspeicher, die verschiedene Features zur Verbesserung der Leistung ermöglichen.

AWS In ist Speicher in drei Formen verfügbar: Objekt, Block und Datei:

- Objektspeicher bietet eine skalierbare, robuste Plattform, damit Daten überall im Internet zugänglich sind. Das gilt für benutzergenerierte Inhalte, aktive Archive, Serverless-Datenverarbeitung, Big Data-Speicher oder die Sicherung und Wiederherstellung. Amazon Simple Storage Service (Amazon S3) ist ein Objektspeicherservice, der branchenführende Skalierbarkeit, Datenverfügbarkeit, Sicherheit und Leistung bietet. Amazon S3 ist auf eine Verfügbarkeit von 99,999999999 % ausgelegt und speichert Daten für Millionen von Anwendungen für Unternehmen weltweit.
- Blockspeicher bietet hochverfügbaren, konsistenten Blockspeicher mit niedriger Latenz für jeden virtuellen Host und entspricht direkt angeschlossenem Speicher (DAS) oder einem Storage Area Network (SAN). Amazon Elastic Block Store (AmazonEBS) wurde für Workloads entwickelt, die persistenten Speicher benötigen, auf den EC2 Instances zugreifen können, sodass Sie Anwendungen mit der richtigen Speicherkapazität, Leistung und den richtigen Kosten optimieren können.
- Dateispeicher bietet auf mehreren Systemen Zugriff auf ein gemeinsam genutztes Dateisystem. Dateispeicherlösungen wie Amazon Elastic File System (AmazonEFS) eignen sich ideal für Anwendungsfälle wie große Inhaltsrepositorys, Entwicklungsumgebungen, Medienspeicher oder Benutzerverzeichnisse. Amazon FSx macht es effizient und kostengünstig, beliebte Dateisysteme

zu starten und auszuführen, sodass Sie die umfangreichen Funktionen und die schnelle Leistung weit verbreiteter Open-Source-Datenbanken und kommerziell lizenzierter Datenbanken nutzen können.

In der folgenden Frage geht es um Überlegungen zur Leistungseffizienz.

PERF3: Wie speichern, verwalten und greifen Sie auf Daten in Ihrem Workload zu?

Die effizientere Speicherlösung für ein System hängt von der Art des Zugriffsvorgangs (Block, Datei oder Objekt), den Zugriffsmustern (zufällig oder sequentiell), dem erforderlichen Durchsatz, der Zugriffshäufigkeit (online, offline, Archivierung), der Aktualisierungshäufigkeit (WORMdynamisch) sowie von Verfügbarkeits- und Haltbarkeitsbeschränkungen ab. Gut geplante Systeme nutzen mehrere Speicherlösungen und bieten unterschiedliche Möglichkeiten zur Leistungsoptimierung und effizienten Ressourcennutzung.

Netzwerk und Bereitstellung von Inhalten

Welche Netzwerklösung für eine Workload optimal ist, richtet sich nach der Latenz, dem erforderlichen Durchsatz, dem Jitter und der Bandbreite. Die Standortoptionen sind von den physischen Einschränkungen abhängig, z. B. von Benutzer- oder On-Premises-Ressourcen. Diese Einschränkungen können durch Edge-Standorte oder die Ressourcenplatzierung wettgemacht werden.

On AWS ist das Netzwerk virtualisiert und in einer Reihe verschiedener Typen und Konfigurationen verfügbar. Dies macht es einfacher, Ihren Netzwerkanforderungen gerecht zu werden. AWS bietet Produktfunktionen (z. B. Enhanced Networking, Amazon EC2 Networking Optimized Instances, Amazon S3 Transfer Acceleration und Dynamic Amazon CloudFront) zur Optimierung des Netzwerkverkehrs. AWS bietet auch Netzwerkfunktionen (z. B. Amazon Route 53-Latenz-Routing, VPC Amazon-Endpunkte und AWS Global Accelerator) AWS Direct Connect, um Netzwerkdistanz oder Jitter zu reduzieren.

In der folgenden Frage geht es um Überlegungen zur Leistungseffizienz.

PERF4: Wie wählen und konfigurieren Sie Netzwerkressourcen in Ihrem Workload?

In dieser Frage werden Anleitungen und bewährte Methoden für die Entwicklung, Konfiguration und den Betrieb effizienter Netzwerk und Inhaltsbereitstellungslösungen in der Cloud bereitgestellt.

Prozess und Kultur

Bei der Architektur von Workloads gibt es Prinzipien und Methoden, die Sie übernehmen können, um effiziente und leistungsstarke Cloud-Workloads besser zu betreiben. Um eine Kultur zu schaffen, die die Leistungseffizienz von Cloud-Workloads fördert, sollten Sie diese Schlüsselprinzipien und -methoden berücksichtigen.

Beachten Sie beim Aufbau dieser Kultur die folgenden Schlüsselprinzipien:

- **Infrastruktur als Code:** Definieren Sie Ihre Infrastruktur als Code mithilfe von Ansätzen wie AWS CloudFormation Vorlagen. Mit Vorlagen können Sie Ihre Infrastruktur zusammen mit Ihrem Anwendungscode und Ihren Konfigurationen per Quellcodeüberwachung verwalten. Dies ermöglicht es Ihnen, dieselben Methoden wie bei der Softwareentwicklung auch auf Ihre Infrastruktur anzuwenden, um von einer schnellen Iteration zu profitieren.
- **Bereitstellungspipeline:** Nutzen Sie zur Bereitstellung der Infrastruktur eine CI/CD-Pipeline (Continuous Integration/Continuous Deployment) wie etwa ein Quellcode-Repository, Build-Systeme sowie automatisierte Bereitstellungs- und Testverfahren. Dies lässt eine reproduzierbare, konsistente und kostengünstige Iteration zu.
- **Klar definierte Metriken:** Richten Sie Metriken ein und überwachen Sie sie, um wichtige Leistungsindikatoren zu erfassen (KPIs). Wir empfehlen die Verwendung technischer und geschäftlicher Metriken. Bei Websites oder mobilen Apps sind Erfassung time-to-first-byte oder Rendern die wichtigsten Kennzahlen. Zu den weiteren allgemein anwendbaren Metriken zählen die Thread-Anzahl, die Garbage Collection-Rate sowie Wartezustände. Anhand von geschäftlichen Metriken wie den aggregierten kumulativen Kosten pro Anforderung können Sie Möglichkeiten zur Kostensenkung ermitteln. Erwägen Sie sorgfältig, wie Metriken interpretiert werden sollen. Sie können beispielsweise anstelle von Durchschnittswerten Maximalwerte oder das 99. Perzentil wählen.
- **Automatische Leistungstests:** Sorgen Sie im Rahmen der Bereitstellung dafür, dass nach dem erfolgreichen Absolvieren der schnelleren Ausführungstests automatisch Leistungstests initiiert werden. Durch die Automatisierung sollte eine neue Umgebung mit entsprechenden Anfangsbedingungen, z. B. Testdaten, entstehen, in der anschließend einige Benchmark- und Lasttests ausgeführt werden. Die Ergebnisse dieser Tests sollten mit dem Build in Verbindung gebracht werden, um Leistungsänderungen verfolgen zu können. Für langwierige Tests können Sie diesen Teil der Pipeline gegenüber dem restlichen Build asynchron ausführen. Alternativ können Sie über Nacht Leistungstests mit Amazon EC2 Spot-Instances durchführen.
- **Lastgenerierung:** Erstellen Sie eine Reihe von Testskripts zum Replizieren synthetischer oder vorab aufgezeichneter Benutzerreisen. Diese Skripts sollten idempotent und nicht gekoppelt

sein. Um gültige Ergebnisse zu erzielen, sind möglicherweise zusätzliche vorbereitende Skripts erforderlich. Die Testskripts sollten das Nutzungsverhalten in der Produktion möglichst authentisch replizieren. Sie können Software oder software-as-a-service (SaaS-) Lösungen verwenden, um die Last zu generieren. Erwägen Sie die Verwendung von [AWS Marketplace](#)-Lösungen und [Spot Instances](#). Dies können kostengünstige Ansätze zum Generieren der Last sein.

- **Leistungstransparenz:** Wichtige Metriken sollten für das ganze Team sichtbar sein. Dies gilt insbesondere für die Metriken der einzelnen Build-Versionen. Damit lassen sich wichtige positive oder negative Trends erkennen. Wichtig sind auch Metriken zur Anzahl der Fehler oder Ausnahmen, um sicherzustellen, dass das System funktioniert.
- **Visualisierung:** Nutzen Sie Visualisierungstechniken, mit denen Leistungsprobleme, Hotspots, Wartezustände oder niedrige Auslastungen klar aufgezeigt werden. Zeigen Sie Leistungsmetriken in Architekturdiagrammen an. Aufrufgrafiken oder Code können die Problemerkennung beschleunigen.
- **Regelmäßiger Prüfungsprozess:** Wenn Architekturen eine schlechte Leistung aufweisen, liegt dies normalerweise daran, dass ein Prozess zur Überprüfung der Leistung fehlt oder fehlerhaft ist. Falls Sie derartige Probleme mit Ihrer Architektur haben, können Sie jederzeit ein Leistungsprüfverfahren implementieren und somit iterative Verbesserungen fördern.
- **Fortlaufende Optimierung:** Schaffen Sie eine Kultur fortlaufender Optimierung der Leistungseffizienz Ihrer Cloud-Workload.

In der folgenden Frage geht es um Überlegungen zur Leistungseffizienz.

PERF5: Welchen Prozess verwenden Sie, um die Leistungseffizienz Ihres Workloads zu erhöhen?

Bei der Architektur von Workloads gibt es Prinzipien und Methoden, die Sie übernehmen können, um effiziente und leistungsstarke Cloud-Workloads besser zu betreiben. Um eine Kultur zu schaffen, die die Leistungseffizienz von Cloud-Workloads fördert, sollten Sie diese Schlüsselprinzipien und -methoden berücksichtigen.

Ressourcen

Weitere Informationen zu bewährten Methoden für die Leistungseffizienz finden Sie in den folgenden Ressourcen.

Dokumentation

- [Leistungsoptimierung mit Amazon S3](#)
- [Leistung von Amazon EBS Volume](#)

Whitepaper

- [Säule der Leistungseffizienz](#)

Video

- [AWS re:Invent 2019: EC2 Amazon-Stiftungen \(-R2\) CMP211](#)
- [AWS re:Invent 2019: Sitzung für Führungskräfte: Der Zustand der Speicherbranche \(01-L\) STG2](#)
- [AWS re:Invent 2019: Sitzung für Führungskräfte: speziell entwickelte Datenbanken \(09-L AWS \) DAT2](#)
- [AWS re:Invent 2019: Konnektivität zu und hybride Netzwerkarchitekturen \(-R1\) AWSAWS NET317](#)
- [AWS re:Invent 2019: Unterstützung für Amazon der nächsten GenerationEC2: Tiefer Einblick in das Nitro-System \(03-R2\) CMP3](#)
- [AWS re:Invent 2019: Skalierung bis zu Ihren ersten 10 Millionen Benutzern \(-R\) ARC211](#)

Kostenoptimierung

Die Säule „Kostenoptimierung“ umfasst die Fähigkeit, Systeme so auszuführen, dass sie geschäftlichen Wert bei geringstmöglichen Kosten liefern.

Die Säule „Kostenoptimierung“ bietet einen Überblick über Designprinzipien, bewährte Methoden und Fragen. Verbindliche Anleitungen zur Implementierung finden Sie im [Whitepaper zur Säule „Kostenoptimierung“](#).

Themen

- [Designprinzipien](#)
- [Definition](#)
- [Best Practices](#)
- [Ressourcen](#)

Designprinzipien

Es gibt fünf Designprinzipien für die Kostenoptimierung in der Cloud:

- Cloud Financial Management implementieren: Um finanziellen Erfolg zu haben und die Wertschöpfung in der Cloud zu beschleunigen, müssen Sie in Cloud-Finanzmanagement und Kostenoptimierung investieren. Ihre Organisation muss Zeit und Ressourcen aufwenden, um Know-how in diesem neuen Bereich des Technologie- und Nutzungsvermanagements aufzubauen. Wie bei Ihren Funktionen zur Sicherheit oder der operativen Exzellenz müssen Sie Fähigkeiten durch Wissen, Programme, Ressourcen und Prozesse entwickeln, damit Sie zu einer kosteneffizienten Organisation werden können.
- Verbrauchsmodell einführen: Zahlen Sie nur für die benötigten Datenverarbeitungsressourcen, und erhöhen oder verringern Sie die Nutzung auf Basis Ihrer Geschäftsanforderungen und nicht durch aufwändige Prognosen. Entwicklungs- und Testumgebungen werden in einer normalen Arbeitswoche beispielsweise nur acht Stunden pro Tag benötigt. Sie können diese Ressourcen anhalten, wenn sie nicht verwendet werden und damit potenzielle Einsparungen von 75 % (40 Stunden vs. 168 Stunden) erzielen.
- Gesamteffizienz messen: Messen Sie die geschäftliche Leistung der Workload und die mit der Bereitstellung verknüpften Kosten. Verwenden Sie diese Kennzahlen, um die Gewinne zu ermitteln, die Sie durch die Erhöhung der Leistung und die Reduzierung der Kosten erzielen.
- Kein Geld mehr für undifferenzierte, aufwendige Arbeiten ausgeben: AWS erledigt die aufwendigsten Arbeiten im Rechenzentrum bezüglich Server-Racks, -Stacks und -Stromversorgung. Außerdem entfällt der betriebliche Aufwand für die Verwaltung von Betriebssystemen und Anwendungen mit verwalteten Services. So können Sie sich auf Ihre Kunden und Geschäftsprojekte anstatt auf die IT-Infrastruktur konzentrieren.
- Ausgaben analysieren und zuordnen: Mit der Cloud ist es einfacher, die Nutzung und die Kosten von Systemen genau zu ermitteln und auf Basis dieser Daten eine transparente Zuordnung der IT-Kosten auf einzelne Workload-Besitzer durchzuführen. Das hilft Ihnen bei der Ermittlung der Umsatzrendite (ROI) und Workload-Besitzer haben die Möglichkeit, ihre Ressourcen zu optimieren und die Kosten zu reduzieren.

Definition

Es gibt fünf Bereiche für bewährte Methoden für die Kostenoptimierung in der Cloud:

- Cloud Financial Management betreiben

- Ausgabenerkennung und Nutzungsbewusstsein
- Kosteneffiziente Ressourcen
- Verwaltung von Nachfrage und Bereitstellung von Ressourcen
- Optimierung im Laufe der Zeit

Wie bei den anderen Säulen innerhalb des Well-Architected Framework sind Kompromisse unvermeidbar, so müssen Sie beispielsweise entscheiden, ob Sie die Markteinführungsgeschwindigkeit oder die Kosten optimieren möchten. In manchen Fällen ist es sinnvoll, die Priorität auf Geschwindigkeit zu legen, z. B. verbunden mit einer raschen Markteinführung, der Bereitstellung neuer Features oder einer simplen Fristerfüllung, statt im Vorfeld in Kostenoptimierung zu investieren. Konzeptionelle Entscheidungen werden gelegentlich durch Eile statt auf Basis von Daten getroffen, und man ist immer der Versuchung ausgesetzt, einem potenziellen Szenario zu viel Bedeutung beizumessen, statt Zeit in die Bestimmung der kostengünstigsten Bereitstellung zu investieren. Dies führt häufig übermäßigen und mangelhaft optimierten Bereitstellungen. Es ist jedoch die richtige Wahl, wenn Sie Ressourcen aus Ihrer On-Premises-Umgebung per Lift and Shift in die Cloud verlagern und die Optimierung anschließend durchführen möchten. Wenn Sie vorab genügend Arbeit in eine Strategie zur Kostenoptimierung investieren, können Sie die wirtschaftlichen Vorteile der Cloud schneller nutzen, indem Sie eine konsistente Einhaltung bewährter Methoden sicherstellen und Überbereitstellungen vermeiden. In den folgenden Abschnitten finden Sie Techniken und bewährte Methoden sowohl für die erste als auch die fortlaufende Implementierung von Cloud-Finanzmanagement und Kostenoptimierung für Ihre Workloads.

Best Practices

Themen

- [Cloud Financial Management betreiben](#)
- [Ausgabenerkennung und Nutzungsbewusstsein](#)
- [Kosteneffiziente Ressourcen](#)
- [Verwaltung von Nachfrage und Bereitstellung von Ressourcen](#)
- [Optimierung im Laufe der Zeit](#)

Cloud Financial Management betreiben

Mit der Einführung der Cloud können Technologieteams dank verkürzter Genehmigungs-, Beschaffungs- und Infrastrukturbereitstellungszyklen schneller innovieren. Ein neuer Ansatz für das Finanzmanagement in der Cloud ist erforderlich, um geschäftlichen Nutzen und finanziellen Erfolg zu erzielen. Dieser Ansatz ist das Cloud-Finanzmanagement. Es baut Funktionen in Ihrer gesamten Organisation auf, indem organisationsweit Wissensaufbau, Programme, Ressourcen und Prozesse implementiert werden.

Viele Organisationen bestehen aus vielen verschiedenen Einheiten mit unterschiedlichen Prioritäten. Durch die Fähigkeit, Ihre Organisation an mehreren vereinbarten Finanzzielen auszurichten und ihr die Mechanismen zur Erreichung der Ziele bereitzustellen, wird die Effizienz der Organisation gesteigert. Eine leistungsfähige Organisation innoviert und entwickelt schneller, ist agiler und passt sich einfacher an beliebige interne oder externe Faktoren an.

In AWS können Sie Cost Explorer und optional Amazon Athena und Amazon QuickSight mit dem Kosten- und Nutzungsbericht (CUR) verwenden. So können Sie Kosten- und Nutzungsbewusstsein in Ihrer gesamten Organisation schaffen. AWS Budgets bietet proaktive Benachrichtigungen zu Kosten und Nutzung. Die AWS-Blogs bieten Informationen zu neuen Services und Features, damit Sie immer über neue Serviceversionen auf dem Laufenden sind.

In der folgenden Frage geht es um Überlegungen zur Kostenoptimierung. (Eine Liste der Fragen und bewährten Methoden zur Kostenoptimierung finden Sie im [Anhang](#).)

KOSTEN 1: Wie implementieren Sie das Cloud-Finanzmanagement?

Die Implementierung von Cloud Financial Management hilft Organisationen, geschäftliche Mehrwerte und einen wirtschaftlichen Erfolg zu erzielen, während sie die Kosten und die Nutzung optimieren und auf AWS skalieren.

Beim Aufbau einer Kostenoptimierungsfunktion sollten Sie Teammitglieder einsetzen und das Team um Experten für CFM und Kostenoptimierung ergänzen. Bestehende Teammitglieder wissen, wie die Organisation derzeit funktioniert und Verbesserungen schnell implementiert werden können. Erwägen Sie auch, Personen mit ergänzenden oder speziellen Kenntnissen, wie im Bereich Analytik oder Projektmanagement, mit einzubinden.

Wenn Sie in Ihrer Organisation ein Kostenbewusstsein implementieren, verbessern Sie vorhandene Programme oder bauen auf diesen auf. Es geht viel schneller, bestehende Prozesse und Programme zu ergänzen, als sie neu zu erstellen. So werden die Ergebnisse viel schneller erreicht.

Ausgabenerkennung und Nutzungsbewusstsein

Die erhöhte Flexibilität und Agilität der Cloud fördert Innovationen und schnelle Entwicklungen und Bereitstellungen. Diese Merkmale eliminieren die manuellen Prozesse und den Zeitaufwand für die Bereitstellung einer On-Premises-Infrastruktur, einschließlich der Identifizierung von Hardware-Spezifikationen, dem Verhandeln von Preisen, der Verwaltung von Bestellungen, der Planung von Lieferungen und schließlich der Bereitstellung der Ressourcen. Die einfache Nutzung und die nahezu unbegrenzte On-Demand-Verfügbarkeit macht neue Wege erforderlich, über Ausgaben nachzudenken.

Viele Unternehmen bestehen aus einer Vielzahl von Systemen, die von unterschiedlichen Teams betrieben werden. Die Möglichkeit, die Ressourcenkosten der jeweiligen Organisation oder den jeweiligen Produkteigentümer zuzuordnen, fördert ein effizientes Nutzungsverhalten und hilft, Verschwendung von Ressourcen einzudämmen. Mit einer präzisen Kostenzuordnung wissen Sie, welche Produkte wirklich profitabel sind, und können fundiertere Entscheidungen in Bezug auf die Budgetaufteilung treffen.

In AWS erstellen Sie mit AWS Organizations oder AWS Control Tower eine Kontostruktur, die eine Trennung ermöglicht und Sie bei der Zuordnung Ihrer Kosten und Nutzung unterstützt. Sie können auch das Ressourcen-Tagging verwenden, um Geschäfts- und Organisationsinformationen auf Ihre Nutzung und Kosten anzuwenden. Verwenden Sie AWS Cost Explorer, um Einblicke in Ihre Kosten und Nutzung zu erhalten, oder erstellen Sie benutzerdefinierte Dashboards und Analytik mit Amazon Athena und Amazon QuickSight. Die Kontrolle Ihrer Kosten und Nutzung erfolgt durch Benachrichtigungen über AWS Budgets sowie Kontrollen mithilfe von AWS Identity and Access Management (IAM) und Service Quotas.

In den folgenden Fragen geht es um Überlegungen zur Kostenoptimierung.

KOSTEN 2: Wie können Sie die Nutzung steuern?

Legen Sie Richtlinien und Mechanismen fest, um zu überprüfen, ob angemessene Kosten anfallen und die Ziele erreicht werden. Durch den Einsatz eines Kontrollsystems können Sie Innovationen vorantreiben, ohne das Budget zu überschreiten.

KOSTEN 3: Wie können Sie die Nutzung und Kosten überwachen?

Definieren Sie Richtlinien und Verfahren, um Ihre Kosten überwachen und richtig zuordnen zu können. So können Sie die Kosteneffizienz einer Workload messen und verbessern.

KOSTEN 4: Wie können Sie Ressourcen außer Betrieb nehmen?

Implementieren Sie vom Beginn bis zum Abschluss eines Projekts eine Änderungskontrolle und Ressourcenverwaltung. Auf diese Weise können Sie ungenutzte Ressourcen herunterfahren oder beenden, um Verschwendungen zu minimieren.

Sie können Tags für die Kostenzuordnung verwenden, um Ihre Nutzung und Kosten in AWS zu kategorisieren und zu verfolgen. Wenn Sie Tags auf Ihre AWS-Ressourcen anwenden (z. B. EC2-Instances oder S3-Buckets), generiert AWS einen Kosten- und Nutzungsbericht mit Ihrer Nutzung und Ihren Tags. Sie können Tags anwenden, die für Organisationskategorien stehen (z. B. Kostenstellen, Workload-Namen oder Besitzer), um Ihre Kosten verschiedenen Services zuzuordnen.

Achten Sie darauf, dass Sie den richtigen Detail- und Granularitätsgrad für die Kosten- und Nutzungsberichterstattung und -überwachung verwenden. Um allgemeine Erkenntnisse zu gewinnen und Trends zu erkennen, verwenden Sie die tägliche Granularität mit AWS Cost Explorer. Für tiefergehende Analysen und Prüfungen verwenden Sie die stündliche Granularität in AWS Cost Explorer oder Amazon Athena und Amazon Quick sowie den Kosten- und Nutzungsbericht (CUR) mit stündlicher Granularität.

Durch die Kombination von mit Tags gekennzeichneten Ressourcen und Entitätslebenszyklus-Tracking (Mitarbeiter, Projekte) können Sie verwaiste Ressourcen oder Projekte identifizieren, die für das Unternehmen keinen Wert mehr generieren und außer Betrieb genommen werden sollten. Sie können Abrechnungsbenachrichtigungen einrichten, um sich über prognostizierte Budgetüberschreitungen zu informieren.

Kosteneffiziente Ressourcen

Die Verwendung geeigneter Instances und Ressourcen für Ihre Workload ist für Kosteneinsparungen von entscheidender Bedeutung. Die Ausführung eines Berichtsprozesses kann auf kleineren Servern beispielsweise bis zu fünf Stunden dauern, auf einem doppelt so teuren großen Server jedoch

lediglich eine Stunde. Auf beiden Servern erhalten Sie dasselbe Ergebnis, der kleinere Server generiert über den Ausführungszeitraum jedoch höhere Kosten.

Architektonisch gute Workloads verwenden die kostengünstigsten Ressourcen; dieses Verhalten kann eine signifikante und positive wirtschaftliche Auswirkung haben. Sie haben außerdem die Möglichkeit, verwaltete Services für die Kostenreduzierung zu verwenden. So können Sie für die E-Mail-Zustellung beispielsweise einen Service nutzen, bei dem die Kosten nach der Anzahl der versendeten Nachrichten berechnet werden, statt Server für diese Aufgabe bereithalten zu müssen.

AWS bietet eine Vielzahl flexibler und kosteneffektiver Preisooptionen für den Erwerb von Instances von Amazon EC2 und anderen Services auf eine Weise, die Ihre Anforderungen effizienter erfüllt. Mit On-Demand-Instances können Sie die genutzte Datenverarbeitungskapazität auf Stundenbasis und ohne Mindestverpflichtungen bezahlen. Savings Plans und Reserved Instances bieten Einsparungen von bis zu 75 % gegenüber On-Demand-Preisen. Mit Spot Instances können Sie ungenutzte Amazon-EC2-Kapazität nutzen und von Einsparungen von bis zu 90 % im Vergleich zum On-Demand-Preis profitieren. Spot Instances eignen sich, wenn das System eine Flotte von Servern toleriert, bei der einzelne Server dynamisch aktiviert und deaktiviert werden können, wie z. B. bei zustandslosen Webservern, bei der Stapelverarbeitung oder bei der Nutzung von HPC und Big Data.

Auch mit der Auswahl geeigneter Services ist es möglich, Nutzung und Kosten zu reduzieren. So können Sie beispielsweise CloudFront nutzen, um das Datenübertragungsvolumen zu reduzieren, oder Kosten vollständig eliminieren, z. B. mit Amazon Aurora on RDS, mit dem Sie kostspielige Datenbanklizenzierungskosten vermeiden können.

In den folgenden Fragen geht es um Überlegungen zur Kostenoptimierung.

KOSTEN 5: Wie können Sie die Kosten bei der Auswahl von Services einschätzen?

Bei Amazon EC2, Amazon EBS und Amazon S3 handelt es sich um AWS-Services, die als einzelne Bausteine angeboten werden. Verwaltete Services, etwa Amazon RDS und Amazon DynamoDB, sind AWS-Services auf einer höheren Ebene oder Anwendungsebene. Wenn Sie sich für die richtigen Bausteine und verwalteten Services entscheiden, können Sie die Kosten dieser Workload optimieren. Durch die Nutzung von verwalteten Services können Sie einen Großteil Ihres administrativen und betrieblichen Overheads reduzieren oder beseitigen und damit Kapazität für anwendungs- und geschäftsbezogene Aktivitäten gewinnen.

KOSTEN 6: Wie können Sie bei der Auswahl des Ressourcentyps, -umfangs und der Anzahl der Ressourcen Kostenziele erfüllen?

Stellen Sie sicher, dass Sie den geeigneten Ressourcenumfang und die Anzahl der Ressourcen für die jeweilige Aufgabe auswählen. Durch die Auswahl des kostengünstigsten Typs, Umfangs und der kostengünstigsten Anzahl minimieren Sie die Verschwendung von Ressourcen.

KOSTEN 7: Wie können Sie Kosten mithilfe von Preismodellen senken?

Verwenden Sie das Preismodell, das sich für Ihre Ressourcen am besten eignet. So halten Sie die Ausgaben möglichst niedrig.

KOSTEN 8: Wie können Sie die Kosten für Datenübertragungen planen?

Damit Sie architekturbezogene Entscheidungen zur Kostenminimierung treffen können, müssen Sie unbedingt die Datenübertragungskosten einplanen und überwachen. Eine geringfügige, aber effektive Änderung an der Architektur kann Ihre Betriebskosten über einen längeren Zeitraum hinweg erheblich senken.

Durch das Einkalkulieren der Kosten während der Serviceauswahl und die Verwendung von Tools wie Cost Explorer und AWS Trusted Advisor zur regelmäßigen Überprüfung Ihrer AWS-Nutzung können Sie Ihre Nutzung aktiv überwachen und Ihre Bereitstellungen entsprechend anpassen.

Verwaltung von Nachfrage und Bereitstellung von Ressourcen

Wenn Sie in die Cloud wechseln, zahlen Sie nur für die genutzten Ressourcen. Sie können Ressourcen so bereitstellen, dass sie dem Workload-Bedarf zum jeweiligen Zeitpunkt entsprechen. Dadurch werden kostspielige Überbereitstellungen überflüssig. Sie können den Bedarf auch anpassen, indem Sie eine Drosselung, einen Puffer oder eine Warteschlange verwenden, um den Bedarf zu glätten und ihn mit weniger Ressourcen zu erfüllen, was zu niedrigeren Kosten führt. Außerdem können Sie ihn mit einem Batch-Service zu einem späteren Zeitpunkt verarbeiten.

In AWS können Sie Ressourcen automatisch so bereitstellen, dass sie den Workload-Bedarf erfüllen. Durch Auto Scaling mit bedarfs- oder zeitbasiertem Ansatz können Sie Ressourcen nach Bedarf hinzufügen und entfernen. Wenn Sie in der Lage sind, Bedarfsänderungen zu antizipieren, können

Sie mehr Kosten einsparen und zugleich sicherstellen, dass Ihre Ressourcen Ihren Workload-Anforderungen entsprechen. Sie können Amazon API Gateway verwenden, um eine Drosselung zu implementieren, oder Amazon SQS einsetzen, um eine Warteschlange für Ihre Workload zu implementieren. Mit beiden können Sie den Bedarf für Ihre Workload-Komponenten anpassen.

In der folgenden Frage geht es um Überlegungen zur Kostenoptimierung.

KOSTEN 9: Wie verwalten Sie die Nachfrage und stellen Ressourcen bereit?

Stellen Sie bei einer Workload mit ausgewogenen Ausgaben und Leistungen sicher, dass alles, wofür Sie bezahlen, genutzt wird, und vermeiden Sie eine erhebliche Unterauslastung der Instances. Eine verschobene Auslastungsmetrik in einer der Richtungen wirkt sich nachteilig auf Ihre Organisation aus, entweder im Hinblick auf die Betriebskosten (verschlechterte Leistung aufgrund von Überbelegung) oder auf die verschwendeten AWS-Ausgaben (aufgrund von Überversorgung).

Wenn Sie planen, dass Ressourcen für Bedarf und Bereitstellung geändert werden können, denken Sie auch an die Nutzungsmuster, die Zeit für die Bereitstellung neuer Ressourcen und die Vorhersehbarkeit des Bedarfsmusters. Stellen Sie beim Verwalten des Bedarfs sicher, dass Ihre Warteschlange oder Ihr Puffer korrekt dimensioniert ist und Sie in der erforderlichen Zeit auf den Workload-Bedarf reagieren.

Optimierung im Laufe der Zeit

Im Zuge der Veröffentlichung neuer Services und Features durch AWS empfiehlt es sich, dass Sie Ihre bestehenden Entscheidungen zur Architektur überdenken, um sicherzustellen, dass diese weiterhin so kostengünstig wie möglich sind. Wenn sich Ihre Anforderungen ändern, zögern Sie nicht, und nehmen Sie Ressourcen, ganze Services und Systeme, die Sie nicht mehr benötigen, außer Betrieb.

Durch die Implementierung neuer Features oder Ressourcentypen können Sie Ihre Workload inkrementell optimieren und gleichzeitig den Aufwand für die Implementierung der Änderung minimieren. Dadurch wird die Effizienz im Laufe der Zeit kontinuierlich verbessert und sichergestellt, dass Sie stets die aktuellste Technologie nutzen, um die Betriebskosten zu senken. Sie können mit neuen Services auch Komponenten der Workload ersetzen oder ihm neue Komponenten hinzufügen. Dies kann zu erheblichen Effizienzsteigerungen führen. Daher ist es wichtig, Ihre Workload regelmäßig zu überprüfen und neue Services und Features zu implementieren.

In den folgenden Fragen geht es um Überlegungen zur Kostenoptimierung.

KOSTEN 10: Wie können Sie neue Services bewerten?

Im Zuge der Veröffentlichung neuer Services und Features durch AWS empfiehlt es sich, dass Sie Ihre bestehenden Entscheidungen zur Architektur überdenken, um sicherzustellen, dass diese weiterhin so kostengünstig wie möglich sind.

Wenn Sie Ihre Bereitstellungen regelmäßig überprüfen, sollten Sie auch bewerten, wie Sie mit neueren Services möglicherweise Geld sparen können. Mit Amazon Aurora in Amazon RDS können Sie beispielsweise die Kosten für relationale Datenbanken reduzieren. Wenn Sie serverlose Technologie wie Lambda verwenden, müssen Sie Instances nicht mehr betreiben und verwalten, um Code auszuführen.

KOSTEN 11: Wie bewerten Sie die Kosten des Aufwands?

Bewerten die Kosten für den Betrieb in der Cloud, prüfen und automatisieren Sie Ihre zeitaufwändigen Cloud-Vorgänge, um durch Einführung von entsprechenden AWS-Services, Produkten von Drittanbietern oder benutzerdefinierten Tools den manuellen Arbeitsaufwand und die Personalkosten zu verringern.

Ressourcen

Weitere Informationen zu bewährten Methoden für die Kostenoptimierung finden Sie in den folgenden Ressourcen.

Dokumentation

- [AWS-Dokumentation](#)

Whitepaper

- [Säule der Kostenoptimierung](#)

Nachhaltigkeit

Bei der Säule „Nachhaltigkeit“ geht es um Auswirkungen auf die Umwelt, insbesondere um Energieverbrauch und -effizienz, da diese wichtige Faktoren für Architekten sind, die ihre direkten Aktionen zur Reduzierung des Ressourcenverbrauchs beeinflussen. Verbindliche Anleitungen zur Implementierung finden Sie im [Whitepaper zur Säule „Nachhaltigkeit“](#).

Themen

- [Designprinzipien](#)
- [Definition](#)
- [Bewährte Methoden](#)
- [Ressourcen](#)

Designprinzipien

Es gibt sechs Designprinzipien für Nachhaltigkeit in der Cloud:

- **Verstehen Ihrer Auswirkungen:** Messen Sie die Auswirkungen Ihrer Cloud-Workloads und modellieren Sie diese Auswirkungen für die Zukunft. berücksichtigen Sie dabei alle relevanten Faktoren, darunter Auswirkungen durch die Verwendung Ihrer Produkte durch Kunden sowie solche durch deren Außerbetriebnahme und Entsorgung. Vergleichen Sie den produktiven Output mit den Gesamtauswirkungen Ihrer Cloud-Workloads, indem Sie die für jede Arbeitseinheit erforderlichen Ressourcen und die damit verbundenen Emissionen ermitteln. Verwenden Sie diese Daten, um wichtige Leistungsindikatoren (KPIs) festzulegen, Möglichkeiten zur Steigerung der Produktivität bei gleichzeitiger Verringerung der Auswirkungen zu bewerten und die Auswirkungen der vorgeschlagenen Änderungen im Laufe der Zeit abzuschätzen.
- **Festlegen von Nachhaltigkeitszielen:** Formulieren Sie für alle Cloud-Workloads langfristige Nachhaltigkeitsziele wie etwa die Reduzierung der pro Transaktion erforderlichen Datenverarbeitungs- und Speicherressourcen. Modellieren Sie den ROI von Verbesserungen in Bezug auf die Nachhaltigkeit vorhandener Workloads. Stellen Sie den Besitzern die nötigen Ressourcen zur Verfügung, um in Nachhaltigkeitsziele investieren zu können. Planen Sie wachstumsorientiert und gestalten Sie Ihre Workloads so, dass das Wachstum mit geringeren Auswirkungen einhergeht – gemessen in einer sinnvollen Einheit, etwa pro Benutzer oder pro Transaktion. Ziele helfen Ihnen, die allgemeinen Nachhaltigkeitsziele Ihres Unternehmens oder Ihrer Organisation zu erreichen, Rückschritte zu identifizieren und Bereiche mit Verbesserungsmöglichkeiten zu priorisieren.

- **Maximieren der Auslastung:** Sorgen Sie für Workloads angemessenen Umfangs und nutzen Sie effiziente Designprinzipien, um eine hohe Auslastung zu gewährleisten und die Energieeffizienz der zugrunde liegenden Hardware so zu maximieren. Zwei Hosts mit 30 % Auslastung sind aufgrund des grundlegenden Energieverbrauchs pro Host weniger effizient als ein Host mit 60 % Auslastung. Gleichzeitig sollten Sie nicht genutzte Ressourcen, Verarbeitungsvorgänge und Speicher reduzieren oder minimieren, um den Gesamtenergieverbrauch für Ihre Workloads zu senken.
- **Antizipieren und Einführen neuer und effizienterer Hardware- und Software-Angebote:** Unterstützen Sie die Verbesserungen, die Ihre Partner und Lieferanten in früheren Prozessphasen vornehmen, um die Auswirkungen Ihrer Cloud-Workloads zu reduzieren. Achten Sie stets auf neue und effizientere Hardware- und Software-Angebote. Planen Sie für Flexibilität, damit neue effiziente Technologien schnell eingeführt werden können.
- **Verwenden von verwalteten Services:** Die gemeinsame Nutzung von Services über eine breite Kundenbasis hinweg hilft dabei, die Ressourcennutzung zu maximieren und dadurch den Umfang der Infrastruktur zu verringern, der für die Unterstützung Ihrer Cloud-Workloads erforderlich ist. Kunden können beispielsweise die Auswirkungen gemeinsamer Rechenzentrumskomponenten wie Stromversorgung und Netzwerke gemeinsam nutzen, indem sie Workloads in die Umgebung migrieren AWS Cloud und Managed Services wie AWS Fargate für serverlose Container einsetzen, die skalierbar arbeiten und für deren effizienten AWS Betrieb verantwortlich sind. Nutzen Sie verwaltete Services, die Ihnen helfen können, Ihre Auswirkungen zu minimieren, wie z. B. die automatische Verschiebung selten aufgerufener Daten in Cold Storage mit Amazon S3 Lifecycle-Konfigurationen oder Amazon EC2 Auto Scaling, um die Kapazität an die Nachfrage anzupassen.
- **Reduzieren der nachgelagerten Auswirkungen Ihrer Cloud-Workloads:** Senken Sie den Energie- oder Ressourcenverbrauch für die Nutzung Ihrer Services. Reduzieren Sie die Erfordernis für Kunden, ihre Geräte zu aktualisieren, um Ihre Services nutzen zu können. Verwenden Sie in Ihren Tests Gerätefarmen, um die zu erwartenden Auswirkungen zu verstehen, und führen Sie Tests mit Kunden durch, um die tatsächlichen Auswirkungen der Nutzung Ihrer Services zu erkennen.

Definition

Es gibt sechs Bereiche für bewährte Methoden für Nachhaltigkeit in der Cloud:

- Auswahl der Region
- Ausrichtung am Bedarf
- Software und Architektur

- Daten
- Hardware und Services
- Prozess und Kultur

Nachhaltigkeit in der Cloud ist ein nahezu kontinuierliches Bestreben, das sich in erster Linie auf die Reduzierung des Energieverbrauchs und die Effizienz aller Komponenten einer Workload konzentriert. Dazu muss der maximale Nutzen aus den bereitgestellten Ressourcen gezogen und die insgesamt erforderlichen Ressourcen müssen minimiert werden. Diese Bemühung kann von der anfänglichen Auswahl einer effizienten Programmiersprache, der Einführung moderner Algorithmen, der Nutzung effizienter Datenspeichertechniken, der Bereitstellung einer korrekt dimensionierten und effizienten Recheninfrastruktur bis hin zur Minimierung der Anforderungen an leistungsstarke Endbenutzerhardware reichen.

Bewährte Methoden

Themen

- [Auswahl der Region](#)
- [Ausrichtung am Bedarf](#)
- [Software und Architektur](#)
- [Datenverwaltung](#)
- [Hardware und Services](#)
- [Prozess und Kultur](#)

Auswahl der Region

Die Wahl der Region für Ihren Workload wirkt sich erheblich auf dessen LeistungKPIs, Kosten und CO2-Fußabdruck aus. Um diese zu verbessernKPIs, sollten Sie Regionen für Ihre Workloads auswählen, die sowohl auf Geschäftsanforderungen als auch auf Nachhaltigkeitszielen basieren.

In der folgenden Frage geht es um Überlegungen zur Nachhaltigkeit. (Eine Liste der Fragen und bewährten Methoden zur Nachhaltigkeit finden Sie im [Anhang](#)).

SUS1: Wie wählen Sie Regionen für Ihren Workload aus?

Die Wahl der Region für Ihren Workload wirkt sich erheblich auf dessen LeistungKPIs, Kosten und CO2-Fußabdruck aus. Um diese zu verbessernKPIs, sollten Sie Regionen für Ihre Workloads auswählen, die sowohl auf Geschäftsanforderungen als auch auf Nachhaltigkeitszielen basieren.

Ausrichtung am Bedarf

Wenn Sie berücksichtigen, wie Benutzer und Anwendungen Ihre Workloads und andere Ressourcen nutzen, können Sie auf diese Weise Verbesserungsmöglichkeiten ermitteln, um Nachhaltigkeitsziele zu erreichen. Skalieren Sie Ihre Infrastruktur so, dass Sie den Bedarf kontinuierlich anpassen können. Sorgen Sie zudem dafür, dass zur Unterstützung Ihrer Benutzer nicht mehr Ressourcen verwendet werden als unbedingt nötig. Richten Sie Service-Levels an den Kundenanforderungen aus. Positionieren Sie Ressourcen so, dass die Netzwerkkapazitäten, die für Benutzer und Anwendungen erforderlich sind, begrenzt werden. Entfernen Sie ungenutzte Komponenten. Stellen Sie Teammitgliedern Geräte zur Verfügung, die ihre Anforderungen bei geringstmöglichen Auswirkungen auf die Nachhaltigkeit erfüllen.

Die folgende Frage konzentriert sich auf diese Überlegungen zur Nachhaltigkeit:

SUS2: Wie passen Sie Cloud-Ressourcen an Ihren Bedarf an?

Wenn Sie berücksichtigen, wie Benutzer und Anwendungen Ihre Workloads und andere Ressourcen nutzen, können Sie auf diese Weise Verbesserungsmöglichkeiten ermitteln, um Nachhaltigkeitsziele zu erreichen. Skalieren Sie Ihre Infrastruktur so, dass Sie den Bedarf kontinuierlich anpassen können. Sorgen Sie zudem dafür, dass zur Unterstützung Ihrer Benutzer nicht mehr Ressourcen verwendet werden als unbedingt nötig. Richten Sie Service-Levels an den Kundenanforderungen aus. Positionieren Sie Ressourcen so, dass die Netzwerkkapazitäten, die für Benutzer und Anwendungen erforderlich sind, begrenzt werden. Entfernen Sie ungenutzte Komponenten. Stellen Sie Teammitgliedern Geräte zur Verfügung, die ihre Anforderungen bei geringstmöglichen Auswirkungen auf die Nachhaltigkeit erfüllen.

Skalieren der Infrastruktur anhand der Benutzerlast: Identifizieren Sie Zeiträume mit geringer oder gar keiner Nutzung und skalieren Sie Ressourcen, um überschüssige Kapazitäten zu entfernen und die Effizienz zu verbessern.

Auf Nachhaltigkeitsziele ausrichten SLAs: Definieren und aktualisieren Sie Service Level Agreements (SLAs) wie Verfügbarkeits- oder Datenaufbewahrungsfristen, um die Anzahl der Ressourcen zu minimieren, die zur Unterstützung Ihrer Arbeitslast erforderlich sind, und gleichzeitig die Geschäftsanforderungen zu erfüllen.

Beenden der Erstellung und Wartung nicht verwendeter Komponenten: Analysieren Sie Anwendungskomponenten (wie vorab kompilierte Berichte, Datensätze und statische Bilder) sowie Zugriffsmuster für Komponenten, um Redundanzen, eine zu geringe Auslastung und mögliche Kandidaten für die Außerbetriebnahme zu identifizieren. Konsolidieren Sie generierte Komponenten mit redundanten Inhalten (z. B. monatliche Berichte mit sich überschneidenden oder gemeinsam genutzten Datensätzen und Ausgaben), um für duplizierte Ausgaben genutzte Ressourcen zu eliminieren. Deaktivieren Sie nicht verwendete Komponenten (z. B. Bilder von Produkten, die nicht mehr verkauft werden), um genutzte Ressourcen freizugeben und die Zahl der Ressourcen zu reduzieren, die zur Unterstützung von Workloads verwendet werden.

Optimieren der geografischen Platzierung von Workloads für Benutzerstandorte: Analysieren Sie Netzwerkzugriffsmuster, um zu erkennen, aus welchen geografischen Regionen Ihre Kunden Verbindungen herstellen. Wählen Sie Regionen und Services im Hinblick auf die Reduzierung der Distanz für den Netzwerkdatenverkehr aus, um die Zahl der Netzwerkressourcen zu verringern, die zur Unterstützung von Workloads benötigt werden.

Optimieren von Ressourcen für Teammitglieder im Hinblick auf die ausgeführten Aktivitäten: Optimieren Sie die Ressourcen, die Teammitgliedern zur Verfügung gestellt werden, um negative Auswirkungen auf die Nachhaltigkeit zu minimieren und gleichzeitig ihre Anforderungen zu erfüllen. Beispielsweise können Sie komplexe Vorgänge wie Rendering und Kompilierung auf intensiv genutzten, geteilten Cloud-Desktops statt auf weniger ausgelasteten Einzelbenutzersystemen mit hohem Energieverbrauch ausführen.

Software und Architektur

Implementieren Sie Muster für den Lastausgleich und die Wahrung einer konsistent hohen Nutzung der bereitgestellten Ressourcen, um die Zahl der genutzten Ressourcen zu minimieren. Komponenten werden möglicherweise aufgrund von Änderungen des Benutzerverhaltens über die Zeit nicht mehr genutzt. Prüfen Sie Muster und Architekturen, um nicht ausreichend genutzte Komponenten zu konsolidieren und so die Nutzung insgesamt zu erhöhen. Nehmen Sie Komponenten außer Betrieb, die nicht mehr benötigt werden. Identifizieren Sie die Leistung Ihrer Workload-Komponenten und optimieren Sie die Komponenten, die die meisten Ressourcen verbrauchen. Achten Sie auf die Geräte, mit denen Ihre Kunden auf Ihre Services zugreifen, und implementieren Sie Muster, um den Bedarf für Geräte-Upgrades zu minimieren.

In den folgenden Fragen geht es um diese Überlegungen zur Nachhaltigkeit:

SUS3: Wie nutzen Sie Software- und Architekturmuster, um Ihre Nachhaltigkeitsziele zu erreichen ?

Implementieren Sie Muster für den Lastausgleich und die Wahrung einer konsistent hohen Nutzung der bereitgestellten Ressourcen, um die Zahl der genutzten Ressourcen zu minimieren. Komponenten werden möglicherweise aufgrund von Änderungen des Benutzerverhaltens über die Zeit nicht mehr genutzt. Prüfen Sie Muster und Architekturen, um nicht ausreichend genutzte Komponenten zu konsolidieren und so die Nutzung insgesamt zu erhöhen. Nehmen Sie Komponenten außer Betrieb, die nicht mehr benötigt werden. Identifizieren Sie die Leistung Ihrer Workload-Komponenten und optimieren Sie die Komponenten, die die meisten Ressourcen verbrauchen. Achten Sie auf die Geräte, mit denen Ihre Kunden auf Ihre Services zugreifen, und implementieren Sie Muster, um den Bedarf für Geräte-Upgrades zu minimieren.

Optimieren von Software und Architektur für asynchrone und geplante Aufträge: Verwenden Sie effiziente Softwaredesigns und Architekturen, um die Zahl der für einzelne Arbeitseinheiten im Durchschnitt benötigten Ressourcen zu minimieren. Implementieren Sie Mechanismen für die gleichmäßige Nutzung von Komponenten, um die Zahl der Ressourcen zu reduzieren, die zwischen Aufgaben nicht genutzt werden, und die Auswirkungen von Lastspitzen zu minimieren.

Entfernen von Workload-Komponenten mit geringer oder keiner Nutzung oder Faktorwechsel: Überwachen Sie die Workload-Aktivität, um Änderungen bei der Nutzung einzelner Komponenten über die Zeit zu erkennen. Entfernen Sie ungenutzte Komponenten, die nicht mehr benötigt werden. Führen Sie einen Faktorwechsel für wenig genutzte Ressourcen durch, um die Verschwendung von Ressourcen zu begrenzen.

Optimieren von Codebereichen, die die meiste Zeit oder die meisten Ressourcen verbrauchen: Überwachen Sie die Workload-Aktivität, um die Anwendungskomponenten zu identifizieren, die die meisten Ressourcen verbrauchen. Optimieren Sie den Code, der innerhalb dieser Komponenten ausgeführt wird, um die Ressourcennutzung zu minimieren und die Leistung zu maximieren.

Optimieren der Auswirkungen auf Geräte und Ausrüstung von Kunden: Identifizieren Sie die Geräte und Einrichtungen, mit denen Ihre Kunden Ihre Services nutzen, ihren voraussichtlichen Lebenszyklus und die finanziellen und nachhaltigkeitsbezogenen Auswirkungen der Ersetzung dieser Komponenten. Implementieren Sie Softwaremuster und Architekturen, die es für Kunden unnötig machen, Geräte zu ersetzen oder ihre Ausrüstung zu aktualisieren. Implementieren

Sie beispielsweise neue Features, die Code verwenden, der mit älterer Hardware und älteren Betriebssystemversionen abwärtskompatibel ist, oder gestalten Sie die Größe von Nutzlasten so, dass sie die Speicherkapazitäten der Zielgeräte nicht überschreiten.

Verwenden von Softwaremustern und Architekturen, die Datenzugriffs- und Speichermuster optimal unterstützen: Identifizieren Sie, wie Daten in Ihrer Workload verwendet, von Benutzern genutzt, übertragen und gespeichert werden. Wählen Sie Technologien aus, die die Anforderungen an Datenverarbeitung und -speicherung minimieren.

Datenverwaltung

In den folgenden Fragen geht es um diese Überlegungen zur Nachhaltigkeit:

SUS4: Wie nutzen Sie Datenmanagementrichtlinien und -muster, um Ihre Nachhaltigkeitsziele zu unterstützen?

Implementieren Sie Verfahren für die Datenverwaltung, die den zur Unterstützung Ihrer Workload bereitgestellten Speicher und die für dessen Nutzung erforderlichen Ressourcen reduzieren. Verstehen Sie Ihre Daten und setzen Sie Speichertechnologien und -konfigurationen ein, die den geschäftlichen Mehrwert der Daten und deren Nutzung am besten fördern. Verschieben Sie die Daten während des Lebenszyklus zu effizienteren Speichern mit geringerer Leistung, wenn die Anforderungen abnehmen. Löschen Sie Daten, die nicht mehr benötigt werden.

Implementieren einer Richtlinie für die Klassifizierung von Daten: Klassifizieren Sie Daten, um ihre Bedeutung für geschäftliche Ergebnisse zu verstehen. Nutzen Sie diese Informationen, um festzulegen, wann Daten in einen energieeffizienteren Speicher übertragen oder auf sichere Weise gelöscht werden können.

Verwenden von Technologien, die Datenzugriff und Speichermuster unterstützen: Nutzen Sie einen Speicher, der den Zugriff auf Ihre Daten und ihre Speicherung jeweils optimal unterstützt, um die Zahl der bereitgestellten Ressourcen zu minimieren und gleichzeitig Ihre Workload zu unterstützen. Festkörpergeräte (SSDs) sind beispielsweise energieintensiver als Magnetlaufwerke und sollten nur für aktive Datenanwendungen verwendet werden. Verwenden Sie für Daten, auf die nicht häufig zugegriffen wird, einen energieeffizienten Archivierungsspeicher.

Verwenden von Lebenszyklusrichtlinien zum Löschen nicht notwendiger Daten: Verwalten Sie den Lebenszyklus aller Daten und setzen Sie automatisch Löschfristen durch, um die Speicheranforderungen Ihrer Workload insgesamt zu minimieren.

Minimieren übermäßiger Bereitstellungen im Blockspeicher: Erstellen Sie zur Minimierung des insgesamt bereitgestellten Speichers Blockspeicher mit Größenzuweisungen entsprechend der jeweiligen Workload. Verwenden Sie elastische Volumes, um den Speicher bei wachsenden Datenmengen erweitern zu können, ohne die Größe des an Datenverarbeitungsressourcen angefügten Speichers ändern zu müssen. Überprüfen Sie elastische Volumes regelmäßig und verkleinern Sie zu große Volumes, um sie an den aktuellen Datenumfang anzupassen.

Entfernen nicht benötigter oder redundanter Daten: Duplizieren Sie Daten nur wie notwendig, um den insgesamt genutzten Speicher zu minimieren. Verwenden Sie Sicherungstechnologien, die Daten auf Datei- und Blockebene deduplizieren. Beschränken Sie die Verwendung von Konfigurationen mit redundanten Arrays unabhängiger Laufwerke (RAID), es sei denn, dies ist erforderlich, um folgende Anforderungen zu erfüllen SLAs.

Verwenden geteilter Dateisysteme oder Objektspeicher für den Zugriff auf allgemeine Daten: Verwenden Sie geteilten Speicher und zentrale Datenquellen, um Datenduplizierungen zu vermeiden und den Gesamtspeicherbedarf Ihrer Workload zu reduzieren. Rufen Sie Daten nur bei Bedarf aus dem gemeinsam genutzten Speicher ab. Trennen Sie nicht genutzte Volumes, um Ressourcen freizugeben. Minimieren Sie Datenübertragungen über Netzwerke hinweg. Verwenden Sie stattdessen einen geteilten Speicher und greifen Sie über regionale Datenspeicher auf die Daten zu, um die Zahl der Netzwerkressourcen zu minimieren, die für Datenübertragungen für Ihre Workload benötigt werden.

Sichern von Daten nur in dem Fall, dass ihre erneute Erstellung schwierig ist: Sichern Sie zur Minimierung der Speichernutzung nur Daten, die einen Unternehmenswert besitzen oder zur Erfüllung von Compliance-Anforderungen benötigt werden. Prüfen Sie Sicherheitsrichtlinien und vermeiden Sie einen flüchtigen Speicher, der in einem Wiederherstellungsszenario keinen Wert bietet.

Hardware und Services

Suchen Sie nach Möglichkeiten, die Auswirkungen auf die Nachhaltigkeit Ihrer Workloads durch Änderungen der Methoden für die Hardwareverwaltung zu reduzieren. Minimieren Sie den Umfang der für die Bereitstellung erforderlichen Hardware und wählen Sie die jeweils effizienteste Hardware und den effizientesten Service für die jeweilige Workload aus.

In den folgenden Fragen geht es um diese Überlegungen zur Nachhaltigkeit:

SUS5: Wie wählen und nutzen Sie Cloud-Hardware und -Services in Ihrer Architektur, um Ihre Nachhaltigkeitsziele zu unterstützen?

Suchen Sie nach Möglichkeiten, die Auswirkungen auf die Nachhaltigkeit Ihrer Workloads durch Änderungen der Methoden für die Hardwareverwaltung zu reduzieren. Minimieren Sie den Umfang der für die Bereitstellung erforderlichen Hardware und wählen Sie die jeweils effizienteste Hardware und den effizientesten Service für die jeweilige Workload aus.

Verwenden der geringstmöglichen Menge an Hardware zur Erfüllung Ihrer Anforderungen: Mit den Möglichkeiten der Cloud können Sie häufige Änderungen für Ihre Workload-Implementierungen ausführen. Aktualisieren Sie bereitgestellte Komponenten, wenn sich Ihre Anforderungen ändern.

Verwendung von Instance-Typen mit den geringsten Auswirkungen: Überwachen Sie kontinuierlich die Einführung neuer Instance-Typen und nutzen Sie Verbesserungen bei der Energieeffizienz, einschließlich Instance-Typen, die zur Unterstützung spezifischer Workloads bestimmt sind, wie z. B. Machine-Learning-Trainings und -Inferenzen und Videotranskodierung.

Nutzen Sie Managed Services: Managed Services verlagern die Verantwortung für die Aufrechterhaltung einer hohen durchschnittlichen Auslastung und die Nachhaltigkeitsoptimierung der eingesetzten Hardware auf AWS. Mit verwalteten Services können Sie die nachhaltigkeitsbezogenen Auswirkungen des Service über alle Mandanten des Service verteilen und so Ihren Beitrag verringern.

Optimieren Sie Ihre Nutzung von GPUs: Grafikprozessoren (GPUs) können einen hohen Stromverbrauch verursachen, und viele GPU Workloads sind sehr variabel, wie z. B. Rendern, Transcodieren sowie Training und Modellierung für maschinelles Lernen. Führen Sie GPUs Instanzen nur für die benötigte Zeit aus und nehmen Sie sie automatisiert außer Betrieb, wenn sie nicht benötigt werden, um den Ressourcenverbrauch zu minimieren.

Prozess und Kultur

Reduzieren Sie nachhaltigkeitsbezogene Auswirkungen, indem Sie Ihre Entwicklungs-, Test- und Bereitstellungsmethoden ändern.

In den folgenden Fragen geht es um diese Überlegungen zur Nachhaltigkeit:

SUS6: Wie unterstützen Ihre organisatorischen Prozesse Ihre Nachhaltigkeitsziele?

Reduzieren Sie nachhaltigkeitsbezogene Auswirkungen, indem Sie Ihre Entwicklungs-, Test- und Bereitstellungsmethoden ändern.

Einführen von Methoden, die schnelle Verbesserungen für die Nachhaltigkeit ermöglichen: Testen und validieren Sie potenzielle Verbesserungen, bevor Sie sie für die Produktion bereitstellen. Berücksichtigen Sie die Testkosten bei der Berechnung des potenziellen zukünftigen Nutzens einer Verbesserung. Entwickeln Sie kostengünstige Testmethoden, um kleine Verbesserungen einzuführen.

Halten Sie Ihren Workload auf dem neuesten Stand: Up-to-date Betriebssysteme, Bibliotheken und Anwendungen können die Workload-Effizienz verbessern und die Einführung effizienterer Technologien fördern. Up-to-date Software kann auch Funktionen enthalten, mit denen Sie die Auswirkungen Ihrer Workloads auf die Nachhaltigkeit genauer messen können, da Anbieter Funktionen bereitstellen, mit denen Sie Ihre eigenen Nachhaltigkeitsziele erreichen können.

Höhere Auslastung von Entwicklungsumgebungen: Verwenden Sie Automatisierung und Infrastructure-as-Code, um Vorproduktionsumgebungen bei Bedarf in Betrieb und bei Nichtverwendung wieder außer Betrieb zu nehmen. Eine typische Vorgehensweise besteht in der Planung von Verfügbarkeitszeiten, die mit den Arbeitszeiten der Entwicklungsteams übereinstimmen. Der Ruhezustand ist ein nützliches Tool, um den Status beizubehalten und Instances nur bei Bedarf schnell online zu schalten. Verwenden Sie Instance-Typen mit Burst-Kapazität, Spot Instances, Services für elastische Datenbanken, Container und andere Technologien, um Entwicklungs- und Testkapazitäten an die Nutzung anzupassen.

Verwenden verwalteter Gerätefarmen für Tests verwenden: Verwaltete Gerätefarmen verteilen die nachhaltigkeitsbezogenen Auswirkungen der Hardwarefertigung und der Ressourcennutzung über zahlreiche Mandanten. Verwaltete Gerätefarmen stellen verschiedene Gerätetypen bereit, unterstützen auch ältere und weniger verbreitete Hardware und vermeiden nachhaltigkeitsbezogene Auswirkungen durch unnötige Geräte-Upgrades seitens Kunden.

Ressourcen

Werfen Sie einen Blick auf die folgenden Ressourcen, um mehr über unsere bewährten Methoden für die Nachhaltigkeit zu erfahren.

Whitepaper

- [Säule „Nachhaltigkeit“](#)

Video

- [The Climate Pledge](#)

Der Überprüfungsprozess

Architekturen müssen nach einheitlichen Gesichtspunkten überprüft werden. Wenn dabei niemand an den Pranger gestellt wird, ist eine Voraussetzung für tief schürfende Analysen gegeben. Der Prozess sollte nicht schwerfällig sein (Stunden, nicht Tage) und als Konversation angelegt sein, nicht als Audit. Architekturen werden überprüft, um festzustellen, ob kritische Mängel vorliegen, gegen die etwas unternommen werden muss – oder um festzustellen, ob bestimmte Bereiche nachgebessert werden können. Am Ende der Überprüfung stehen Maßnahmen, die dem Kunden, der mit der Workload arbeitet, ein angenehmeres Erlebnis ermöglichen.

Wie bereits im Abschnitt „Architekturüberlegungen“ angesprochen, ist es in Ihrem Interesse, dass jedes Teammitglied Verantwortung für die Qualität der Architektur übernimmt. Wir empfehlen, dass die Teammitglieder, die die Architektur entwerfen, mit Hilfe des Well-Architected Framework ihre Architektur fortlaufend überprüfen, anstatt eine formelle Überprüfungsbesprechung anzusetzen. Findet die Überprüfung nahezu fortlaufend statt, können Ihre Teammitglieder parallel mit der Entwicklung der Architektur Antworten aktualisieren und mit jedem neuen Feature die Architektur verbessern.

Das AWS Well-Architected Framework ist auf die Art und Weise ausgerichtet, wie Systeme und Dienste intern AWS überprüft werden. Es basiert auf einer Reihe von Entwurfsprinzipien, die den architektonischen Ansatz beeinflussen, sowie auf Fragen, die sicherstellen, dass Bereiche, die häufig in der Ursachenanalyse () behandelt werden, nicht vernachlässigt werden. RCA Immer wenn es ein schwerwiegendes Problem mit einem internen System, einem AWS Service oder einem Kunden gibt, prüfen wir, ob wir die RCA von uns verwendeten Überprüfungsprozesse verbessern können.

Die Überprüfungen müssen an wichtigen Meilensteinen des Produktzyklus erfolgen – früh in der Entwurfsphase, um Einbahnstraßen zu vermeiden, an denen schwer nachzubessern ist. Und zuletzt schließlich kurz vor dem Go-Live. (Viele Entscheidungen können rückgängig gemacht werden; es gibt zwei Möglichkeiten. Für diese Entscheidungen reicht ein schlanker Prozess. Gibt es nur eine Möglichkeit, kann diese nur schwer oder gar nicht rückgängig gemacht werden und muss genauer inspiziert werden, bevor sie gewählt wird.) Nachdem Sie in Produktion gehen, verändert sich Ihre Workload weiter, da neue Funktionen hinzukommen und Sie Technologieimplementierungen anpassen. Die Architektur einer Workload verändert sich mit der Zeit. Treffen Sie durchdachte Hygienemaßnahmen, um zu verhindern, dass die Qualität seiner architektonischen Merkmale im Zuge der Weiterentwicklung nachlässt. Wenn Sie an der Architektur signifikante Änderungen vornehmen, müssen Sie bestimmte Hygieneprozesse befolgen, z. B. eine Überprüfung nach dem Well-Architected-Prinzip.

Wenn die Überprüfung als einmalige Momentaufnahme oder unabhängige Messung vorgesehen ist, müssen alle wichtigen Beteiligten in die Konversation eingebunden sein. Häufig ist die Überprüfung der Punkt, an dem einem Team das erste Mal richtig klar wird, was es implementiert hat. Wird die Workload eines anderen Teams überprüft, ist es sinnvoll, mehrere informelle Konversationen über deren Architektur einzuplanen. In diesen Gesprächen erhalten Sie Antworten auf die meisten Fragen. Im Anschluss daran können Sie in ein oder zwei Besprechungen Punkte abklären und ausführlich auf Unklarheiten oder eventuelle Risiken eingehen.

Damit Ihre Besprechungen erfolgreich verlaufen, empfehlen wir folgende Ausstattung:

- Besprechungszimmer mit Whiteboards
- Diagramme und Entwurfsnotizen ausgedruckt auf Papier
- Maßnahmenliste mit Fragen, für out-of-band deren Beantwortung Nachforschungen erforderlich sind (z. B. „Haben wir die Verschlüsselung aktiviert oder nicht?“)

Nach der Überprüfung sollten Sie eine Liste mit Problemen vorliegen haben. Welche Sie priorisieren, hängt vom geschäftlichen Kontext ab. Sie sollten auch die Auswirkungen dieser Probleme auf die day-to-day Arbeit Ihres Teams berücksichtigen. Wenn Sie die Probleme frühzeitig angehen, gewinnen Sie vielleicht Zeit. Zeit, in der Sie geschäftlichen Mehrwert schaffen können, anstatt sich um wiederkehrende Probleme zu kümmern. Während Sie die Probleme aus der Welt schaffen, können Sie Ihre Überprüfung aktualisieren und so verfolgen, wie sich die Architektur verbessert.

Wie hilfreich eine Überprüfung war, zeigt sich erst danach. Neue Teams widersetzen sich möglicherweise zuerst. Sie können Einwänden der Teams entgegen, indem Sie sie über die Vorteile einer Überprüfung aufklären:

- „Wir sind zu beschäftigt!“ (Häufig im Vorfeld großer Produktstarts zu hören.)
 - Wenn ihr euch auf einen großen Launch vorbereitet, sollte der möglichst glatt über die Bühne gehen. Die Überprüfung deckt Schwachstellen auf, die ihr vielleicht übersehen habt.
 - Wir empfehlen, dass ihr früh im Produktzyklus Überprüfungen einbaut, um Risiken aufzudecken und einen Auffangplan auszuarbeiten, der auf die Roadmap für die Feature-Bereitstellung abgestimmt ist.
- „Wir haben nicht die Zeit, um mit den Ergebnissen etwas anzufangen!“ (Oft zu hören, wenn ein unverrückbares Ereignis näher rückt, z. B. eine große Sportveranstaltung, auf das alles ausgerichtet ist.)

- Diese Ereignisse lassen sich nicht verschieben. Wollt ihr da wirklich reingehen, ohne die Risiken eurer Architektur zu kennen? Selbst wenn ihr nicht alle Probleme wegbekommt, könnt ihr euch immer noch mit Playbooks helfen, wenn sie tatsächlich eintreten.
- „Wir möchten nicht, dass andere die Geheimnisse unserer Lösungsimplementierung kennenlernen!“
- Wenn Sie die Aufmerksamkeit des Teams auf die Fragen im Well-Architected Framework richten, erkennen sie, dass keine der Fragen kommerziell oder technisch geschützte Informationen preisgibt.

Wenn Sie mit Teams aus Ihrer Organisation mehrere Überprüfungen durchführen, identifizieren Sie möglicherweise thematische Fragen. So könnte sich beispielsweise herausstellen, dass mehrere Teams in einer bestimmten Säule oder einem bestimmten Themengebiet mehrere zusammenhängende Probleme haben. Werfen Sie einen ganzheitlichen Blick auf all Ihre Überprüfungen und identifizieren Sie Mechanismen, Trainings oder Principal-Engineer-Vorträge, mit deren Hilfe sich diese thematischen Fragen angehen lassen.

Schlussfolgerung

Das AWS Well-Architected Framework bietet architektonische Best Practices in den sechs Säulen für die Entwicklung und den Betrieb zuverlässiger, sicherer, effizienter, kostengünstiger und nachhaltiger Systeme in der Cloud. Die Fragen aus dem Framework erlauben Ihnen, bestehende und geplante Architekturen zu überprüfen. Es bietet auch eine Reihe von AWS Best Practices für jede Säule. Als fester Bestandteil Ihres Architekturdesigns fördert das Framework stabile und effiziente Systeme. Anschließend können Sie sich auf Ihre funktionalen Anforderungen konzentrieren.

Mitwirkende

Folgende Personen und Organisationen haben zu diesem Dokument beigetragen:

- Brian Carlson, Operations Lead Well-Architected, Amazon Web Services
- Ben Potter, Security Lead Well-Architected, Amazon Web Services
- Seth Eliot, Reliability Lead Well-Architected, Amazon Web Services
- Eric Pullen, Sr. Solutions Architect, Amazon Web Services
- Rodney Lester, Principal Solutions Architect, Amazon Web Services
- Jon Steele, Sr. Technical Account Manager, Amazon Web Services
- Max Ramsay, Principal Security Solutions Architect, Amazon Web Services
- Callum Hughes, Solutions Architect, Amazon Web Services
- Ben Mergen, Senior Cost Lead Solutions Architect, Amazon Web Services
- Chris Kozlowski, Senior Specialist Technical Account Manager, Enterprise Support, Amazon Web Services
- Alex Livingstone, Principal Specialist Solutions Architect, Cloud Operations, Amazon Web Services
- Paul Moran, Principal Technologist, Enterprise Support, Amazon Web Services
- Peter Mullen, Advisory Consultant, Professional Services, Amazon Web Services
- Chris Pates, Senior Specialist Technical Account Manager, Enterprise Support, Amazon Web Services
- Arvind Raghunathan, Principal Specialist Technical Account Manager, Enterprise Support, Amazon Web Services
- Sam Mokhtari, Senior Efficiency Lead Solutions Architect, Amazon Web Services

Weitere Informationen

[AWS -Architekturzentrum](#)

[AWS Cloud-Compliance](#)

[AWS Well-Architected-Partnerprogramm](#)

[AWS Well-Architected Tool](#)

[AWS Well-Architected Homepage](#)

[Whitepaper zur Säule „Betriebliche Exzellenz“](#)

[Whitepaper zur Säule „Sicherheit“](#)

[Whitepaper zur Säule „Zuverlässigkeit“](#)

[Whitepaper zur Säule „Leistungseffizienz“](#)

[Whitepaper zur Säule „Kostenoptimierung“](#)

[Whitepaper zur Säule „Nachhaltigkeit“](#)

[Die Amazon Builders' Library](#)

Dokumentversionen

Abonnieren Sie den RSS-Feed, um über Aktualisierungen des Whitepapers benachrichtigt zu werden.

Änderung	Beschreibung	Datum
Größere Aktualisierung	Best Practices wurden mit neuen Leitfäden in den folgenden Bereichen aktualisiert: Zuverlässigkeit, Sicherheit, Operational Excellence, Nachhaltigkeit und Leistungseffizienz. Die Säule der Zuverlässigkeit wurde umfassend aktualisiert und viele bewährte Verfahren wurden aktualisiert. Die Leitfäden in den Bereichen Sicherheit und Operational Excellence wurden aktualisiert und mit neuen Service- und Vorschlägen von generativer KI verfeinert. Die Nachhaltigkeit wurde auf der Grundlage von AWS-Services und neuen Best Practices mehrfach aktualisiert.	6. November 2024
Größere Aktualisierung	In den Säulen wurden umfangreiche Aktualisierungen der bewährten Methoden vorgenommen. Für Sicherheit und Kosten wurden neue bewährte Methoden hinzugefügt.	27. Juni 2024

<u>Größere Aktualisierung</u>	Größere Aktualisierungen der Säulen.	3. Oktober 2023
<u>Größere Aktualisierung</u>	Bewährte Methoden mit verbindlichen Anleitungen aktualisiert und neue bewährte Methoden hinzugefügt. Neue Fragen zu den Säulen Sicherheit und Kostenoptimierung hinzugefügt.	10. April 2023
<u>Kleines Update</u>	Eine Definition für Grad des Aufwands wurde hinzugefügt und bewährte Methoden im Anhang wurden aktualisiert.	20. Oktober 2022
<u>Größere Aktualisierung</u>	Die Säule „Nachhaltigkeit“ wurde hinzugefügt und Links wurden aktualisiert.	2. Dezember 2021
<u>Größere Aktualisierung</u>	Die Säule „Nachhaltigkeit“ wurde dem Framework hinzugefügt.	20. November 2021
<u>Kleines Update</u>	Nicht inklusive Sprache entfernt.	22. April 2021
<u>Kleines Update</u>	Zahlreiche Links wurden repariert.	10. März 2021
<u>Kleines Update</u>	Kleinere redaktionelle Änderungen im gesamten Dokument.	15. Juli 2020
<u>Größere Aktualisierung</u>	Prüfung und Umformulierung der meisten Fragen und Antworten.	8. Juli 2020

Whitepaper aktualisiert	Ergänzung des AWS Well-Architected Tool, Links zu AWS Well-Architected Labs und AWS Well-Architected-Partnern, kleinere Fehlerbehebungen zur Aktivierung mehrerer Sprachversionen des Frameworks.	1. Juli 2019
Whitepaper aktualisiert	Die meisten Fragen und Antworten wurden noch einmal durchgelesen und umgeschrieben, damit die Fragen jeweils nur ein Thema behandeln. Dabei wurden einige Fragen in mehrere Einzelfragen aufgeteilt. Häufig verwendete Begriffe (Workload, Komponente usw.) wurden definiert. Darstellung der Fragen im Textkorpus wurde bearbeitet, um Platz zu schaffen für Erläuterungen.	1. November 2018
Whitepaper aktualisiert	Fragentext ist nach mehreren Aktualisierungen einfacher formuliert, Antworten sind standardisiert und die Lesbarkeit wurde verbessert.	1. Juni 2018
Whitepaper aktualisiert	Betriebliche Exzellenz wurde vor die anderen Säulen gesetzt und umgeschrieben. Umfasst jetzt die anderen Säulen. Die anderen Säulen wurden aktualisiert, um der Weiterentwicklung von AWS Rechnung zu tragen.	1. November 2017

Whitepaper aktualisiert	Aktualisierung des Framework . Dieses enthält jetzt die Säule „Betriebliche Exzellenz“. Die anderen Säulen wurden überarbeitet und aktualisiert. Dabei wurden Doppelungen ausgeräumt und Erkenntnisse aus Überprüfungen bei mehreren Tausend Kunden aufgenommen.	1. November 2016
Kleinere Updates	Aktualisierung des Anhangs mit neuen Informationen zu Amazon CloudWatch Logs.	1. November 2015
Erste Veröffentlichung	AWS Well-Architected-Framework wurde veröffentlicht.	1. Oktober 2015

 Note

Um RSS-Aktualisierungen zu abonnieren, muss ein RSS-Plugin für den von Ihnen verwendeten Browser aktiviert sein.

Framework-Versionen:

- [27.06.2024](#)
- [03.10.2023](#)
- [10.04.2023](#)
- [31.03.2022](#)

Anhang: Fragen und bewährte Methoden

Dieser Anhang fasst alle Fragen und bewährten Methoden im AWS Well-Architected Framework zusammen.

Säulen

- [Operative Exzellenz](#)
- [Sicherheit](#)
- [Zuverlässigkeit](#)
- [Leistungseffizienz](#)
- [Kostenoptimierung](#)
- [Nachhaltigkeit](#)

Operative Exzellenz

Operative Exzellenz (OE) beinhaltet die Verpflichtung, Software korrekt zu entwickeln und dabei durchgehend ein hervorragendes Kundenerlebnis zu bieten. Dies umfasst bewährte Methoden für die Organisation Ihres Teams, die Gestaltung Ihres Workloads, den Betrieb in großem Maßstab und die Weiterentwicklung im Laufe der Zeit. Verbindliche Anleitungen zur Implementierung finden Sie im [Whitepaper „Säule der betrieblichen Exzellenz“](#).

Bereiche für bewährte Methoden

- [Organisation](#)
- [Vorbereitung](#)
- [Betrieb](#)
- [Weiterentwicklung](#)

Organisation

Fragen

- [OPS 1. Wie können Sie Ihre Prioritäten bestimmen?](#)
- [OPS 2. Wie strukturieren Sie Ihr Unternehmen, um die gewünschten Geschäftsergebnisse zu erzielen?](#)

- [OPS 3. Wie unterstützt Ihre Unternehmenskultur Ihre Geschäftsergebnisse?](#)

OPS 1. Wie können Sie Ihre Prioritäten bestimmen?

Jeder muss verstehen, welchen Beitrag er zum Geschäftserfolg leistet. Setzen Sie sich gemeinsame Ziele, damit Sie die Prioritäten für Ressourcen festlegen können. Dadurch erzielen Ihre Bemühungen den größtmöglichen Nutzen.

Best Practices

- [OPS01-BP01 Bewerten der Bedürfnisse externer Kunden](#)
- [OPS01-BP02 Evaluieren Sie die internen Kundenbedürfnisse](#)
- [OPS01-BP03 Bewertung der Governance-Anforderungen](#)
- [OPS01-BP04 Bewerten der Compliance-Anforderungen](#)
- [OPS01-BP05 Bewerten Sie die Bedrohungslandschaft](#)
- [OPS01-BP06 Bewerten von Kompromissen und Abwägen der Vorteile und Risiken](#)

OPS01-BP01 Bewerten der Bedürfnisse externer Kunden

Binden Sie alle wichtigen Stakeholder ein, einschließlich Geschäfts-, Entwicklungs- und Betriebsteams, um zu bestimmen, welche Bereiche verstärkt auf die Bedürfnisse der externen Kunden ausgerichtet werden müssen. Dadurch wird sichergestellt, dass Sie mit der betrieblichen Unterstützung vertraut sind, die erforderlich ist, um die gewünschten geschäftlichen Ergebnisse zu erzielen.

Gewünschtes Ergebnis:

- Sie arbeiten rückwärts von den Kundenergebnissen aus.
- Sie wissen, wie Ihre betrieblichen Praktiken Geschäftsergebnisse und -ziele unterstützen.
- Sie binden alle relevanten Parteien ein.
- Sie verfügen über Mechanismen, um die Bedürfnisse externer Kunden zu erfassen.

Typische Anti-Muster:

- Sie haben sich entschieden, außerhalb der Kerngeschäftszeiten keinen Kundenservice zu bieten, aber Sie haben dazu keine historischen Supportanfragedaten analysiert. Daher wissen Sie nicht, ob diese Entscheidung Auswirkungen auf Ihre Kunden hat.

- Sie entwickeln ein neues Feature, haben aber Ihre Kunden nicht miteinbezogen, um herauszufinden, ob die Funktion erwünscht ist und wie sie genau aussehen sollte. Außerdem haben Sie keine Tests durchgeführt, um die Nachfrage und die Methode der Bereitstellung zu validieren.

Vorteile der Nutzung dieser bewährten Methode: Kunden, deren Anforderungen erfüllt sind, bleiben mit höherer Wahrscheinlichkeit als Kunden erhalten. Die Bewertung und das Verständnis externer Kundenbedürfnisse liefert die Grundlage dafür, wie Sie Ihre Anstrengungen zur Bereitstellung eines geschäftlichen Mehrwerts priorisieren.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Geschäftliche Anforderungen verstehen: Der geschäftliche Erfolg basiert auf gemeinsamen Zielen und der Kommunikation zwischen allen Stakeholdern, zu denen auch die Teams aus den Bereichen Geschäft, Entwicklung und Betrieb gehören.

Besprechen der geschäftlichen Ziele, Anforderungen und Prioritäten externer Kunden: Führen Sie wichtige Beteiligte zusammen, einschließlich Geschäfts-, Entwicklungs- und Betriebsteams, um die Ziele, Anforderungen und Prioritäten externer Kunden zu besprechen.. Dadurch wird sichergestellt, dass Sie mit der betrieblichen Unterstützung vertraut sind, die erforderlich ist, um die gewünschten Geschäfts- und Kundenergebnisse zu erzielen.

Schaffen Sie ein gemeinsames Verständnis: Sorgen Sie dafür, dass alle Beteiligten die Geschäftsfunktionen des Workloads und die Rollen der einzelnen Teams bei den Workload-spezifischen betrieblichen Abläufen kennen. Außerdem sollte bekannt sein, wie diese Faktoren die gemeinsamen Geschäftsziele mit internen und externen Kunden beeinflussen.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS11-BP03 Implementieren von Feedbackschleifen](#)

OPS01-BP02 Evaluieren Sie die internen Kundenbedürfnisse

Binden Sie alle wichtigen Stakeholder ein, einschließlich Geschäfts-, Entwicklungs- und Betriebsteams, um zu bestimmen, welche Bereiche verstärkt auf die Bedürfnisse der internen

Kunden ausgerichtet werden müssen. Dadurch wird sichergestellt, dass Sie mit der betrieblichen Unterstützung vertraut sind, die erforderlich ist, um geschäftliche Ergebnisse zu erzielen.

Gewünschtes Ergebnis:

- Anhand Ihrer etablierten Prioritäten können Sie erkennen, an welchen Stellen die Verbesserungsbemühungen konzentriert werden sollten (z. B. Teamfähigkeiten entwickeln, die Workload-Leistung verbessern, Kosten senken, Runbooks automatisieren oder die Überwachung ausbauen).
- Wenn sich Anforderungen ändern, aktualisieren Sie Ihre Prioritäten entsprechend.

Typische Anti-Muster:

- Sie haben sich entschieden, die Zuweisung von IP-Adressen für Ihre Produktteams zu ändern, um die Netzwerkverwaltung zu vereinfachen. Dabei haben Sie jedoch nicht mit den Mitarbeitern gesprochen. Sie wissen also nicht, welche Auswirkungen diese Änderung auf Ihre Produktteams haben wird.
- Sie implementieren ein neues Entwicklungstool, haben aber Ihre internen Kunden nicht einbezogen, um herauszufinden, ob das Tool benötigt wird oder mit den Abläufen der Kunden kompatibel ist.
- Sie implementieren ein neues Überwachungssystem, haben aber Ihre internen Kunden nicht kontaktiert, um herauszufinden, ob spezifische Überwachungs- oder Berichtsanforderungen vorliegen, die berücksichtigt werden sollten.

Vorteile der Nutzung dieser bewährten Methode: Die Bewertung und das Verständnis interner Kundenbedürfnisse liefert die Grundlage dafür, wie Sie Ihre Anstrengungen zur Bereitstellung eines geschäftlichen Mehrwerts priorisieren.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

- Verstehen Sie die geschäftlichen Anforderungen: Der geschäftliche Erfolg basiert auf gemeinsamen Zielen und der Kommunikation zwischen allen Stakeholdern, zu denen auch die Teams aus den Bereichen Geschäft, Entwicklung und Betrieb gehören.
- Überprüfen Sie die geschäftlichen Ziele, Anforderungen und Prioritäten interner Kunden: Führen Sie wichtige Stakeholder zusammen, einschließlich Geschäfts-, Entwicklungs- und Betriebsteams,

um die Ziele, Anforderungen und Prioritäten interner Kunden zu besprechen. Dadurch wird sichergestellt, dass Sie mit der betrieblichen Unterstützung vertraut sind, die erforderlich ist, um die gewünschten Geschäfts- und Kundenergebnisse zu erzielen.

- Schaffen Sie ein gemeinsames Verständnis: Sorgen Sie dafür, dass alle Beteiligten die Geschäftsfunktionen des Workloads und die Rollen der einzelnen Teams bei den Workload-spezifischen betrieblichen Abläufen kennen. Außerdem sollte bekannt sein, wie diese Faktoren die gemeinsamen Geschäftsziele mit internen und externen Kunden beeinflussen.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS11-BP03 Implementieren Sie Feedback-Schleifen](#)

OPS01-BP03 Bewertung der Governance-Anforderungen

Governance bezeichnet die Richtlinien, Regeln oder Rahmen, die ein Unternehmen nutzt, um die geschäftlichen Ziele zu erreichen. Die Governance-Anforderungen werden innerhalb Ihrer Organisation erstellt. Sie können sich darauf auswirken, welche Arten von Technologien Sie nutzen oder wie Sie Ihre Workload ausführen. Integrieren Sie die Governance-Anforderungen Ihrer Organisation in Ihren Workload. Konformität ist die Fähigkeit, nachzuweisen, dass Sie die Governance-Anforderungen implementiert haben.

Gewünschtes Ergebnis:

- Die Governance-Anforderungen werden in das Architekturdesign und den Betrieb Ihres Workloads integriert.
- Sie können nachweisen, dass Sie den Governance-Anforderungen nachkommen.
- Die Governance-Anforderungen werden regelmäßig überprüft und aktualisiert.

Typische Anti-Muster:

- Ihre Organisation verlangt Multi-Faktor-Authentifizierung für das Stammkonto. Sie haben diese Anforderung nicht implementiert und das Stammkonto wurde kompromittiert.
- Während des Entwurfs Ihres Workloads wählen Sie einen Instance-Typ, der nicht von der IT-Abteilung genehmigt wurde. Sie können Ihren Workload nicht starten und müssen ihn überarbeiten.

- Sie sind verpflichtet, über einen Plan für die Notfallwiederherstellung zu verfügen. Sie haben keinen solchen Plan erstellt und Ihr Workload ist von einem längeren Ausfall betroffen.
- Ihr Team möchte neue Instances verwenden, Ihre Governance-Anforderungen wurden jedoch nicht aktualisiert, sodass die Instances nicht zulässig sind.

Vorteile der Nutzung dieser bewährten Methode:

- Durch das Erfüllen der Governance-Anforderungen wird Ihr Workload auf die größeren Organisationsrichtlinien abgestimmt.
- Die Governance-Anforderungen spiegeln Branchenstandards und bewährte Methoden für Ihre Organisation wider.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Ermitteln Sie Governance-Anforderungen, indem Sie mit Stakeholdern und Governance-Organisationen zusammenarbeiten. Integrieren Sie die Governance-Anforderungen in Ihren Workload. Seien Sie in der Lage, nachzuweisen, dass Sie den Governance-Anforderungen nachkommen.

Kundenbeispiel

Bei AnyCompany Retail arbeitet das Cloud-Operations-Team mit Stakeholdern im gesamten Unternehmen zusammen, um die Governance-Anforderungen zu entwickeln. Sie verbieten beispielsweise den SSH Zugriff auf EC2 Amazon-Instances. Wenn Teams Systemzugriff benötigen, müssen sie AWS Systems Manager Session Manager verwenden. Das Cloud-Operations-Team aktualisiert die Governance-Anforderungen regelmäßig, sobald neue Services verfügbar sind.

Implementierungsschritte

1. Identifizieren Sie die Stakeholder für Ihren Workload, einschließlich zentralisierter Teams.
2. Arbeiten Sie mit den Stakeholdern zusammen, um Governance-Anforderungen zu ermitteln.
3. Nachdem Sie eine Liste erstellt haben, ordnen Sie die Verbesserungspunkte entsprechend der Priorität und beginnen Sie damit, sie in Ihren Workload zu implementieren.
 - a. Verwenden Sie Dienste wie [AWS Config](#) die Erstellung governance-as-code und Überprüfung der Einhaltung von Governance-Anforderungen.

- b. Wenn Sie [AWS Organizations](#) verwenden, können Sie Governance-Anforderungen mithilfe von Service-Kontrollrichtlinien (Service Control Policies, SCP) implementieren.
4. Stellen Sie Unterlagen bereit, die die Implementierung bestätigen.

Aufwand für den Implementierungsplan: Mittel. Die Implementierung fehlender Governance-Anforderungen kann dazu führen, dass Sie Ihren Workload überarbeiten müssen.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS01-BP04 Bewerten der Compliance-Anforderungen](#) – Compliance ist ähnlich wie Unternehmensführung, kommt jedoch von außerhalb des Unternehmens.

Zugehörige Dokumente:

- [AWS Leitfaden für Verwaltung und Governance zur Cloud-Umgebung](#)
- [Bewährte Methoden für Richtlinien zur AWS Organizations Servicesteuerung in einer Umgebung mit mehreren Konten](#)
- [Unternehmensführung in der AWS Cloud: Das richtige Gleichgewicht zwischen Agilität und Sicherheit](#)
- [Was sind Unternehmensführung, Risiko und Compliance \(GRC\)?](#)

Zugehörige Videos:

- [AWS Management und Unternehmensführung: Konfiguration, Compliance und Prüfung — AWS Online Tech Talks](#)
- [AWS re:INFORCE 2019: Governance für das Cloud-Zeitalter \(-R1\) DEM12](#)
- [AWS re:Invent 2020: Konformität als Code erreichen mit AWS Config](#)
- [AWS re:Invent 2020: Agile Unternehmensführung auf AWS GovCloud \(US\)](#)

Zugehörige Beispiele:

- [AWS Config Beispiele für das Konformitätspaket](#)

Zugehörige Services:

- [AWS Config](#)
- [AWS Organizations - Richtlinien zur Servicekontrolle](#)

OPS01-BP04 Bewerten der Compliance-Anforderungen

Regulatorische, branchenspezifische und interne Compliance-Anforderungen sind ein wichtiger Faktor, wenn Sie die Prioritäten Ihrer Organisation definieren. Ihr Compliance-Regelwerk hindert Sie möglicherweise daran, spezifische Technologien oder geografische Standorte zu nutzen. Wenden Sie die erforderliche Sorgfalt an, wenn keine externen Compliance-Regelwerke identifiziert sind. Erstellen Sie Audits oder Berichte, die die Compliance bestätigen.

Wenn Sie damit werben, dass Ihr Produkt bestimmte Compliance-Standards erfüllt, benötigen Sie einen internen Prozess zur kontinuierlichen Gewährleistung der Compliance. Beispiele für Compliance-Standards umfassen PCI DSS, FedRAMP und HIPAA. Die geltenden Compliance-Standards werden durch verschiedene Faktoren bestimmt, beispielsweise dadurch, welche Datentypen von der Lösung gespeichert oder gesendet werden und welche geografischen Regionen die Lösung unterstützt.

Gewünschtes Ergebnis:

- Die regulatorischen, branchenspezifischen und internen Compliance-Anforderungen werden bei der Auswahl der Architektur berücksichtigt.
- Sie können die Compliance bestätigen und Audit-Berichte erstellen.

Typische Anti-Muster:

- Teile Ihres Workloads fallen unter das Regelwerk des Payment Card Industry Data Security Standard (PCI-DSS), Ihr Workload speichert Kreditkartendaten jedoch unverschlüsselt.
- Ihren Software-Entwicklern und -Architekten ist das Compliance-Regelwerk, das Ihre Organisation einhalten muss, nicht bekannt.
- Das jährliche Audit Systems and Organizations Control (SOC2) Type II steht bevor und Sie können nicht nachweisen, dass Kontrollelemente implementiert sind.

Vorteile der Nutzung dieser bewährten Methode:

- Die Bewertung und das Verständnis der Compliance-Anforderungen für Ihren Workload liefern die Grundlage dafür, wie Sie Ihre Anstrengungen zur Bereitstellung eines geschäftlichen Mehrwerts priorisieren.
- Sie wählen die Ihrem Compliance-Regelwerk entsprechenden Standorte und Technologien.
- Indem Sie Ihren Workload so entwerfen, dass Überprüfungen möglich sind, können Sie leichter nachweisen, dass Sie das Compliance-Regelwerk einhalten.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Wenn Sie diese bewährte Methode implementieren, bedeutet dies, dass Sie Compliance-Anforderungen in den Entwurfsprozess für Ihre Architektur integrieren. Ihren Teammitgliedern ist das erforderliche Compliance-Regelwerk bekannt. Sie bestätigen Ihre Compliance mit diesem Regelwerk.

Kundenbeispiel

AnyCompany Retail speichert Kreditkarteninformationen für Kunden. Die Entwickler im Team für die Kartenspeicherung wissen, dass sie das PCI-DSS-Regelwerk einhalten müssen. Sie haben Schritte unternommen, um nachzuweisen, dass die Kreditkarteninformationen in Übereinstimmung mit dem PCI-DSS-Regelwerk sicher gespeichert und aufgerufen werden. Jedes Jahr arbeiten sie mit dem Sicherheitsteam zusammen, um die Compliance zu bestätigen.

Implementierungsschritte

1. Arbeiten Sie mit Ihrem Sicherheits- und Governance-Team zusammen, um zu ermitteln, welche branchenspezifischen, regulatorischen oder internen Compliance-Regelwerke Ihr Workload einhalten muss. Integrieren Sie die Compliance-Regelwerke in Ihren Workload.
 - a. Validieren Sie die kontinuierliche Compliance von AWS-Ressourcen mit Services wie [AWS Compute Optimizer](#) und [AWS Security Hub CSPM](#).
2. Informieren Sie Ihre Teammitglieder über die Compliance-Anforderungen, damit diese den Workload in Übereinstimmung mit den Anforderungen betreiben und weiterentwickeln können. Die Compliance-Anforderungen sollten bei architektur- und technologiebezogenen Entscheidungen berücksichtigt werden.
3. Je nach Compliance-Regelwerk müssen Sie möglicherweise einen Audit- oder Compliance-Bericht erstellen. Arbeiten Sie mit Ihrer Organisation zusammen, um diesen Prozess so weit wie möglich zu automatisieren.

- a. Nutzen Sie Services wie [AWS Audit Manager](#), um die Compliance zu validieren und Auditberichte zu erstellen.
- b. Sie können AWS-Sicherheits- und Compliance-Dokumente mit [AWS Artifact](#) herunterladen.

Aufwand für den Implementierungsplan: Mittel. Die Implementierung von Compliance-Regelwerken kann eine Herausforderung darstellen. Das Erstellen von Audit-Berichten oder Compliance-Dokumenten sorgt für zusätzlichen Aufwand.

Ressourcen

Zugehörige bewährte Methoden:

- [SEC01-BP03 Identifizieren und Validieren von Kontrollzielen](#) – Sicherheits-Kontrollziele sind ein wichtiger Teil der Gesamt-Compliance.
- [SEC01-BP06 Automatisieren von Tests und Validierung von Sicherheitskontrollen in Pipelines](#) – Validieren Sie die Sicherheitskontrollen als Teil Ihrer Pipeline. Sie können auch eine Compliance-Dokumentation für neue Änderungen erstellen.
- [SEC07-BP02 Definieren von Datenschutzkontrollen](#) – Bei vielen Compliance-Frameworks sind Datenverarbeitung und -speicherung richtlinienbasiert.
- [SEC10-BP03 Vorbereiten forensischer Funktionen](#) – Forensische Funktionen können manchmal bei der Prüfungs-Compliance verwendet werden.

Zugehörige Dokumente:

- [AWS Compliance Center](#)
- [AWS-Compliance-Ressourcen](#)
- [AWS-Risiko und -Compliance \(Whitepaper\)](#)
- [AWSModell der übergreifenden Verantwortlichkeit](#)
- [AWS-Services im Rahmen des Compliance-Programms](#)

Zugehörige Videos:

- [AWS re:Invent 2020: Achieve compliance as code using AWS Compute Optimizer](#)
- [AWS re:Invent 2.021 - Cloud compliance, assurance, and auditing](#)
- [AWS Summit ATL 2.022 - Implementing compliance, assurance, and auditing on AWS \(COP202\)](#)

Zugehörige Beispiele:

- [PCI DSS und AWS Foundational Security Best Practices auf AWS](#)

Zugehörige Services:

- [AWS Artifact](#)
- [AWS Audit Manager](#)
- [AWS Compute Optimizer](#)
- [AWS Security Hub CSPM](#)

OPS01-BP05 Bewerten Sie die Bedrohungslandschaft

Bewerten Sie Bedrohungen für das Unternehmen (z. B. Wettbewerb, Geschäftsrisiken und -verpflichtungen, operative Risiken und Bedrohungen der Informationssicherheit) und pflegen Sie aktuelle Informationen in einem Risikoregister. Berücksichtigen Sie die Auswirkungen von Risiken, wenn Sie bestimmen, auf welche Bereiche die Anstrengungen fokussiert werden sollen.

Das [Well-Architected Framework](#) legt den Schwerpunkt auf Lernen, Messen und Verbessern. Es bietet Ihnen einen konsistenten Ansatz zur Bewertung von Architekturen und zur Implementierung von Designs, die sich im Laufe der Zeit skalieren lassen. AWS bietet die [AWS Well-Architected Tool](#) Möglichkeit, Ihren Ansatz vor der Entwicklung, den Status Ihrer Workloads vor der Produktion und den Status Ihrer Workloads in der Produktion zu überprüfen. Sie können sie mit den neuesten bewährten AWS Architekturpraktiken vergleichen, den Gesamtstatus Ihrer Workloads überwachen und Einblicke in potenzielle Risiken gewinnen.

AWS Kunden haben Anspruch auf eine geführte Well-Architected-Überprüfung ihrer unternehmenskritischen Workloads, um ihre Architekturen anhand von Best Practices [zu](#) bewerten. AWS Für Kunden mit Enterprise Support wird eine [Betriebsüberprüfung \(Operations Review\)](#) angeboten. Damit haben sie die Möglichkeit, Lücken in ihrem Cloud-Ansatz aufzuzeigen.

Aufgrund der teamübergreifenden Natur dieser Überprüfungen erhalten Sie ein allgemeines Verständnis Ihrer Workloads und können erkennen, wie Team-Rollen zum Erfolg beitragen. Die bei den Überprüfungen gefundenen Punkte können Ihnen beim Festlegen Ihrer Prioritäten helfen.

[AWS Trusted Advisor](#) bietet als Tool Zugriff auf verschiedene wichtige Prüfungen, die Optimierungsempfehlungen ausgeben. Diese Informationen können Ihnen beim Festlegen Ihrer Prioritäten helfen. [Kunden mit Business und Enterprise Support](#) erhalten Zugriff auf weitere

Prüfungen in den Bereichen Sicherheit, Zuverlässigkeit, Leistung und Kostenoptimierung, die beim Festlegen von Prioritäten noch hilfreicher sind.

Gewünschtes Ergebnis:

- Du überprüfst Well-Architected und die Ergebnisse regelmäßig und reagierst entsprechend Trusted Advisor
- Sie sind über den neuesten Patch-Status Ihrer Services informiert.
- Sie kennen das Risiko und die Auswirkungen bekannter Bedrohungen und handeln entsprechend.
- Sie implementieren bei Bedarf Abhilfemaßnahmen.
- Sie kommunizieren Aktionen und Kontext.

Typische Anti-Muster:

- Sie verwenden in Ihrem Produkt eine alte Version einer Softwarebibliothek. Ihnen ist nicht bewusst, dass für die Bibliothek Sicherheitsaktualisierungen vorliegen, mit denen Probleme behoben werden, die unbeabsichtigte Auswirkungen auf Ihren Workload haben können.
- Ein Mitbewerber hat soeben eine Version seines Produkts veröffentlicht, in der viele Probleme behoben werden, die Kunden an Ihrem Produkt bemängeln. Die Behebung dieser bekannten Probleme hatte für Sie bisher keine Priorität.
- Regulierungsbehörden nehmen Unternehmen wie Ihres, die nicht den gesetzlichen Compliance-Anforderungen entsprechen, verstärkt ins Visier. Sie haben Ihre ausstehenden Compliance-Anforderungen nicht priorisiert.

Vorteile der Nutzung dieser bewährten Methode: Sie identifizieren und verstehen die Bedrohungen für Ihr Unternehmen und Ihren Workload und können daher besser bestimmen, welche Bedrohungen angegangen werden müssen, wo die Prioritäten liegen und welche Ressourcen dafür erforderlich sind.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

- Bewerten der Bedrohungsszenarien: Bewerten Sie Bedrohungen für das Unternehmen (z. B. Konkurrenz, Geschäftsrisiken und -verpflichtungen, operative Risiken und Bedrohungen der Informationssicherheit), damit Sie die jeweiligen Auswirkungen berücksichtigen können, wenn Sie bestimmen, auf welche Bereiche die operativen Anstrengungen konzentriert werden sollten.

- [Aktuelle AWS -Sicherheitsmitteilungen](#)
- [AWS Trusted Advisor](#)
- Verwalten eines Bedrohungsmodells: Erstellen und verwalten Sie ein Bedrohungsmodell, in dem potenzielle Bedrohungen, geplante und vorhandene Maßnahmen und deren Priorität festgehalten werden. Untersuchen Sie, wie wahrscheinlich es ist, dass sich Bedrohungen als Vorfälle äußern, wie hoch die Kosten für die Wiederherstellung nach diesen Vorfällen sind, welche Schäden zu erwarten sind und wie viel es kostet, diese Vorfälle zu verhindern. Überarbeiten Sie die Prioritäten, wenn sich der Inhalt des Bedrohungsmodells ändert.

Ressourcen

Zugehörige bewährte Methode:

- [SEC01-BP07 Identifizieren Sie Bedrohungen und priorisieren Sie Abhilfemaßnahmen mithilfe eines Bedrohungsmodells](#)

Zugehörige Dokumente:

- [AWS Cloud -Compliance](#)
- [Aktuelle AWS -Sicherheitsmitteilungen](#)
- [AWS Trusted Advisor](#)

Zugehörige Videos:

- [AWS re:Inforce 2023 – Tool für eine bessere Bedrohungsmodellierung](#)

OPS01-BP06 Bewerten von Kompromissen und Abwägen der Vorteile und Risiken

Konkurrierende Interessen mehrerer Parteien können eine Herausforderung darstellen, wenn es darum geht, Anstrengungen zu priorisieren, Fähigkeiten aufzubauen und Ergebnisse zu erzielen, die auf die Geschäftsstrategien abgestimmt sind. So können Sie möglicherweise aufgefordert werden, die Markteinführung neuer Features zu beschleunigen, anstatt die Kosten für die IT-Infrastruktur zu optimieren. Dies kann dazu führen, dass die Interessen zweier Parteien miteinander in Widerspruch stehen. In solchen Situationen muss eine höhere Stelle hinzugezogen werden, um eine Entscheidung zur Lösung des Konflikts zu treffen. Daten sind erforderlich, um den Entscheidungsprozess von emotionalen Komponenten zu befreien.

Ähnliche Herausforderungen können auf taktischer Ebene auftreten. Beispielsweise kann die Wahl zwischen relationalen oder nicht relationalen Datenbanktechnologien erhebliche Auswirkungen auf den Betrieb einer Anwendung haben. Daher ist es wichtig, die voraussichtlichen Ergebnisse verschiedener Entscheidungen zu verstehen.

AWS kann Ihnen helfen, Ihre Teams über AWS und die verfügbaren Services zu schulen, sodass alle Mitarbeiter wissen, welche Auswirkungen ihre Entscheidungen auf Ihre Workload haben können. Nutzen Sie bei der Schulung Ihrer Teams die vom [Support](#) ([AWS Knowledge Center](#), [AWS-Diskussionsforen](#) und [Support Center](#)) bereitgestellten Ressourcen und [AWS-Dokumente](#). Bei weiteren Fragen wenden Sie sich bitte an Support.

AWS stellt in der [Amazon Builders' Library](#) auch bewährte betriebliche Methoden und Muster vor. Eine Vielzahl weiterer nützlicher Informationen finden Sie im [AWS-Blog](#) und im [offiziellen AWS-Podcast](#).

Gewünschtes Ergebnis: Sie verfügen über ein klar definiertes Governance-Framework zur Entscheidungsfindung, um wichtige Entscheidungen auf jeder Ebene in Ihrem Cloud-Bereitstellungsunternehmen zu erleichtern. Dieses Framework umfasst Features wie ein Risikoregister, definierte Rollen mit Entscheidungsbefugnissen und definierte Modelle für die einzelnen Entscheidungsebenen. Dieses Framework legt im Voraus fest, wie Konflikte gelöst werden, welche Daten präsentiert werden müssen und wie Optionen priorisiert werden, sodass Sie einmal gefasste Beschlüsse sofort umsetzen können. Das Framework zur Entscheidungsfindung beinhaltet einen standardisierten Ansatz zur Überprüfung und Abwägung der Vorteile und Risiken einzelner Entscheidungen, um die Tragweite etwaiger Kompromisse abzuschätzen. Dazu können externe Faktoren gehören wie die Einhaltung gesetzlicher Vorschriften.

Typische Anti-Muster:

- Ihre Investoren fordern, dass Sie die Compliance mit Payment Card Industry Data Security Standards (PCI DSS) nachweisen. Sie denken nicht über einen möglichen Kompromiss zwischen der Erfüllung dieser Anfrage und der Fortsetzung Ihrer derzeitigen Entwicklungsaktivitäten nach. Stattdessen fahren Sie mit der Entwicklung fort, ohne einen Compliance-Nachweis zu erbringen. Ihre Investoren beenden die Unterstützung Ihres Unternehmens, da sie Bedenken bezüglich der Sicherheit Ihrer Plattform und ihrer Investitionen haben.
- Sie haben sich entschieden, eine Bibliothek einzubinden, die einer Ihrer Entwickler „im Internet entdeckt“ hat. Sie haben keine Bewertung der Risiken durchgeführt, die die Einführung dieser Bibliothek aus einer unbekannten Quelle bergen kann, und wissen nicht, ob sie Schwachstellen oder schädlichen Code enthält.

- Die ursprüngliche geschäftliche Begründung für Ihre Migration basierte auf der Modernisierung von 60 % Ihrer Anwendungsworkloads. Aufgrund technischer Schwierigkeiten wurde jedoch beschlossen, nur 20 % zu modernisieren. Dies führte langfristig zu einer Reduzierung der geplanten Leistungen, zu einem erhöhten Aufwand für die Infrastrukturteams bei der manuellen Wartung von Legacy-Systemen und zu einer stärkeren Abhängigkeit von der Entwicklung neuer Fähigkeiten in Ihren Infrastrukturteams, die diese Änderung nicht geplant hatten.

Vorteile der Nutzung dieser bewährten Methode: Vollständige Abstimmung und Unterstützung der Geschäftsprioritäten auf Vorstandsebene, Verständnis der Erfolgsrisiken, Treffen fundierter Entscheidungen und angemessenes Handeln, wenn Risiken die Erfolgschancen beeinträchtigen. Indem Sie die Auswirkungen und Konsequenzen Ihrer Entscheidungen verstehen, können Sie Ihre Optionen priorisieren und Führungskräfte schneller zu einer Einigung bringen, was zu besseren Geschäftsergebnissen führt. Wenn Sie die Vorteile Ihrer Entscheidungen erkennen und sich der Risiken für Ihre Organisation bewusst sind, können Sie datengestützte Entscheidungen treffen, anstatt sich auf Anekdoten verlassen zu müssen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Die Abwägung von Nutzen und Risiken sollte von einem Leitungsorgan übernommen werden, das die Anforderungen für wichtige Entscheidungen festlegt. Sie möchten, dass Entscheidungen basierend auf ihrem Nutzen für die Organisation getroffen und priorisiert werden und die damit verbundenen Risiken bekannt sind. Präzise Informationen bilden die Grundlage für die Entscheidungen Ihrer Organisation. Diese sollten auf soliden Messungen beruhen und durch branchenübliche Verfahren der Kosten-Nutzen-Analyse definiert werden. Damit Entscheidungen auf diese Art getroffen werden können, müssen Sie ein Gleichgewicht zwischen zentralisierter und dezentralisierter Autorität herstellen. Es gibt immer einen Kompromiss. Daher ist es wichtig zu verstehen, wie sich jede Entscheidung auf definierte Strategien und angestrebte Geschäftsergebnisse auswirkt.

Implementierungsschritte

1. Formalisieren Sie die Verfahren zur Leistungsmessung innerhalb eines ganzheitlichen Cloud-Governance-Frameworks.
 - a. Bringen Sie die zentrale Kontrolle der Entscheidungsfindung in Einklang mit konkreten dezentralen Entscheidungsbefugnissen.
 - b. Machen Sie sich bewusst, dass nicht für jeden Beschluss aufwendige Entscheidungsprozesse vonnöten sind, da sie Sie verlangsamen können.

- c. Integrieren Sie externe Faktoren in Ihren Entscheidungsprozess (wie Compliance-Anforderungen).
2. Richten Sie ein gemeinsames Framework zur Entscheidungsfindung für verschiedene Entscheidungsebenen ein, in dem festgelegt ist, wer Entscheidungen bei widersprüchlichen Interessen trifft.
 - a. Zentralisieren Sie einseitige Entscheidungen, die irreversibel sein könnten.
 - b. Lassen Sie leicht revidierbare Entscheidungen von Führungskräften auf niedrigerer Ebene treffen.
3. Machen Sie sich mit den Nutzen und Risiken vertraut und wägen Sie sie ab. Wägen Sie den Nutzen von Entscheidungen gegen die damit einhergehenden Risiken ab.
 - a. Ermitteln von Vorteilen: Ermitteln Sie die Vorteile auf Basis der geschäftlichen Ziele, Anforderungen und Prioritäten. Beispiele hierfür sind die Auswirkungen auf den Business Case, die Markteinführungszeit, Sicherheit, Zuverlässigkeit, Leistung und Kosten.
 - b. Ermitteln von Risiken: Ermitteln Sie die Risiken auf Basis der geschäftlichen Ziele, Anforderungen und Prioritäten. Zu diesen Prioritäten zählen beispielsweise eine kurze Markteinführungszeit, Sicherheit, Zuverlässigkeit, Leistung und Kosten.
 - c. Abwägen von Vorteilen und Risiken und Treffen fundierter Entscheidungen: Bestimmen Sie die Auswirkungen von Vorteilen und Risiken anhand der Ziele, Anforderungen und Prioritäten der wichtigsten Beteiligten, zu denen auch Geschäfts-, Entwicklungs- und Betriebsteams zählen. Bewerten Sie den Wert eines Vorteils anhand der Wahrscheinlichkeit, dass sich das Risiko tatsächlich bewahrheitet, sowie der Kosten der jeweiligen Auswirkungen. Eine schnellere Markteinführung zu Lasten der Zuverlässigkeit könnte beispielsweise einen Wettbewerbsvorteil bedeuten. Wenn jedoch Probleme mit der Zuverlässigkeit auftreten, kann dies zu einer verringerten Betriebszeit führen.
4. Setzen Sie wichtige Entscheidungen programmatisch um, um die Einhaltung von Compliance-Anforderungen zu automatisieren.
5. Nutzen Sie branchenübliche Frameworks und Funktionen wie Value Stream Analysis und LEAN, um die aktuelle Leistung und Geschäftsmetriken abzubilden und Iterationen der Fortschritte zur Verbesserung dieser Metriken zu definieren.

Aufwand für den Implementierungsplan: Mittel-Hoch

Ressourcen

Zugehörige bewährte Methoden:

- [OPS01-BP05 Bewerten der Bedrohungsszenarien](#)

Zugehörige Dokumente:

- [Elemente der Day-1-Kultur von Amazon | Hochwertige und schnelle Entscheidungen treffen](#)
- [Cloud-Governance](#)
- [Verwaltungs- und Governance-Cloud-Umgebung](#)
- [Governance in der Cloud und im digitalen Zeitalter: Teil eins und Teil zwei](#)

Zugehörige Videos:

- [Podcast | Jeff Bezos | So trifft man Entscheidungen](#)

Zugehörige Beispiele:

- [Mithilfe von Daten fundierte Entscheidungen treffen \(The DevOps Sagas\)](#)
- [Verwendung von Wertstromanalysen für die Entwicklung, um Hindernisse für DevOps-Ergebnisse zu identifizieren](#)

OPS 2. Wie strukturieren Sie Ihr Unternehmen, um die gewünschten Geschäftsergebnisse zu erzielen?

Ihre Teams müssen ihre Rolle beim Erreichen von Geschäftsergebnissen verstehen. Teams sollten ihre Rolle für den Erfolg anderer Teams und die Rolle anderer Teams für ihren Erfolg verstehen und gemeinsame Ziele haben. Indem sie die Konzepte Verantwortlichkeit und Zuständigkeit verstehen und wissen, wie Entscheidung getroffen werden und wer dazu berechtigt ist, können ihre Anstrengungen fokussiert und der Nutzen Ihrer Teams maximiert werden.

Best Practices

- [OPS02-BP01 Ressourcen haben Eigentümer identifiziert](#)
- [OPS02-BP02 Prozesse und Verfahren haben feste Besitzer](#)
- [OPS02-BP03 Betriebsaktivitäten haben feste Besitzer, die für ihre Leistung verantwortlich sind](#)
- [OPS02-BP04 Es gibt Mechanismen zur Verwaltung von Verantwortlichkeiten und Zuständigkeiten](#)
- [OPS02-BP05 Mechanismen zum Anfordern von Ergänzungen, Änderungen und Ausnahmen sind vorhanden](#)

- [OPS02-BP06 Zuständigkeiten zwischen Teams werden vordefiniert oder ausgehandelt](#)

OPS02-BP01 Ressourcen haben Eigentümer identifiziert

Die Ressourcen für Ihren Workload müssen für die Änderungskontrolle, die Fehlerbehebung und andere Funktionen feste Verantwortliche haben. Verantwortliche werden für Workloads, Konten, Infrastruktur, Plattformen und Anwendungen zugewiesen. Die Verantwortlichkeit wird mit Tools wie einem Zentralverzeichnis oder Metadaten zu Ressourcen erfasst. Der Unternehmenswert der Komponenten bestimmt, welche Prozesse und Verfahren auf diese angewendet werden.

Gewünschtes Ergebnis:

- Mithilfe von Metadaten oder einem Zentralverzeichnis werden feste Verantwortliche für die Ressourcen identifiziert.
- Die Teammitglieder können erkennen, wer für eine bestimmte Ressource verantwortlich ist.
- Konten haben wenn möglich einen festen Verantwortlichen.

Typische Anti-Muster:

- Die alternativen Kontakte für Sie AWS-Konten sind nicht eingetragen.
- Die Ressourcen sind nicht mit Tags markiert, die kennzeichnen, wer dafür verantwortlich ist.
- Sie haben eine ITSM Warteschlange ohne E-Mail-Zuordnung.
- Zwei Teams haben sich überschneidende Verantwortlichkeit für einen wichtigen Teil der Infrastruktur.

Vorteile der Nutzung dieser bewährten Methode:

- Dank der zugewiesenen Verantwortlichkeit ist die Änderungskontrolle ganz einfach.
- Wenn Probleme auftreten, können die richtigen Verantwortlichen einbezogen werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Definieren Sie, was Verantwortlichkeit für die Ressourcen-Anwendungsfälle in Ihrer Umgebung bedeutet. Verantwortlichkeit kann bedeuten, Änderungen an der Ressource zu beaufsichtigen, die Ressource während der Fehlerbehebung zu unterstützen oder die finanzielle Verantwortung zu

tragen. Legen Sie Verantwortliche für Ressourcen fest und dokumentieren Sie diese. Die Angaben sollten den Namen, die Kontaktinformationen, die Organisation und das Team beinhalten.

Kundenbeispiel

AnyCompany Der Einzelhandel definiert Eigenverantwortung als das Team oder die Einzelperson, die für Änderungen verantwortlich ist und Ressourcen unterstützt. Sie nutzen AWS Organizations, um ihre zu verwalten AWS-Konten. Die alternativen Kontakte für die Konten werden mit Gruppenpostfächern konfiguriert. Jede ITSM Warteschlange ist einem E-Mail-Alias zugeordnet. Mithilfe von Tags wird angegeben, wem AWS Ressourcen gehören. Für andere Plattformen und Infrastruktur gibt es eine Wiki-Seite, auf der die Verantwortlichkeiten und die Kontaktinformationen angegeben sind.

Implementierungsschritte

1. Beginnen Sie damit, die Verantwortlichkeiten für Ihre Organisation zu definieren. Verantwortlichkeit kann bedeuten, wer für das Risiko für die Ressource oder für Änderungen an der Ressource verantwortlich ist oder wer die Ressource im Fall einer Fehlerbehebung unterstützt. Verantwortlichkeit kann auch die finanzielle oder administrative Verantwortlichkeit für die Ressource umfassen.
2. Nutzen Sie [AWS Organizations](#) zur Verwaltung von Konten. Sie können die alternativen Kontakte für Ihre Konten zentral verwalten.
 - a. Durch die Verwendung von E-Mail-Adressen und Telefonnummern des Unternehmens als Kontaktdaten können Sie auch dann auf sie zugreifen, wenn die Personen, zu denen sie gehören, nicht mehr Teil Ihrer Organisation sind. Erstellen Sie beispielsweise separate E-Mail-Verteilerlisten für die Abrechnung, die Produktion und die Sicherheit und konfigurieren Sie sie in jedem aktiven AWS-Konto als Abrechnungs-, Sicherheits- und Produktionskontakte. Mehrere Personen erhalten AWS Benachrichtigungen und können antworten, auch wenn jemand im Urlaub ist, die Rolle wechselt oder das Unternehmen verlässt.
 - b. Wenn ein Konto nicht von [AWS Organizations](#) verwaltet wird, helfen alternative Kontaktkontakte AWS dabei, bei Bedarf Kontakt mit den entsprechenden Personen aufzunehmen. Konfigurieren Sie die alternativen Kontakte für ein Konto so, dass sie auf eine Gruppe verweisen, und nicht auf eine Einzelperson.
3. Verwenden Sie Tags, um Eigentümer von AWS Ressourcen zu identifizieren. Sie können die Verantwortlichen und ihre Kontaktdaten in verschiedenen Tags angeben.
 - a. Sie können [AWS Config](#)-Regeln verwenden, um durchzusetzen, dass Ressourcen über die erforderlichen Eigentümerkennungen verfügen.

- b. Ausführliche Anleitungen zur Entwicklung einer Tagging-Strategie für Ihr Unternehmen finden Sie im [Whitepaper „Bewährte AWS -Tagging-Methoden“](#).
4. Verwenden Sie [Amazon Q Business](#), einen Konversationsassistenten, der auf generativer KI basiert, um die Produktivität Ihrer Mitarbeiter zu steigern, Fragen zu beantworten und Aufgaben auf der Grundlage von Informationen in Ihren Unternehmenssystemen zu erledigen.
 - a. Verbinden Sie Amazon Q Business mit der Datenquelle Ihres Unternehmens. Amazon Q Business bietet vorgefertigte Konnektoren für über 40 unterstützte Datenquellen, darunter Amazon Simple Storage Service (Amazon S3), Microsoft SharePoint, Salesforce und Atlassian Confluence. Weitere Informationen finden Sie unter [Amazon Q Business-Konnektoren](#).
5. Erstellen Sie für andere Ressourcen, Plattformen und Infrastruktur eine Dokumentation mit Informationen zur jeweiligen Verantwortlichkeit. Diese sollte für alle Teammitglieder zugänglich sein.

Aufwand für den Implementierungsplan: Niedrig. Nutzen Sie Kontokontaktinformationen und Tags, um die Inhaberschaft von Ressourcen zuzuweisen. AWS Für andere Ressourcen können Sie etwas so Einfaches wie eine Tabelle in einem Wiki verwenden, um Eigentums- und Kontaktinformationen aufzuzeichnen, oder ein ITSM Tool verwenden, um die Eigentumsverhältnisse zuzuordnen.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS02-BP02 Für Prozesse und Verfahren wurden die Verantwortlichen identifiziert](#)
- [OPS02-BP04 Es gibt Mechanismen zur Verwaltung von Verantwortlichkeiten und Eigenverantwortung](#)

Zugehörige Dokumente:

- [AWS -Kontoverwaltung – Aktualisieren der Kontaktinformationen](#)
- [AWS Organizations - Aktualisierung alternativer Ansprechpartner in Ihrer Organisation](#)
- [Bewährte AWS -Tagging-Methoden \(Whitepaper\)](#)
- [Entwickeln Sie private und sichere generative KI-Apps für Unternehmen mit Amazon Q Business and AWS IAM Identity Center](#)
- [Amazon Q Business \(jetzt allgemein verfügbar\) ermöglicht die Steigerung der Produktivität der Mitarbeiter mithilfe von generativer KI](#)

- [AWS Cloud Operations & Migrations Blog — Implementierung automatisierter und zentralisierter Tagging-Steuerungen mit und AWS ConfigAWS Organizations](#)
- [AWS Sicherheitsblog — Erweitern Sie Ihre Pre-Commit-Hooks mit AWS CloudFormation Guard](#)
- [AWS DevOps Blog — Integration AWS CloudFormation Guard in CI/CD-Pipelines](#)

Zugehörige Workshops:

- [AWS -Workshop – Tagging](#)

Zugehörige Beispiele:

- [AWS-Config-Regeln - Amazon EC2 mit den erforderlichen Tags und gültigen Werten](#)

Zugehörige Services:

- [AWS-Config-Regeln - erforderliche Tags](#)
- [AWS Organizations](#)

OPS02-BP02 Prozesse und Verfahren haben feste Besitzer

Verschaffen Sie sich einen Überblick darüber, wer für die Definition einzelner Prozesse und Verfahren zuständig ist, warum diese spezifischen Prozesse und Verfahren verwendet werden und warum diese Zuständigkeit besteht. Wenn Sie wissen, warum bestimmte Prozesse und Verfahren verwendet werden, können Sie Verbesserungsmöglichkeiten identifizieren.

Gewünschtes Ergebnis: Ihre Organisation verfügt über gut definierte und verwaltete Prozesse und Verfahren für betriebliche Aufgaben. Der Prozess und die Verfahren werden an einem zentralen Ort gespeichert und stehen Ihren Teammitgliedern zur Verfügung. Prozesse und Verfahren werden regelmäßig aktualisiert, wobei die Zuständigkeit eindeutig zugewiesen wird. Wo möglich, werden Skripte, Vorlagen und Automatisierungsdokumente als Code implementiert.

Typische Anti-Muster:

- Prozesse sind nicht dokumentiert. Möglicherweise sind fragmentierte Skripte auf isolierten Bedienerarbeitsplätzen vorhanden.
- Das Wissen über den Umgang mit Skripten wird von wenigen Personen oder informell als Teamwissen vermittelt.

- Ein veralteter Prozess muss aktualisiert werden, aber die Zuständigkeit für die Aktualisierung ist unklar, und der ursprüngliche Autor gehört nicht mehr zur Organisation.
- Prozesse und Skripte sind nicht auffindbar und daher nicht sofort verfügbar, wenn sie benötigt werden (z. B. als Reaktion auf einen Vorfall).

Vorteile der Nutzung dieser bewährten Methode:

- Prozesse und Verfahren unterstützen Sie bei der Bewältigung Ihrer Workloads.
- Neue Teammitglieder werden schneller handlungsfähig.
- Die Zeit bis zur Behebung von Vorfällen wird reduziert.
- Verschiedene Teammitglieder (und Teams) können dieselben Prozesse und Verfahren auf einheitliche Weise verwenden.
- Teams können ihre Prozesse durch wiederholbare Prozesse skalieren.
- Standardisierte Prozesse und Verfahren tragen dazu bei, die Auswirkungen der Übertragung von Workload-Verantwortlichkeiten zwischen Teams abzumildern.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

- Prozesse und Verfahren haben feste Besitzer, die für ihre Definition verantwortlich sind.
 - Identifizieren Sie die Betriebsaktivitäten, die zur Unterstützung Ihrer Workloads durchgeführt werden. Dokumentieren Sie diese Aktivitäten an einem auffindbaren Ort.
 - Legen Sie die Person oder Personen fest, die für die Spezifikation einer Aktivität verantwortlich sind. Sie sind dafür verantwortlich, sicherzustellen, dass die Aktivität von einem ausreichend qualifizierten Teammitglied durchgeführt wird, das die entsprechenden Berechtigungen, Zugriffsrechte und Tools hat. Wenn bei der Durchführung dieser Aktivität Probleme auftreten, sind die zuständigen Teammitglieder dafür verantwortlich, detailliertes Feedback bereitzustellen, das für die Verbesserung der Aktivität erforderlich ist.
- Erfassen Sie die Zuständigkeit in den Metadaten des Aktivitätsartefakts durch Services wie AWS Systems Manager, durch Dokumente und AWS Lambda. Erfassen Sie die Ressourcenzuständigkeit mithilfe von Tags oder Ressourcengruppen und geben Sie Zuständigkeits- und Kontaktinformationen an. Verwenden Sie AWS Organizations, um Markierungsrichtlinien zu erstellen sowie Zuständigkeits- und Kontaktinformationen zu erfassen.

- Mit der Zeit sollten diese Verfahren so weiterentwickelt werden, dass sie als Code ausgeführt werden können, sodass weniger menschliche Eingriffe erforderlich sind.
- Erwägen Sie beispielsweise AWS Lambda-Funktionen, CloudFormation-Vorlagen oder AWS Systems Manager-Automatisierungsdokumente.
- Führen Sie die Versionskontrolle in den entsprechenden Repositories durch.
- Fügen Sie geeignetes Ressourcen-Tagging hinzu, damit Eigentümer und Dokumentation leicht identifiziert werden können.

Kundenbeispiel

AnyCompany Retail legt fest, dass das Team oder die Person, die für die Prozesse einer Anwendung oder einer Gruppe von Anwendungen (die gemeinsame architektonische Praktiken und Technologien nutzen) zuständig ist, der Besitzer ist. Zunächst werden der Prozess und die Verfahren in Form von schrittweisen Anleitungen im Dokumentenverwaltungssystem dokumentiert, die über Tags für das AWS-Konto, das die Anwendung hostet, und für bestimmte Ressourcengruppen innerhalb des Kontos auffindbar sind. Das Unternehmen verwendet AWS Organizations für die Verwaltung seiner AWS-Konten. Im Laufe der Zeit werden diese Prozesse in Code umgewandelt und Ressourcen werden mithilfe von Infrastructure as Code (z. B. CloudFormation oder AWS Cloud Development Kit (AWS CDK)-Vorlagen) definiert. Die Betriebsprozesse werden zu Automatisierungsdokumenten in AWS Systems Manager- oder AWS Lambda-Funktionen, die als geplante Aufgaben, als Reaktion auf Ereignisse wie AWS Systems Manager-Alarme oder AWS EventBridge-Ereignisse oder durch Anfragen innerhalb einer IT-Servicemanagement-Plattform (ITSM) gestartet werden können. Alle Prozesse sind mit Tags versehen, um die Zuständigkeit zu identifizieren. Die Dokumentation für die Automatisierung und den Prozess wird auf den Wiki-Seiten verwaltet, die vom Code-Repository für den Prozess generiert werden.

Implementierungsschritte

1. Dokumentieren Sie die bestehenden Prozesse und Verfahren.
 - a. Überprüfen Sie sie und halten Sie sie auf dem neuesten Stand.
 - b. Identifizieren Sie einen Besitzer für jeden Prozess und jede Prozedur.
 - c. Stellen Sie sie unter Versionskontrolle.
 - d. Wenn möglich, nutzen Sie Prozesse und Verfahren für Workloads und Umgebungen mit gemeinsamen Architekturentwürfen.
2. Richten Sie Mechanismen für Feedback und Verbesserung ein.
 - a. Definieren Sie Richtlinien dafür, wie oft Prozesse überprüft werden sollten.

- b. Definieren Sie Prozesse für Prüfende und Genehmigende.
 - c. Implementieren Sie Probleme oder eine Ticket-Warteschlange, um Feedback zu geben und zu verfolgen.
 - d. Wo immer es möglich ist, sollten Prozesse und Verfahren vorab von einem Gremium zur Genehmigung von Änderungen genehmigt und in eine Risikoklasse eingestuft werden.
3. Stellen Sie sicher, dass Prozesse und Verfahren für diejenigen, die sie ausführen müssen, zugänglich und auffindbar sind.
- a. Verwenden Sie Tags, um anzugeben, wo der Prozess und die Verfahren für die Workload aufgerufen werden können.
 - b. Verwenden Sie aussagekräftige Fehler- und Ereignismeldungen, um die geeigneten Prozesse oder Verfahren zur Behebung eines Problems anzugeben.
 - c. Verwenden Sie Wikis und Dokumentenmanagement und machen Sie Prozesse und Verfahren organisationsweit durchsuchbar.
4. Verwenden Sie [Amazon Q Business](#), einen Konversationsassistenten, der auf generativer KI basiert, um die Produktivität Ihrer Mitarbeiter zu steigern, Fragen zu beantworten und Aufgaben auf der Grundlage von Informationen in Ihren Unternehmenssystemen zu erledigen.
- a. Verbinden Sie Amazon Q Business mit der Datenquelle Ihres Unternehmens. Amazon Q Business bietet vorgefertigte Konnektoren für über 40 unterstützte Datenquellen, darunter Amazon S3, Microsoft SharePoint, Salesforce und Atlassian Confluence. Weitere Informationen finden Sie unter [Amazon-Q-Konnektoren](#).
5. Automatisieren Sie dies gegebenenfalls.
- a. Automatisierungen sollten entwickelt werden, wenn Services und Technologien eine API bereitstellen.
 - b. Informieren Sie sich angemessen über Prozesse. Entwickeln Sie die nötigen Benutzerszenarien und Anforderungen, um diese Prozesse zu automatisieren.
 - c. Messen Sie die erfolgreiche Nutzung Ihrer Prozesse und Verfahren und erstellen Sie Probleme oder Tickets, um eine iterative Verbesserung zu unterstützen.

Aufwand für den Implementierungsplan: Mittel

Ressourcen

Zugehörige bewährte Methoden:

- [OPS02-BP01 Ressourcen haben feste Verantwortliche](#)

- [OPS02-BP04 Es gibt Mechanismen zur Verwaltung von Verantwortlichkeiten und Zuständigkeiten](#)
- [OPS11-BP04 Wissensmanagement](#)

Zugehörige Dokumente:

- [AWS Whitepaper – Einführung in DevOps in AWS](#)
- [AWS-Whitepaper – Bewährte Methoden für das Taggen von AWS-Ressourcen](#)
- [AWS-Whitepaper – Organisieren Sie Ihre AWS-Umgebung mit mehreren Konten](#)
- [AWS Cloud Operations and Migrations Blog – Amazon Q Business zur Optimierung Ihrer Betriebsabläufe verwenden](#)
- [AWS Cloud Operations & Migrations Blog – Aufbau einer Cloud-Automatisierungspraxis für betriebliche Exzellenz: Bewährte Methoden von AWS Managed Services](#)
- [AWS Cloud Operations & Migrations Blog – Implementierung automatisierter und zentraler Tagging-Kontrollen mit AWS Config und AWS Organizations](#)
- [AWS Sicherheits-Blog – Pre-Commit-Hooks mit erweitern AWS CloudFormation Guard](#)
- [AWS-DevOps-Blog – Integration von AWS CloudFormation Guard in CI/CD-Pipelines](#)

Zugehörige Workshops:

- [AWS Well-Architected-Workshop zur betrieblichen Exzellenz](#)
- [AWS Workshop – Tagging](#)

Zugehörige Videos:

- [Automatisierung von IT-Abläufen in AWS](#)
- [AWS re:Invent 2020 – Automatisierung aller Abläufe mit AWS Systems Manager](#)
- [AWS re:Inforce 2022 – Automatisierung der Patch-Verwaltung und -Compliance mit AWS \(NIS306\)](#)
- [Supports You – Tiefer Einblick in AWS Systems Manager](#)

Zugehörige Services:

- [AWS Systems Manager - Automatisierung](#)
- [AWS Service Management Connector](#)

OPS02-BP03 Betriebsaktivitäten haben feste Besitzer, die für ihre Leistung verantwortlich sind

Verschaffen Sie sich einen Überblick darüber, wer für spezifische Aktivitäten in festgelegten Workloads verantwortlich ist und warum diese Zuständigkeit besteht. Wenn Sie wissen, wer für die Durchführung von Aktivitäten verantwortlich ist, können Sie nachvollziehen, wer die Aktivität durchführen, das Ergebnis validieren und dem Besitzer der Aktivität Feedback geben wird.

Gewünschtes Ergebnis:

Ihre Organisation definiert klar die Verantwortlichkeiten, um bestimmte Aktivitäten anhand definierter Workloads durchzuführen und auf Ereignisse zu reagieren, die durch die Workloads verursacht werden. Die Organisation dokumentiert die Zuständigkeit für Prozesse und deren Erfüllung und macht diese Informationen auffindbar. Sie überprüfen und aktualisieren die Zuständigkeiten, wenn organisatorische Änderungen stattfinden, und die Teams verfolgen und messen die Leistung der Aktivitäten zur Identifizierung von Fehlern und Ineffizienzen. Sie implementieren Feedback-Mechanismen, um Fehler und Verbesserungen nachzuverfolgen und iterative Verbesserungen zu unterstützen.

Typische Anti-Muster:

- Sie dokumentieren keine Verantwortlichkeiten.
- Fragmentierte Skripte sind auf isolierten Bedienerarbeitsplätzen vorhanden. Nur wenige Personen wissen, wie man sie verwendet, oder bezeichnen sie informell als Teamwissen.
- Ein veralteter Prozess muss aktualisiert werden, aber niemand weiß, wer für den Prozess zuständig ist, und der ursprüngliche Autor gehört nicht mehr zur Organisation.
- Prozesse und Skripte sind nicht auffindbar und nicht sofort verfügbar, wenn sie benötigt werden (z. B. als Reaktion auf einen Vorfall).

Vorteile der Nutzung dieser bewährten Methode:

- Sie wissen, wer die verantwortliche Person für die Durchführung einer Aktivität ist, wer benachrichtigt werden muss, wenn eine Aktion erforderlich ist, und wer die Aktion ausführen, das Ergebnis validieren und dem Besitzer der Aktivität Feedback geben wird.
- Prozesse und Verfahren unterstützen Sie bei der Bewältigung Ihrer Workloads.
- Neue Teammitglieder werden schneller handlungsfähig.
- Sie reduzieren die Zeit, die zur Behebung von Vorfällen benötigt wird.

- Verschiedene Teams verwenden dieselben Prozesse und Verfahren, um Aufgaben auf einheitliche Weise auszuführen.
- Teams können ihre Prozesse durch wiederholbare Prozesse skalieren.
- Standardisierte Prozesse und Verfahren tragen dazu bei, die Auswirkungen der Übertragung von Workload-Verantwortlichkeiten zwischen Teams abzumildern.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Um mit der Definition von Verantwortlichkeiten zu beginnen, beginnen Sie mit der vorhandenen Dokumentation, wie Zuständigkeitsmatrizen, Prozessen und Verfahren, Rollen und Verantwortlichkeiten sowie Tools und Automatisierung. Überprüfen und besprechen Sie die Verantwortlichkeiten für dokumentierte Prozesse. Ermitteln Sie gemeinsam mit den Teams, ob Abweichungen zwischen den Verantwortlichkeiten und Prozessen für Dokumente vorliegen. Besprechen Sie die angebotenen Dienstleistungen mit internen Kunden dieses Teams, um unterschiedliche Erwartungen zwischen den Teams zu identifizieren.

Analysieren und beheben Sie die Diskrepanzen. Identifizieren Sie Verbesserungsmöglichkeiten und suchen Sie nach häufig nachgefragten, ressourcenintensiven Aktivitäten, bei denen es sich in der Regel um gute Kandidaten für Verbesserungen handelt. Informieren Sie sich über bewährte Methoden, Muster und verbindliche Anleitungen, um Verbesserungen zu vereinfachen und zu standardisieren. Erfassen Sie Verbesserungsmöglichkeiten und verfolgen Sie die Verbesserungen bis zur Fertigstellung.

Mit der Zeit sollten diese Verfahren so weiterentwickelt werden, dass sie als Code ausgeführt werden, sodass weniger menschliche Eingriffe erforderlich sind. Beispielsweise können Verfahren als AWS Lambda-Funktionen, CloudFormation-Vorlagen oder AWS Systems Manager-Automatisierungsdokumente initiiert werden. Stellen Sie sicher, dass diese Verfahren in den entsprechenden Repositories versionskontrolliert sind und ein geeignetes Ressourcen-Tagging enthalten, sodass die Teams die Eigentümer und die Dokumentation leicht identifizieren können. Dokumentieren Sie die Verantwortung für die Durchführung der Aktivitäten und überwachen Sie dann die Automatisierungen, um sicherzustellen, dass sie erfolgreich initiiert und ausgeführt werden und dass die gewünschten Ergebnisse erzielt werden.

Kundenbeispiel

AnyCompany Retail legt fest, dass das Team oder die Person, die für die Prozesse einer Anwendung oder einer Gruppe von Anwendungen (die gemeinsame architektonische Praktiken und Technologien

nutzen) zuständig ist, der Besitzer ist. Zunächst dokumentiert das Unternehmen die Prozesse und Verfahren als schrittweise Anleitungen im Dokumentenmanagementsystem. Es macht die Verfahren mithilfe von Tags auf dem AWS-Konto, das die Anwendung hostet, und anhand bestimmter Gruppen von Ressourcen innerhalb des Kontos auffindbar und verwendet AWS Organizations zur Verwaltung der AWS-Konten. Im Laufe der Zeit konvertiert AnyCompany Retail diese Prozesse in Code und definiert Ressourcen mithilfe von Infrastructure as Code (über Services wie CloudFormation oder AWS Cloud Development Kit (AWS CDK)-Vorlagen). Die Betriebsprozesse werden zu Automatisierungsdokumenten in AWS Systems Manager- oder AWS Lambda-Funktionen, die als geplante Aufgaben, als Reaktion auf Ereignisse wie Amazon CloudWatch-Alarme oder Amazon EventBridge-Ereignisse oder durch Anfragen innerhalb einer IT-Servicemanagement-Plattform (ITSM) gestartet werden können. Alle Prozesse sind mit Tags versehen, um die Zuständigkeit zu identifizieren. Teams verwalten die Dokumentation für die Automatisierung und den Prozess auf den Wiki-Seiten, die vom Code-Repository für den Prozess generiert werden.

Implementierungsschritte

1. Dokumentieren Sie die bestehenden Prozesse und Verfahren.
 - a. Überprüfen und vergewissern Sie sich, dass sie auf dem neuesten Stand sind.
 - b. Stellen Sie sicher, dass jeder Prozess oder jedes Verfahren einen Besitzer hat.
 - c. Stellen Sie die Verfahren unter Versionskontrolle.
 - d. Wenn möglich, nutzen Sie Prozesse und Verfahren für Workloads und Umgebungen mit gemeinsamen Architekturentwürfen.
2. Richten Sie Mechanismen für Feedback und Verbesserung ein.
 - a. Definieren Sie Richtlinien dafür, wie oft Prozesse überprüft werden sollten.
 - b. Definieren Sie Prozesse für Prüfende und Genehmigende.
 - c. Implementieren Sie Probleme oder eine Ticket-Warteschlange, um Feedback zu geben und zu verfolgen.
 - d. Wo immer es möglich ist, sollten Prozesse und Verfahren vorab von einem Gremium zur Genehmigung von Änderungen genehmigt und in eine Risikoklasse eingestuft werden.
3. Machen Sie Prozesse und Verfahren für Benutzer zugänglich und auffindbar, die sie ausführen müssen.
 - a. Verwenden Sie Tags, um anzugeben, wo der Prozess und die Verfahren für die Workload aufgerufen werden können.
 - b. Verwenden Sie aussagekräftige Fehler- und Ereignismeldungen, um die geeigneten Prozesse oder Verfahren zur Behebung des Problems anzugeben.

- c. Verwenden Sie Wikis oder Dokumentenmanagement, um Prozesse und Verfahren unternehmensweit durchsuchbar zu machen.
4. Automatisieren Sie, wenn es angemessen ist.
 - a. Entwickeln Sie Automatisierungen, wenn Services und Technologien eine API bereitstellen.
 - b. Stellen Sie sicher, dass die Prozesse gut verstanden werden, und entwickeln Sie Benutzerberichte und Anforderungen, um diese Prozesse zu automatisieren.
 - c. Messen Sie die erfolgreiche Nutzung der Prozesse und Verfahren und unterstützen Sie eine iterative Verbesserung anhand der Problemverfolgung.

Aufwand für den Implementierungsplan: Mittel

Ressourcen

Zugehörige bewährte Methoden:

- [OPS02-BP01 Ressourcen haben feste Verantwortliche](#)
- [OPS02-BP02 Prozesse und Verfahren haben feste Besitzer](#)
- [OPS02-BP04 Es gibt Mechanismen zur Verwaltung von Verantwortlichkeiten und Zuständigkeiten](#)
- [OPS02-BP05 Mechanismen zur Identifizierung von Verantwortlichkeit und Eigentümerschaft sind vorhanden](#)
- [OPS11-BP04 Wissensmanagement](#)

Zugehörige Dokumente:

- [AWS-Whitepaper | Einführung in DevOps in AWS](#)
- [AWS-Whitepaper | Bewährte Methoden für das Taggen von AWS-Ressourcen](#)
- [AWS-Whitepaper | Organisieren Sie Ihre AWS-Umgebung mit mehreren Konten](#)
- [AWS Cloud Operations & Migrations Blog | Aufbau einer Cloud-Automatisierungspraxis für betriebliche Exzellenz: Bewährte Methoden von AWS Managed Services](#)
- [AWS-Workshop – Tagging](#)
- [AWS Service Management-Konnektor](#)

Zugehörige Videos:

- [AWS Knowledge Center Live | Tagging von AWS-Ressourcen](#)

- [AWS re:Invent 2020 | Automatisierung aller Abläufe mit AWS Systems Manager](#)
- [AWS re:Inforce 2022 | Automatisierung der Patch-Verwaltung und -Compliance mit AWS \(NIS306\)](#)
- [Supports You | Detaillierte Einblicke in AWS Systems Manager](#)

OPS02-BP04 Es gibt Mechanismen zur Verwaltung von Verantwortlichkeiten und Zuständigkeiten

Verstehen Sie die Verantwortlichkeiten Ihrer Rolle und, wie Sie zu Geschäftsergebnissen beitragen, da Ihnen dieses Wissen ermöglicht, Ihre Aufgaben entsprechend zu priorisieren und die Bedeutung Ihrer Rolle nachzuvollziehen. Auf diese Weise können Teammitglieder Anforderungen erkennen und entsprechend reagieren. Wenn die Teammitglieder ihre Rolle kennen, können sie Verantwortung übernehmen, Verbesserungsmöglichkeiten erkennen und verstehen, wie sie Einfluss nehmen oder entsprechende Änderungen vornehmen können.

Gelegentlich kann es vorkommen, dass eine Verantwortlichkeit keinen eindeutigen Besitzer hat. Entwerfen Sie in diesen Situationen einen Mechanismus, um diese Lücke zu schließen. Erstellen Sie einen klar definierten Eskalationsweg zu jemandem, der die Befugnis hat, die Verantwortung zu übertragen, oder entwickeln Sie einen Plan zur Deckung des Bedarfs.

Gewünschtes Ergebnis: Teams in Ihrer Organisation haben klar definierte Verantwortlichkeiten, zu denen auch gehört, in welcher Beziehung sie zu Ressourcen, durchzuführenden Aktionen, Prozessen und Verfahren stehen. Diese Verantwortlichkeiten entsprechen den Verantwortlichkeiten und Zielen des Teams sowie den Verantwortlichkeiten anderer Teams. Sie dokumentieren die Eskalationswege auf konsistente und nachvollziehbare Weise und nehmen diese Entscheidungen in Dokumentationsartefakte wie Zuständigkeitsmatrizen, Teamdefinitionen oder Wiki-Seiten auf.

Typische Anti-Muster:

- Die Verantwortlichkeiten des Teams sind mehrdeutig oder schlecht definiert.
- Das Team stimmt Rollen nicht mit Verantwortlichkeiten ab.
- Das Team stimmt seine Ziele und Verantwortlichkeiten nicht aufeinander ab, was es schwierig macht, den Erfolg zu messen.
- Die Verantwortlichkeiten der Teammitglieder sind nicht am Team und der gesamten Organisation ausgerichtet.
- Ihr Team hält die Verantwortlichkeiten nicht auf dem neuesten Stand, was dazu führt, dass sie nicht mit den vom Team ausgeführten Aufgaben übereinstimmen.
- Eskalationswege zur Festlegung von Zuständigkeiten sind nicht definiert oder unklar.

- Eskalationswege haben keinen eindeutigen Besitzer, um eine zeitnahe Reaktion zu gewährleisten.
- Rollen, Zuständigkeiten und Eskalationswege sind nicht auffindbar und bei Bedarf nicht sofort verfügbar (z. B. als Reaktion auf einen Vorfall).

Vorteile der Nutzung dieser bewährten Methode:

- Wenn Sie wissen, wer verantwortlich oder zuständig ist, können Sie sich an das entsprechende Team oder Teammitglied wenden, um eine Anfrage zu stellen oder eine Aufgabe zu übergeben.
- Um das Risiko von Untätigkeit und ungedecktem Bedarf zu verringern, haben Sie eine Person festgelegt, die befugt ist, Verantwortung und Zuständigkeit zu übertragen.
- Wenn Sie den Umfang einer Verantwortlichkeit klar definieren, gewinnen Ihre Teammitglieder an Autonomie und Eigenverantwortung.
- Ihre Verantwortlichkeiten wirken sich auf Ihre Entscheidungen, Ihre Aktionen und die Übergabe von Aktivitäten an die ordnungsgemäßen Besitzer aus.
- Es ist einfach, aufgegebene Verantwortlichkeiten zu identifizieren, da Sie genau wissen, was außerhalb der Verantwortung Ihres Teams liegt, was die Eskalation zur Aufklärung erleichtert.
- Es kommt innerhalb der Teams zu weniger Verwirrung und Spannungen und sie können ihre Workloads und Ressourcen besser verwalten.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Legen Sie die Rollen und Verantwortlichkeiten von Teammitgliedern fest und vergewissern Sie sich, dass sie die Anforderungen ihrer Rolle kennen. Diese Informationen sollten leicht auffindbar sein, damit Mitglieder Ihrer Organisation herausfinden können, an wen sie sich für bestimmte Anforderungen wenden müssen (an ein Team oder eine Person). In dem Bestreben, die Chancen der Migration und Modernisierung in AWS zu nutzen, können sich auch die Rollen und Verantwortlichkeiten ändern. Sorgen Sie dafür, dass sich Ihre Teams und ihre Mitglieder ihrer Verantwortlichkeiten bewusst sind, und schulen Sie sie angemessen, damit sie ihre Aufgaben während dieser Veränderung erfüllen.

Legen Sie fest, an welche Rolle oder welches Team eskaliert werden soll, um die Verantwortlichkeit und Zuständigkeit zu bestimmen. Dieses Team kann mit verschiedenen Stakeholdern zusammenarbeiten, um eine Entscheidung zu treffen. Es sollte jedoch die Verantwortung für die Verwaltung des Entscheidungsprozesses tragen.

Stellen Sie Mitgliedern Ihrer Organisation zugängliche Mechanismen bereit, um Zuständigkeiten und Verantwortlichkeiten zu ermitteln und zuzuordnen. Diese Mechanismen vermitteln ihnen, an wen sie sich bei spezifischen Bedürfnissen wenden können.

Kundenbeispiel

AnyCompany Retail hat kürzlich eine Migration von Workloads von einer On-Premises-Umgebung zu ihrer Landing Zone in AWS mit einem Lift-and-Shift-Ansatz durchgeführt. Das Unternehmen führte eine Betriebsüberprüfung durch, um festzustellen, wie allgemeine betriebliche Aufgaben erfüllt werden, und verifizierte, dass seine bestehende Verantwortungsmatrix die Abläufe in der neuen Umgebung widerspiegelt. Bei der Migration von On-Premises zu AWS reduzierte es die Verantwortlichkeiten der Infrastrukturteams in Bezug auf die Hardware und die physische Infrastruktur. Dieser Schritt eröffnete auch neue Möglichkeiten, das Betriebsmodell für seine Workloads weiterzuentwickeln.

Es identifizierte, adressierte und dokumentierte die meisten Verantwortlichkeiten, definierte aber auch Eskalationswege für alle Verantwortlichkeiten, die übersehen wurden oder die sich im Zuge der Weiterentwicklung der betrieblichen Abläufe möglicherweise ändern müssen. Um neue Möglichkeiten zur Standardisierung und Effizienzsteigerung Ihrer Workloads zu erkunden, bieten Sie Zugriff auf Betriebstools wie AWS Systems Manager und Sicherheitstools wie AWS Security Hub CSPM und Amazon GuardDuty. AnyCompany Retail überprüft die Verantwortlichkeiten und die Strategie auf der Grundlage der Verbesserungen, die zuerst angegangen werden sollen. Wenn das Unternehmen neue Arbeitsweisen und Technologiemuster einführt, passt es seine Verantwortungsmatrix entsprechend an.

Implementierungsschritte

1. Beginnen Sie mit der vorhandenen Dokumentation. Zu den typischen Quelldokumenten gehören möglicherweise:
 - a. Verantwortungs- oder RACI-Matrizen (Responsible, Accountable, Consulted, Informed)
 - b. Teamdefinitionen oder Wiki-Seiten
 - c. Servicedefinitionen und Angebote
 - d. Rollen- oder Stellenbeschreibungen
2. Überprüfen und besprechen Sie die dokumentierten Verantwortlichkeiten:
 - a. Führen Sie Besprechungen mit den Teams durch, um Abweichungen zwischen den dokumentierten Verantwortlichkeiten und den vom Team üblicherweise wahrgenommenen Verantwortlichkeiten zu identifizieren.

- b. Erörtern Sie mögliche Services, die von internen Kunden angeboten werden, um unterschiedliche Erwartungen zwischen den Teams zu identifizieren.
3. Analysieren und beheben Sie die Diskrepanzen.
4. Identifizieren Sie Verbesserungsmöglichkeiten.
 - a. Identifizieren Sie häufig nachgefragte, ressourcenintensive Anfragen, bei denen es sich in der Regel um gute Verbesserungsmöglichkeiten handelt.
 - b. Informieren Sie sich über bewährte Methoden, verstehen Sie Muster, folgen Sie verbindlichen Anleitungen und vereinfachen und standardisieren Sie Verbesserungen.
 - c. Erfassen Sie Verbesserungsmöglichkeiten und verfolgen Sie sie bis zur Fertigstellung.
5. Wenn ein Team noch nicht die Verantwortung für die Verwaltung und die Verfolgung der Zuweisung von Verantwortlichkeiten trägt, benennen Sie jemanden im Team, der diese Verantwortung übernimmt.
6. Definieren Sie einen Prozess, nach dem Teams eine Klärung der Verantwortlichkeiten anfordern können.
 - a. Überprüfen Sie den Prozess und stellen Sie sicher, dass er klar und einfach umzusetzen ist.
 - b. Stellen Sie sicher, dass jemand die Verantwortung für die Eskalationen trägt und sie bis zu ihrem Ende verfolgt.
 - c. Legen Sie betriebliche Metriken fest, um die Effektivität zu messen.
 - d. Schaffen Sie Feedback-Mechanismen, um sicherzustellen, dass Teams Verbesserungsmöglichkeiten hervorheben können.
 - e. Implementieren Sie einen Mechanismus für die regelmäßige Überprüfung.
7. Führen Sie Dokumente an einem auffindbaren und zugänglichen Ort.
 - a. Wikis oder das Dokumentationsportal sind gängige Optionen.

Aufwand für den Implementierungsplan: Mittel

Ressourcen

Zugehörige bewährte Methoden:

- [OPS01-BP06 Bewerten von Kompromissen](#)
- [OPS03-BP02 Teammitglieder sind befugt, Maßnahmen zu ergreifen, wenn Ergebnisse gefährdet sind](#)
- [OPS03-BP03 Eskalation wird empfohlen](#)

- [OPS03-BP07 Teams mit entsprechenden Ressourcen ausstatten](#)
- [OPS09-BP01 Messen operativer Ziele und KPIs mit Metriken](#)
- [OPS09-BP03 Überprüfen der Betriebsmetriken und Priorisieren von Verbesserungen](#)
- [OPS11-BP01 Implementieren eines Prozesses für die kontinuierliche Verbesserung](#)

Zugehörige Dokumente:

- [AWS-Whitepaper – Einführung in DevOps in AWS](#)
- [AWS-Whitepaper – AWS Cloud Adoption Framework: Betriebsperspektive](#)
- [AWS Well-Architected-Framework – Betriebliche Exzellenz — Betriebsmodelltopologien auf Workload-Ebene](#)
- [AWS Prescriptive Guidance – Aufbau Ihres Cloud-Betriebsmodells](#)
- [AWS Prescriptive Guidance – Erstellen einer RACI- oder RASCI-Matrix für ein Cloud-Betriebsmodell](#)
- [AWS Cloud Operations & Migrations Blog – Schaffen von Mehrwert für Ihr Unternehmen mit Cloud-Plattform-Teams](#)
- [AWS Cloud Operations & Migrations Blog – Vorteile eines Cloud-Betriebsmodells](#)
- [AWS DevOps Blog – So modernisieren Unternehmen für den Cloud-Betrieb](#)

Zugehörige Videos:

- [AWS Summit Online – Cloud-Betriebsmodelle für eine schnellere Transformation](#)
- [AWS re:Invent 2023 – Cloud-Sicherheit zukunftssicher machen: Ein neues Betriebsmodell](#)

OPS02-BP05 Mechanismen zum Anfordern von Ergänzungen, Änderungen und Ausnahmen sind vorhanden

Sie können Anfragen an Verantwortliche für Prozesse, Verfahren und Ressourcen stellen. Die Anfragen umfassen Ergänzungen, Änderungen und Ausnahmen. Diese Anfragen durchlaufen einen Änderungsverwaltungsprozess. Treffen Sie fundierte Entscheidungen, um angemessene Anfragen nach einer Bewertung der Vorteile und Risiken zu genehmigen.

Gewünschtes Ergebnis:

- Sie können Anfragen zum Ändern von Prozessen, Verfahren und Ressourcen basierend auf der zugewiesenen Verantwortlichkeit stellen.
- Änderungen werden nach einem sorgfältigen Abwägen der Vorteile und Risiken vorgenommen.

Typische Anti-Muster:

- Sie müssen die Art und Weise der Bereitstellung Ihrer Anwendung aktualisieren, es gibt jedoch keine Möglichkeit, eine Änderung am Bereitstellungsprozess beim Produktionsteam zu beantragen.
- Der Notfallwiederherstellungsplan muss aktualisiert werden, es ist jedoch kein Verantwortlicher kenntlich gemacht, an den Anträge auf Änderungen übermittelt werden können.

Vorteile der Nutzung dieser bewährten Methode:

- Prozesse, Verfahren und Ressourcen können sich weiterentwickeln, wenn sich die Anforderungen ändern.
- Die Verantwortlichen können fundierte Entscheidungen treffen, wann Änderungen vorgenommen werden sollten.
- Änderungen werden nach sorgfältigen Überlegungen vorgenommen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Um diese bewährte Methode zu implementieren, müssen Sie Änderungen an Prozessen, Verfahren und Ressourcen beantragen können. Der Änderungsverwaltungsprozess kann einfach sein. Dokumentieren Sie den Änderungsverwaltungsprozess.

Kundenbeispiel

AnyCompany Retail verwendet für die Angabe, wer für Änderungen an Prozessen, Verfahren und Ressourcen verantwortlich ist, eine Verantwortlichkeitsmatrix (RACI). Es gibt einen dokumentierten Änderungsverwaltungsprozess, der einfach und leicht zu befolgen ist. Mithilfe der RACI-Matrix und des Prozesses können alle Personen Änderungsanträge übermitteln.

Implementierungsschritte

1. Ermitteln Sie die Prozesse, Verfahren und Ressourcen für Ihren Workload sowie die jeweiligen Verantwortlichen. Dokumentieren Sie sie in Ihrem Wissensmanagementsystem.

- a. Wenn Sie [OPS02-BP01 Ressourcen haben Eigentümer identifiziert](#), [OPS02-BP02 Prozesse und Verfahren haben feste Besitzer](#) oder [OPS02-BP03 Betriebsaktivitäten haben feste Besitzer, die für ihre Leistung verantwortlich sind](#) nicht implementiert haben, beginnen Sie damit.
2. Arbeiten Sie mit den Stakeholdern in Ihrer Organisation zusammen, um einen Änderungsverwaltungsprozess zu entwickeln. Der Prozess sollte Ergänzungen, Änderungen und Ausnahmen für Ressourcen, Prozesse und Verfahren umfassen.
 - a. Sie können [AWS Systems Manager Change Manager](#) als Änderungsverwaltungsplattform für Workload-Ressourcen verwenden.
3. Dokumentieren Sie den Änderungsverwaltungsprozess in Ihrem Wissensmanagementsystem.

Aufwand für den Implementierungsplan: Mittel. Die Entwicklung eines Änderungsverwaltungsprozesses erfordert die Abstimmung mit mehreren Stakeholdern in Ihrer Organisation.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS02-BP01 Ressourcen haben Eigentümer identifiziert](#) – Bevor Sie einen Änderungsverwaltungsprozess entwickeln, müssen für Ressourcen die Besitzer identifiziert werden.
- [OPS02-BP02 Prozesse und Verfahren haben feste Besitzer](#) – Bevor Sie einen Änderungsverwaltungsprozess entwickeln, müssen für Prozesse die Besitzer identifiziert werden.
- [OPS02-BP03 Betriebsaktivitäten haben feste Besitzer, die für ihre Leistung verantwortlich sind](#) – Bevor Sie einen Änderungsverwaltungsprozess entwickeln, müssen für Betriebsaktivitäten die Besitzer identifiziert werden.

Zugehörige Dokumente:

- [AWS Prescriptive Guidance – Grundlagen-Playbook für große AWS_Migrationen: Erstellung von RACI-Matrizen](#)
- [Whitepaper „Änderungsmanagement in der Cloud“](#)

Zugehörige Services:

- [AWS Systems Manager Change Manager](#)

OPS02-BP06 Zuständigkeiten zwischen Teams werden vordefiniert oder ausgehandelt

Es gibt definierte oder ausgehandelte Vereinbarungen zwischen Teams, in denen die Zusammenarbeit und gegenseitige Unterstützung beschrieben wird (z. B. Reaktionszeiten, Service-Level-Ziele oder Service-Level-Agreements). Die Kanäle für die teamübergreifende Kommunikation werden dokumentiert. Wenn bekannt ist, welche Auswirkungen die Arbeit der Teams auf die Geschäftsergebnisse und die Ergebnisse anderer Teams und Organisationen hat, können die Teams ihre Aufgaben priorisieren und entsprechend handeln.

Wenn Verantwortlichkeit und Zuständigkeit nicht definiert oder unbekannt sind, besteht das Risiko, dass sowohl die erforderlichen Aktivitäten nicht rechtzeitig ausgeführt als auch redundante und potenziell widersprüchliche Anstrengungen unternommen werden, um diese Anforderungen zu erfüllen.

Gewünschtes Ergebnis:

- Es werden Vereinbarungen zur teamübergreifenden Zusammenarbeit oder Unterstützung getroffen und dokumentiert.
- Teams, die zusammenarbeiten oder sich gegenseitig unterstützen, verfügen über definierte Kommunikationskanäle und Erwartungen in Bezug auf die Reaktion.

Typische Anti-Muster:

- Während der Produktion tritt ein Problem auf und zwei separate Teams beginnen unabhängig voneinander mit der Fehlersuche. Aufgrund der getrennten Bemühungen verlängert sich der Ausfall.
- Das Produktionsteam benötigt Unterstützung vom Entwicklungsteam, es gibt jedoch keine Vereinbarung in Bezug auf die Reaktionszeit. Die Anfrage wird zurückgestellt.

Vorteile der Nutzung dieser bewährten Methode:

- Die Teams wissen, wie sie miteinander interagieren und sich gegenseitig unterstützen können.
- Die Erwartungen in Bezug auf die Reaktionszeit sind bekannt.
- Die Kommunikationskanäle sind klar definiert.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Niedrig

Implementierungsleitfaden

Wenn Sie diese bewährte Methode implementieren, bedeutet dies, dass es in Bezug auf die Zusammenarbeit zwischen Teams keine Unklarheiten gibt. Mithilfe von formellen Vereinbarungen wird festgelegt, wie Teams zusammenarbeiten oder sich gegenseitig unterstützen. Die Kanäle für die teamübergreifende Kommunikation werden dokumentiert.

Kundenbeispiel

Das SRE-Team bei AnyCompany Retail hat ein Service-Level-Agreement mit dem Entwicklungsteam abgeschlossen. Wenn das Entwicklungsteam eine Anfrage über das Ticketing-System einreicht, kann es innerhalb von 15 Minuten eine Antwort erwarten. Bei Standortausfällen übernimmt das SRE-Team mit Unterstützung durch das Entwicklungsteam die Leitung der Untersuchung.

Implementierungsschritte

1. Arbeiten Sie zusammen mit den Stakeholdern in Ihrer Organisation und auf Grundlage der Prozesse und Verfahren Vereinbarungen zwischen Teams aus.
 - a. Entwickeln Sie für gemeinsame Prozesse oder Verfahren von zwei Teams ein Runbook für die Zusammenarbeit.
 - b. Wenn Abhängigkeiten zwischen Teams bestehen, vereinbaren Sie ein SLA für die Reaktionszeit bei Anfragen.
2. Dokumentieren Sie die Verantwortlichkeiten in Ihrem Wissensmanagementsystem.

Aufwand für den Implementierungsplan: Mittel. Wenn keine Vereinbarungen zwischen Teams vorhanden sind, kann es mühsam sein, eine Vereinbarung mit den Stakeholdern in Ihrer Organisation zu treffen.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS02-BP02 Prozesse und Verfahren haben feste Besitzer](#) – Die Verantwortlichkeiten für Prozesse müssen bestimmt werden, bevor Teams Vereinbarungen miteinander treffen.
- [OPS02-BP03 Betriebsaktivitäten haben feste Besitzer, die für ihre Leistung verantwortlich sind](#) – Die Verantwortlichkeiten für Betriebsaktivitäten müssen bestimmt werden, bevor Teams Vereinbarungen miteinander treffen.

Zugehörige Dokumente:

- [AWS Executive Insights – Innovation fördern mit dem Zwei-Pizzen-Team](#)
- [Einführung in DevOps auf AWS – Zwei-Pizzen-Teams](#)

OPS 3. Wie unterstützt Ihre Unternehmenskultur Ihre Geschäftsergebnisse?

Lassen Sie Ihren Teammitgliedern Unterstützung zukommen, damit sie effektiver handeln und Ihr Geschäftsergebnis unterstützen können.

Best Practices

- [OPS03-BP01 Förderung durch die Geschäftsführung gewährleisten](#)
- [OPS03-BP02 Teammitglieder sind befugt, Maßnahmen zu ergreifen, wenn Ergebnisse gefährdet sind](#)
- [OPS03-BP03 Eskalation wird gefördert](#)
- [OPS03-BP04 Die Kommunikation ist zeitnah, klar und umsetzbar](#)
- [OPS03-BP05 Experimentieren wird empfohlen](#)
- [OPS03-BP06 Teammitglieder werden ermutigt, ihre Fähigkeiten zu pflegen und zu erweitern](#)
- [OPS03-BP07 Ressourcenteams angemessen](#)

OPS03-BP01 Förderung durch die Geschäftsführung gewährleisten

Auf höchster Ebene fungiert die Geschäftsleitung als Executive Sponsor, um Erwartungen klar festzulegen und die Richtung für die Ergebnisse der Organisation vorzugeben sowie den Erfolg zu bewerten. Der Sponsor befürwortet und fördert die Einführung von bewährten Methoden und die Weiterentwicklung der Organisation.

Gewünschtes Ergebnis: Organisationen, die einen Cloud-Betrieb einführen, transformieren oder optimieren möchten, legen eine klare Führung und Rechenschaftspflicht zum Erreichen der gewünschten Ergebnisse fest. Die Organisation kennt jede Fähigkeit, die sie benötigt, um ein neues Ergebnis zu erzielen, und überträgt den Funktionsteams die Verantwortung für die Entwicklung dieser Fähigkeiten. Die Führung gibt diese Richtung aktiv vor, weist Verantwortung zu, übernimmt Verantwortung und definiert die Arbeit. Dadurch können Mitarbeiter in der gesamten Organisation mobilisieren, sich inspiriert fühlen und aktiv auf die gewünschten Ziele hinarbeiten.

Typische Anti-Muster:

- Workload-Besitzer sind aufgefordert, Workloads zu AWS zu migrieren – ohne klare Unterstützung oder einen Plan für den Cloud-Betrieb. Dies führt dazu, dass Teams nicht gezielt zusammenarbeiten, um ihre operativen Fähigkeiten zu verbessern und weiterzuentwickeln. Der Mangel an Betriebsstandards mit bewährten Methoden führt dazu, dass die Teams überfordert sind (z. B. durch Überarbeitung der Mitarbeiter, Bereitschaftsdienste und technische Schulden) und die Innovation ins Stocken gerät.
- Es wurde ein neues organisationsweites Ziel gesetzt, eine neue Technologie einzuführen, ohne die Führung, den Sponsor und die Strategie anzugeben. Die Teams interpretieren Ziele unterschiedlich, was zu Verwirrung darüber führt, worauf sie sich konzentrieren sollten, warum sie wichtig sind und wie Auswirkungen gemessen werden sollen. Folglich verliert die Organisation bei der Einführung der Technologie an Dynamik.

Vorteile der Nutzung dieser bewährten Methode: Wenn die Geschäftsführung Vision, Ausrichtung und Ziele klar kommuniziert und teilt, wissen die Teammitglieder, was von ihnen erwartet wird. Wenn sich die Führungskräfte aktiv einbringen, beginnen Einzelpersonen und Teams, ihre Bemühungen intensiv in dieselbe Richtung zu lenken, um festgelegte Ziele zu erreichen. So kann die Organisation ihre Erfolgsfähigkeit maximieren. Wenn Sie den Erfolg evaluieren, können Sie Barrieren besser identifizieren, um anschließend von der Führung gezielt ausgeräumt werden können.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

- In jeder Phase des Wegs in die Cloud (Migration, Einführung oder Optimierung) erfordert der Erfolg eine aktive Beteiligung auf höchster Führungsebene mit einem leitenden Unterstützer. Der leitende Unterstützer richtet die Denkweise, Fähigkeiten und Arbeitsweisen des Teams an der definierten Strategie aus.
- Erläutern des Warums: Sorgen Sie für Klarheit und erläutern Sie die Gründe für die Vision und die Strategie.
- Festlegen der Erwartungen: Definieren und veröffentlichen Sie Ziele für Ihre Organisationen, einschließlich der Art und Weise, wie Fortschritt und Erfolg gemessen werden.
- Verfolgen der Zielerreichung: Messen Sie regelmäßig das schrittweise Erreichen von Zielen (nicht nur die Erledigung von Aufgaben). Teilen Sie die Ergebnisse mit, damit geeignete Maßnahmen ergriffen werden können, wenn die Ergebnisse gefährdet sind.
- Bereitstellen der erforderlichen Ressourcen zum Erreichen Ihrer Ziele: Bringen Sie Menschen und Teams zusammen, um zusammenzuarbeiten und die richtigen Lösungen zu entwickeln,

die zu den definierten Ergebnissen führen. Dies reduziert oder beseitigt Reibungspunkte in der Organisation.

- **Unterstützen Ihrer Teams:** Bleiben Sie mit Ihren Teams in Kontakt, damit Sie ihre Leistung verstehen und herausfinden können, ob diese durch externe Faktoren beeinflusst wird. Identifizieren Sie Hindernisse für den Fortschritt Ihrer Teams. Treten Sie für Ihre Teams ein und beseitigen Sie Hindernisse und unnötige Belastungen. Wenn sich äußere Faktoren negativ auf Ihre Teams auswirken, bewerten Sie die Ziele neu und passen Sie sie entsprechend an.
- **Fördern der Einführung von bewährten Methoden:** Würdigen Sie bewährte Methoden, die messbare Vorteile bieten, und schenken Sie ihren Entwicklern und Anwendern Ihre Anerkennung. Ermutigen Sie Ihre Teams zur Annahme dieser Methoden, um die Vorteile zu maximieren.
- **Ermuntern Sie Ihre Teams zur Weiterentwicklung:** Schaffen Sie eine Kultur der kontinuierlichen Verbesserung und lernen Sie proaktiv aus Fortschritten und Fehlschlägen. Fördern Sie Wachstum und Entwicklung sowohl im Persönlichen als auch im Betrieblichen. Entwickeln Sie die Vision und Strategie anhand von Daten und Anekdoten weiter.

Kundenbeispiel

AnyCompany Retail befindet sich inmitten einer Geschäftstransformation mit dem Ziel, das Kundenerlebnis schnell neu zu erfinden, die Produktivität zu steigern und das Wachstum durch generative KI zu beschleunigen.

Implementierungsschritte

1. Ernennen Sie einen einzelnen Verantwortlichen und einen leitenden Unterstützer, der die Transformation leitet und vorantreibt.
2. Definieren Sie klare Geschäftsergebnisse für Ihre Transformation, weisen Sie Verantwortlichkeiten zu und fordern Sie Eigenverantwortung ein. Erteilen Sie der leitenden Führungskraft die Befugnis, wichtige Entscheidungen zu leiten und zu treffen.
3. Stellen Sie sicher, dass Ihre Transformationsstrategie sehr klar ist und vom leitenden Sponsor auf allen Ebenen der Organisation umfassend kommuniziert wird.
 - a. Legen Sie klar definierte Geschäftsziele für IT- und Cloud-Initiativen fest.
 - b. Dokumentieren Sie wichtige Geschäftsmetriken, um die IT- und Cloud-Transformation voranzutreiben.
 - c. Kommunizieren Sie die Vision konsequent an alle Teams und Personen, die für Teile der Strategie verantwortlich sind.

4. Entwickeln Sie Matrizen zur Kommunikationsplanung, die vorgeben, welche Botschaft bestimmten Führungskräften, Managern und einzelnen Mitarbeitern übermittelt werden muss. Legen Sie fest, welche Person oder welches Team diese Nachricht übermitteln soll.
 - a. Erfüllen Sie Kommunikationspläne konsistent und zuverlässig.
 - b. Setzen und steuern Sie Ihre Erwartungen regelmäßig in persönlichen Meetings.
 - c. Nehmen Sie Feedback zur Effektivität der Kommunikation an, passen Sie die Kommunikation an und planen Sie entsprechend.
 - d. Planen Sie Kommunikationsveranstaltungen, um die Herausforderungen der Teams proaktiv zur Kenntnis zu nehmen, und richten Sie eine konsistente Feedback-Schleife ein, um den Kurs bei Bedarf zu korrigieren.
5. Beschäftigen Sie sich aktiv mit jeder Initiative aus der Führungsperspektive, um sicherzustellen, dass alle betroffenen Teams die Ergebnisse verstehen, für deren Erreichung sie verantwortlich sind.
6. Bei jedem Status-Meeting sollten die leitenden Unterstützer nach Hindernissen Ausschau halten, etablierte Metriken, Anekdoten oder das Feedback der Teams überprüfen und die Fortschritte bei der Erreichung der Ziele messen.

Aufwand für den Implementierungsplan: Mittel

Ressourcen

Zugehörige bewährte Methoden:

- [OPS03-BP04 Die Kommunikation ist zeitnah, klar und umsetzbar](#)
- [OP11-BP01 Implementieren eines Prozesses für die kontinuierliche Verbesserung](#)
- [OPS11-BP07 Prüfung von Betriebsmetriken](#)

Zugehörige Dokumente:

- [Entwirren Ihrer Organisation: Stark ausgerichtet](#)
- [Die lebendige Transformation: Veränderungen pragmatisch angehen](#)
- [Transformation zu einem zukunftsfähigen Unternehmen](#)
- [7 Fehler, die Sie bei der Einrichtung eines CCOE vermeiden sollten](#)
- [Navigation in der Cloud: Wichtige Leistungskennzahlen für den Erfolg](#)

Zugehörige Videos:

- [AWS re:Invent 2023: Leitfaden für Führungskräfte zur generativen KI: Berücksichtigung der Geschichte bei der Gestaltung der Zukunft \(SEG204\)](#)

Zugehörige Beispiele:

- [Prosci: Rolle und Bedeutung des leitenden Unterstützers](#)

OPS03-BP02 Teammitglieder sind befugt, Maßnahmen zu ergreifen, wenn Ergebnisse gefährdet sind

Eine von der Führung vermittelte Kultur der Eigenverantwortung führt dazu, dass sich die Mitarbeiter bestärkt fühlen, im Namen des gesamten Unternehmens über ihren definierten Rollen- und Verantwortungsbereich hinaus zu handeln. Die Mitarbeiter können handeln, um auftretende Risiken proaktiv zu erkennen und geeignete Maßnahmen ergreifen. Eine solche Kultur ermöglicht es den Mitarbeitern, die Situation zu überblicken und wichtige Entscheidungen zu treffen.

Amazon verwendet beispielsweise [Führungsprinzipien](#) als Richtlinien, um das gewünschte Verhalten der Mitarbeiter zu fördern, damit sie in Situationen vorankommen, Probleme lösen, mit Konflikten umgehen und Maßnahmen ergreifen können.

Gewünschtes Ergebnis: Die Führung hat eine neue Kultur beeinflusst, die es Einzelpersonen und Teams ermöglicht, wichtige Entscheidungen zu treffen – selbst auf niedrigeren Ebenen der Organisation (solange Entscheidungen mit überprüfbaren Genehmigungen und Sicherheitsmechanismen definiert sind). Misserfolge werden als Lernerfahrung angesehen, und Teams lernen schrittweise, ihre Entscheidungen und Maßnahmen zu optimieren, um in Zukunft ähnliche Situationen zu bewältigen. Wenn die Maßnahmen einer Person zu einer Verbesserung führen, von der andere Teams profitieren können, werden die aus solchen Maßnahmen gewonnenen Erkenntnisse proaktiv geteilt. Die Geschäftsführung misst betriebliche Verbesserungen und bietet dem Einzelnen sowie der Organisation Anreize für die Übernahme solcher Muster.

Typische Anti-Muster:

- In einer Organisation gibt es keine klaren Leitlinien oder Mechanismen dafür, was zu tun ist, wenn ein Risiko erkannt wird. Wenn ein Mitarbeiter beispielsweise einen Phishing-Angriff bemerkt und dies nicht dem Sicherheitsteam meldet, kann dies zur Folge haben, dass ein großer Teil der Organisation auf den Angriff hereinfällt. Dies führt zu einer Datenschutzverletzung.

- Ihre Kunden beschwerten sich über die Nichtverfügbarkeit von Services, die hauptsächlich auf fehlgeschlagene Bereitstellungen zurückzuführen ist. Ihr SRE-Team ist für das Bereitstellungstool verantwortlich, und ein automatisiertes Rollback für Bereitstellungen ist Teil der langfristigen Roadmap. Bei einer kürzlichen Anwendungseinführung entwickelte einer der Engineers eine Lösung, um das Rollback seiner Anwendung auf eine frühere Version zu automatisieren. Obwohl die Lösung zum Vorbild für SRE-Teams werden könnte, wird sie von anderen Teams nicht übernommen, da kein Prozess zur Nachverfolgung solcher Verbesserungen vorhanden ist. Die Organisation wird weiterhin durch fehlgeschlagene Bereitstellungen unter Druck gesetzt, die sich auf die Kunden auswirken und die Reputation des Unternehmens gefährden.
- Zur Wahrung der Compliance überwacht Ihr Infosec-Team einen seit langem etablierten Prozess, bei dem gemeinsam genutzte SSH-Schlüssel im Namen der Betreiber, die eine Verbindung zu ihren Amazon EC2-Linux-Instances herstellen, regelmäßig rotieren. Die InfoSec-Teams brauchen mehrere Tage für die Schlüsselrotation. In dieser Zeit können Sie keine Verbindung zu diesen Instances herstellen. Bisher gab es keine Vorschläge, weder seitens von Infosec noch von außerhalb, zur Nutzung anderer Optionen in AWS, um dasselbe Ergebnis zu erzielen.

Vorteile der Nutzung dieser bewährten Methode: Indem Sie die Entscheidungsbefugnisse dezentralisieren und Ihre Teams in die Lage versetzen, wichtige Entscheidungen zu treffen, können Sie Probleme schneller lösen und die Erfolgsquoten steigern. Darüber hinaus beginnen die Teams, ein Gefühl der Eigenverantwortung zu entwickeln, und Misserfolge werden als Lernerfahrungen angesehen. Experimentieren wird zu einem Eckpfeiler der Unternehmenskultur. Manager und Bereichsleiter haben nicht das Gefühl, dass sie in allen Aspekten bis ins kleinste Detail gemanagt werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

1. Entwickeln Sie eine Kultur, in der damit gerechnet wird, dass Fehler auftreten können.
2. Definieren Sie klare Verantwortlichkeiten und Zuständigkeiten für verschiedene Funktionsbereiche innerhalb der Organisation.
3. Vermitteln Sie Eigenverantwortung und Rechenschaftspflicht, damit alle wissen, wo sie bei dezentralen Entscheidungen Unterstützung erhalten können.
4. Definieren Sie unumkehrbare und leicht revidierbare Entscheidungen, damit die Mitarbeiter wissen, wann sie Beschlüsse an höhere Führungsebenen eskalieren müssen.

5. Schaffen Sie in der Organisation ein Bewusstsein dafür, dass alle Mitarbeiter in der Lage sind, auf verschiedenen Ebenen Maßnahmen zu ergreifen, wenn Ergebnisse gefährdet sind. Stellen Sie Ihren Teammitgliedern Unterlagen über Governance, Befugnisebenen, Tools sowie Möglichkeiten zur Verfügung, um die erforderlichen Fähigkeiten für eine effektive Reaktion zu üben.
6. Geben Sie Ihren Teammitgliedern die Möglichkeit, die notwendigen Fähigkeiten zu üben, um auf verschiedene Entscheidungen zu reagieren. Sobald die Entscheidungsebenen festgelegt sind, führen Sie GameDays durch, um sicherzustellen, dass alle Mitarbeiter den Prozess verstehen und umsetzen können.
 - a. Stellen Sie alternative sichere Umgebungen bereit, in denen Prozesse und Verfahren getestet und eingeübt werden können.
 - b. Erkennen Sie an und schaffen Sie ein Bewusstsein dafür, dass Teammitglieder befugt sind, Maßnahmen zu ergreifen, wenn das Ergebnis ein vordefiniertes Risikoniveau aufweist.
 - c. Verschaffen Sie den Teammitgliedern die erforderliche Autorität, um Maßnahmen zu ergreifen, indem Sie ihnen Berechtigungen und Zugriff auf ihre Workloads und Komponenten geben.
7. Bieten Sie Teams die Möglichkeit, ihre Erfahrungen (betriebliche Erfolge und Misserfolge) auszutauschen.
8. Ermöglichen Sie Teams, den Status quo in Frage zu stellen, und stellen Sie Mechanismen zur Verfügung, mit denen Verbesserungen sowie deren Auswirkungen auf die Organisation verfolgt und gemessen werden können.

Aufwand für den Implementierungsplan: Mittel

Ressourcen

Zugehörige bewährte Methoden:

- [OPS01-BP06 Bewerten von Kompromissen und Abwägen der Vorteile und Risiken](#)
- [OPS02-BP05 Mechanismen zur Identifizierung von Verantwortlichkeit und Eigentümerschaft sind vorhanden](#)

Zugehörige Dokumente:

- [AWS-Blog-Beitrag | Das agile Unternehmen](#)
- [AWS-Blog-Beitrag | Erfolg messen: Ein Paradoxon und ein Plan](#)
- [AWS-Blog-Beitrag | Loslassen: Autonomie in Teams ermöglichen](#)

- [Zentralisieren oder Dezentralisieren?](#)

Zugehörige Videos:

- [re:Invent 2023 | So sabotieren Sie Ihre Transformation nicht \(SEG201\)](#)
- [re:Invent 2021 | Die Amazon Builders' Library: Betriebliche Exzellenz von Amazon](#)
- [Zentralisierung und Dezentralisierung im Vergleich](#)

Zugehörige Beispiele:

- [Verwendung von Aufzeichnungen über architektonische Entscheidungen zur Optimierung der technischen Entscheidungsfindung für ein Softwareentwicklungsprojekt](#)

OPS03-BP03 Eskalation wird gefördert

Die Teammitglieder werden von der Führung ermutigt, Probleme und Bedenken an übergeordnete Entscheidungsträger und Stakeholder zu eskalieren, wenn sie der Meinung sind, dass die gewünschten Ergebnisse gefährdet sind und die erwarteten Standards nicht erfüllt werden. Dies ist ein Feature der Organisationskultur und wird auf allen Ebenen vorangetrieben. Die Eskalation sollte frühzeitig und lieber zu oft vorgenommen werden, damit Risiken identifiziert und Vorfälle verhindert werden können. Die Führung tadelt Mitarbeiter nicht dafür, wenn sie ein Problem eskalieren.

Gewünschtes Ergebnis: Personen in der gesamten Organisation sind vertraut damit, Probleme an ihre unmittelbaren und höheren Führungsebenen zu eskalieren. Die Führung hat bewusst und gezielt die Erwartung aufgestellt, dass sich ihre Teams sicher fühlen sollen, Probleme zu eskalieren. Es wurde ein Mechanismus eingerichtet, um Probleme auf allen Organisationsebenen zu eskalieren. Wenn Mitarbeiter eine Angelegenheit an ihren Vorgesetzten eskalieren, entscheiden sie gemeinsam über das Ausmaß der Auswirkungen und eine mögliche Eskalation des Problems. Eine Eskalation setzt voraus, dass die Mitarbeiter einen empfohlenen Arbeitsplan zur Behebung des Problems beifügen. Wenn die nächsthöhere Führungsebene nicht rechtzeitig Maßnahmen ergreift, sind die Mitarbeiter angehalten, Probleme an die oberste Führungsebene weiterzuleiten, wenn sie der festen Überzeugung sind, dass die Risiken für die Organisation eine Eskalation rechtfertigen.

Typische Anti-Muster:

- Führungskräfte haken während Ihrer Statusbesprechung zum Cloud-Transformationsprogramm nicht ausreichend nach, um herauszufinden, wo Probleme und Hindernisse auftreten. Stattdessen

werden nur gute Nachrichten präsentiert. Sie CIO hat deutlich gemacht, dass sie nur gerne gute Nachrichten hört, da sie aufgrund der aufgeworfenen Probleme CEO glauben, dass das Programm scheitert.

- Sie sind als Cloud-Betriebsentwickler tätig und stellen fest, dass das neue Wissensmanagementsystem von den Anwendungsteams kaum verwendet wird. Das Unternehmen investierte ein Jahr und mehrere Millionen Dollar in die Implementierung eines neuen Wissensmanagementsystems, aber die Mitarbeiter verfassen ihre Runbooks noch immer lokal und teilen sie in einer internen Cloud-Umgebung, was die Suche nach Wissen erschwert, das für unterstützte Workloads relevant ist. Sie versuchen, die Führungskräfte darauf aufmerksam zu machen, da die konsequente Verwendung dieses Systems die betriebliche Effizienz verbessern kann. Als Sie das Problem der Bereichsleiterin vorlegen, die für die Implementierung des Wissensmanagementsystems zuständig ist, werden Sie von ihr kritisiert, weil dadurch die Investition in Frage gestellt wird.
- Das für die Absicherung der Rechenressourcen zuständige Infosec-Team hat beschlossen, einen Prozess einzuführen, bei dem die erforderlichen Scans durchgeführt werden müssen, um sicherzustellen, dass die EC2 Instanzen vollständig gesichert sind, bevor das Compute-Team die Ressource zur Nutzung freigibt. Dadurch kam es zu einer Zeitverzögerung von einer zusätzlichen Woche für die Bereitstellung von Ressourcen, wodurch deren SLA Verfügbarkeit beeinträchtigt wird. Das Computing-Team hat Angst, dies über die Cloud an den VP zu eskalieren, da der VP für Informationssicherheit dadurch in ein schlechtes Licht gerückt werden könnte.

Vorteile der Nutzung dieser bewährten Methode:

Komplexe oder kritische Probleme werden angegangen, bevor sie sich auf das Geschäft auswirken. Es wird weniger Zeit verschwendet. Risiken werden minimiert. Teams werden bei der Lösung von Problemen proaktiver und ergebnisorientierter.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Die Bereitschaft und Fähigkeit, auf allen Organisationsebenen uneingeschränkt zu eskalieren, ist eine bedeutende Eigenschaft der Organisation und ihrer Kultur, die bewusst weiterentwickelt werden sollte, und zwar durch gezielte Schulungen, Kommunikationen der Führungsebene, Erwartungssetzung und den Einsatz von Mechanismen auf allen Organisationsebenen.

Implementierungsschritte

1. Definieren Sie Richtlinien, Standards und Erwartungen für Ihre Organisation.
 - a. Sorgen Sie für eine breite Anwendung und Kenntnis der Richtlinien, Erwartungen und Standards.
2. Ermutigen, schulen und befähigen Sie die Mitarbeiter, damit sie frühzeitig und häufig eskalieren, wenn die Standards nicht eingehalten werden.
3. Bekräftigen Sie in der Organisation, dass die frühe und häufige Eskalation die bewährte Methode ist. Akzeptieren Sie im Unternehmen, dass sich Eskalationen zwar als unbegründet herausstellen können, es sich aber trotzdem insgesamt lohnt, wenn ein echter Vorfall dadurch verhindert wird.
 - a. Entwickeln Sie einen Mechanismus für Eskalationen (z. B. ein Andon-Cord-System).
 - b. Sorgen Sie für dokumentierte Verfahren, die definieren, wann und wie eine Eskalation erfolgen soll.
 - c. Definieren Sie die Abfolge der Personen mit zunehmenden Befugnissen, um Maßnahmen zu ergreifen oder zu genehmigen, sowie die Kontaktinformationen der einzelnen Stakeholder.
4. Im Falle einer Eskalation sollte sie so lange fortgesetzt werden, bis das Teammitglied davon überzeugt ist, dass das Risiko durch entsprechende Maßnahmen der Führung gemindert wurde.
 - a. Eskalationen sollten Folgendes beinhalten:
 - i. Beschreibung der Situation und Art des Risikos
 - ii. Kritikalität der Situation
 - iii. Wer oder was betroffen ist
 - iv. Umfang der Auswirkungen
 - v. Dringlichkeit, falls eine Auswirkung eintritt
 - vi. Vorgeschlagene Abhilfemaßnahmen und Risikominderungsplan
 - b. Schützen Sie Mitarbeiter, die ein Problem eskalieren. Führen Sie eine Richtlinie ein, die Teammitglieder vor Konsequenzen schützt, wenn sie an einen ablehnend eingestellten Entscheidungsträger oder Stakeholder eskalieren. Schaffen Sie Mechanismen, um solche Szenarien zu erkennen, und leiten Sie entsprechende Maßnahmen ein.
5. Fördern Sie eine Kultur der kontinuierlichen Verbesserung durch Feedback-Schleifen in allen Bereichen der Organisation. Feedback-Schleifen fungieren als kleine Eskalationen an die verantwortlichen Personen und identifizieren Verbesserungsmöglichkeiten, auch wenn eine Eskalation nicht erforderlich ist. Eine Kultur der kontinuierlichen Verbesserung zwingt alle dazu, proaktiver zu werden.

6. Die Führung sollte regelmäßig an die Richtlinien, Standards und Mechanismen erinnern sowie an den Wunsch nach offener Eskalation und kontinuierlichen Feedback-Schleifen ohne Vergeltungsmaßnahmen jedweder Art.

Aufwand für den Implementierungsplan: Mittel

Ressourcen

Zugehörige bewährte Methoden:

- [OPS02-BP05 Mechanismen zum Anfordern von Ergänzungen, Änderungen und Ausnahmen sind vorhanden](#)

Zugehörige Dokumente:

- [Wie fördert man eine Kultur der kontinuierlichen Verbesserung und des Lernens von Andon- und Eskalationssystemen?](#)
- [Das Andon-Cord \(IT-Revolution\)](#)
- [AWS DevOps Leitlinien | Etablieren Sie klare Eskalationspfade und fördern Sie konstruktive Meinungsverschiedenheiten](#)

Zugehörige Videos:

- [Jeff Bezos erklärt, wie man Entscheidungen trifft \(und die Geschwindigkeit erhöht\)](#)
- [Toyota Product System: Anhalten der Produktion, ein Knopf und einer Andon-Elektroplatine](#)
- [Andon Cord in der Fertigung LEAN](#)

Zugehörige Beispiele:

- [Arbeiten mit Eskalationsplänen in Incident Manager](#)

OPS03-BP04 Die Kommunikation ist zeitnah, klar und umsetzbar

Die Führung ist für eine überzeugende und effektive Kommunikation zuständig, insbesondere wenn die Organisation vor der Einführung neuer Strategien, Technologien oder Arbeitsweisen steht. Führungskräfte sollten Erwartungen an alle Mitarbeiter stellen, damit sie auf die Unternehmensziele hinarbeiten können. Entwickeln Sie Kommunikationsmechanismen für die Bildung und

Aufrechterhaltung des geforderten Bewusstseins in Teams, die für die Durchführung von Plänen verantwortlich sind, die von der Führung finanziert und unterstützt werden. Machen Sie sich die organisationsübergreifende Vielfalt zunutze und hören Sie sich verschiedene einzigartige Perspektiven aufmerksam an. Nutzen Sie diese Perspektive, um Innovation zu fördern, Ihre Annahmen in Frage zu stellen und das Risiko einer Verzerrung durch automatische Bestätigung zu reduzieren. Stärken Sie Inklusion, Vielfalt und Zugehörigkeit innerhalb Ihrer Teams, um nützliche Perspektiven zu gewinnen.

Gewünschtes Ergebnis: Ihre Organisation entwickelt Kommunikationsstrategien, um den Auswirkungen von Veränderungen auf die Organisation Rechnung zu tragen. Die Teams werden informiert und motiviert, weiter miteinander statt gegeneinander zu arbeiten. Einzelpersonen kennen die Bedeutung ihrer Rolle, um die angegebenen Ziele zu erreichen. E-Mail ist nur ein passiver Kommunikationsmechanismus und wird als solcher behandelt. Das Management verbringt Zeit mit seinen einzelnen Mitarbeitern, um ihnen ihre Verantwortung, die zu erledigenden Aufgaben und die Bedeutung ihrer Arbeit für die Gesamtmision zu vermitteln. Bei Bedarf binden Führungskräfte ihre Mitarbeiter an kleineren Veranstaltungsorten direkt ein, um Botschaften zu kommunizieren, und sie stellen sicher, dass diese Botschaften effektiv übermittelt werden. Die Organisation erfüllt oder übertrifft die Erwartungen der Führung mithilfe geeigneter Kommunikationsstrategien. Die Führung begrüßt und fördert unterschiedliche Meinungen innerhalb und zwischen Teams.

Typische Anti-Muster:

- Ihre Organisation hat einen Fünf-Jahres-Plan für die Migration aller Workloads in AWS. Der Business Case für die Cloud beinhaltet die Modernisierung von 25 % aller Workloads, um die Vorteile der Serverless-Technologie zu nutzen. Der CIO kommuniziert diese Strategie direkt unterstellten Mitarbeitern und erwartet, dass die Führungskräfte diese Präsentation ohne persönliche Gespräche an Manager, Bereichsleiter und einzelne Mitarbeiter weiterleiten. Der CIO zieht sich zurück und erwartet, dass seine Organisation die neue Strategie umsetzt.
- Die Führung bietet oder nutzt keine Feedback-Mechanismen, und die Erwartungslücke wächst, was dazu führt, dass einzelne Projekte ins Stocken geraten.
- Sie werden gebeten, eine Änderung an Ihren Sicherheitsgruppen vorzunehmen, ohne konkrete Informationen über die Änderung zu erhalten oder darüber, welche Auswirkungen sie auf alle Workloads haben könnte und bis wann sie umzusetzen ist. Der Manager leitet eine E-Mail vom VP von InfoSec weiter und fügt folgende Nachricht hinzu: „Setzen Sie das um.“
- An Ihrer Migrationsstrategie wurden Änderungen vorgenommen, die die Anzahl der geplanten Modernisierungen von 25 auf 10 % reduzieren. Dies hat nachgelagerte Auswirkungen auf die Betriebsorganisation. Sie wurden nicht über diese strategische Änderung informiert und verfügen

daher nicht über genügend qualifizierte Mitarbeiter, um einen größeren Lift-and-Shift-Aufwand von Workloads in AWS zu bewältigen.

Vorteile der Nutzung dieser bewährten Methode:

- Ihre Organisation ist über neue oder geänderte Strategien hinreichend informiert und die Mitarbeiter sind hochmotiviert, um sich gegenseitig dabei zu unterstützen, die von der Führung festgelegten Gesamtziele und Metriken zu erreichen.
- Es gibt Mechanismen und sie werden angewandt, um Teammitglieder rechtzeitig über bekannte Risiken und geplante Ereignisse zu informieren.
- Neue Arbeitsweisen (einschließlich Änderungen bzgl. Belegschaft, Organisation, Prozessen oder Technologien) werden zusammen mit den erforderlichen Fähigkeiten von der Organisation effektiver übernommen. Darüber hinaus erreicht Ihre Organisation schneller Geschäftsvorteile.
- Die Teammitglieder verfügen über die notwendigen Hintergrundinformationen zu den eingehenden Kommunikationen und können ihre Arbeit effektiver erledigen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Zur Implementierung dieser bewährten Methode müssen Sie mit Beteiligten aus der gesamten Organisation zusammenarbeiten, um Kommunikationsstandards zu vereinbaren. Machen Sie diese Standards in der Organisation bekannt. Bei allen wichtigen IT-Umstellungen kann ein etabliertes Planungsteam die Auswirkungen der Änderungen auf seine Mitarbeiter erfolgreicher bewältigen als eine Organisation, die diese Methode nicht anwendet. In größeren Organisationen können Veränderungen schwieriger umzusetzen sein, da es auf eine hohe Zustimmung aller einzelnen Mitarbeiter zu einer neuen Strategie ankommt. In Ermangelung eines solchen Umstellungsplanungsteams trägt die Führung zu 100 % die Verantwortung für eine effektive Kommunikation. Wenn Sie ein Umstellungsplanungsteam einrichten, weisen Sie die Teammitglieder an, mit der gesamten Organisationsführung zusammenzuarbeiten, um eine effektive Kommunikation auf allen Ebenen zu definieren und zu gewährleisten.

Kundenbeispiel

AnyCompany Retail hat sich für den AWS Enterprise Support registriert und ist für seine Cloud-Betriebsabläufe auf andere Drittanbieter angewiesen. Das Unternehmen nutzt Chat und Chatops als zentrales Kommunikationsmedium für seine betrieblichen Aktivitäten. Für Warnmeldungen und

andere Informationen werden spezielle Kanäle genutzt. Wenn eine Maßnahme erforderlich ist, wird das erwartete Ergebnis klar formuliert, und in vielen Fällen gibt es ein Runbook oder Playbook dafür. Das Unternehmen verwendet einen Änderungskalender für die Planung größerer Änderungen an Produktionssystemen.

Implementierungsschritte

1. Richten Sie innerhalb der Organisation ein Kernteam ein, das für die Erstellung und Initiierung von Kommunikationsplänen für Änderungen verantwortlich ist, die auf mehreren Ebenen innerhalb der Organisation stattfinden.
2. Fordern Sie Eigenverantwortlichkeit, um ein hohes Maß an Übersicht zu fördern. Geben Sie den einzelnen Teams die Möglichkeit, unabhängig voneinander Innovationen zu entwickeln, und sorgen Sie für einen ausgewogenen Einsatz einheitlicher Mechanismen, die das richtige Maß an Einsicht und Zielgerichtetheit ermöglichen.
3. Arbeiten Sie mit allen Stakeholdern in Ihrer Organisation zusammen, um Kommunikationsstandards, -methoden und -pläne zu vereinbaren.
4. Stellen Sie sicher, dass das zentrale Kommunikationsteam mit der Organisations- und Programmleitung zusammenarbeitet, um im Namen der Führungskräfte Botschaften an die zuständigen Mitarbeiter zu verfassen.
5. Entwickeln Sie strategische Kommunikationsmechanismen, um Veränderungen mithilfe von Ankündigungen, gemeinsamen Kalendern, Besprechungen mit allen Mitarbeitern und persönlichen oder Einzelgesprächen zu bewältigen, sodass die Teammitglieder die richtigen Erwartungen bezüglich der zu ergreifenden Maßnahmen haben.
6. Stellen Sie den erforderlichen Kontext, Details und die nötige Zeit bereit (wenn möglich), um festzustellen, ob Maßnahmen erforderlich sind. Wenn Maßnahmen erforderlich sind, identifizieren Sie die erforderlichen Maßnahmen und deren Auswirkungen.
7. Implementieren Sie Tools, die eine taktische Kommunikation fördern, z. B. interne Chats, E-Mails und Wissensmanagement.
8. Implementieren Sie Mechanismen, um zu messen und zu überprüfen, ob mit allen Kommunikationen die gewünschten Ergebnisse erreicht werden.
9. Richten Sie eine Feedback-Schleife ein, die die Effektivität aller Kommunikationen misst, insbesondere wenn darin der Widerstand gegen Veränderungen in der Organisation thematisiert wird.

- 10 Richten Sie für ganz AWS-Konten [alternative Kontakte](#) für Abrechnung, Sicherheit und Betrieb ein. Idealerweise sollte es sich bei diesen Kontakten um E-Mail-Verteilerlisten und nicht um Einzelpersonen handeln.
- 11 Erstellen Sie einen Kommunikationsplan für die Eskalation und die umgekehrte Eskalation, um mit Ihren internen und externen Teams, einschließlich AWS Support und anderen Drittanbietern, zusammenzuarbeiten.
- 12 Initiieren Sie Kommunikationsstrategien und setzen Sie sie während der gesamten Laufzeit jedes Transformationsprogramms konsequent um.
- 13 Priorisieren Sie Maßnahmen, die nach Möglichkeit wiederholbar sind, um sie sicher und in großem Maßstab zu automatisieren.
- 14 Wenn Kommunikation in Szenarien mit automatisierten Maßnahmen erforderlich ist, sollte die Kommunikation hauptsächlich der Information der Teams oder Audits dienen oder Teil des Änderungsverwaltungsprozesses sein.
- 15 Analysieren Sie die Kommunikation Ihrer Warnsysteme auf Fehlalarme oder Warnmeldungen, die ständig generiert werden. Entfernen Sie diese Warnmeldungen oder ändern Sie sie so, dass sie nur ausgelöst werden, wenn menschliches Eingreifen erforderlich ist. Stellen Sie ein Runbook oder Playbook bereit, wenn eine Warnmeldung ausgelöst wird.
 - a. Sie können [AWSSystems Manager-Dokumente](#) verwenden, um Playbooks und Runbooks für Warnmeldungen zu erstellen.
- 16 Es gibt Mechanismen zur Benachrichtigung über Risiken oder geplante Ereignisse auf eine klare und unterstützende Weise mit ausreichend Zeit für geeignete Maßnahmen. Verwenden Sie E-Mail-Listen oder Chat-Kanäle zum Senden von Benachrichtigungen vor geplanten Ereignissen.
 - a. [AWS Chatbot](#) kann verwendet werden, um Warnmeldungen zu senden und auf Ereignisse innerhalb der Messaging-Plattform Ihrer Organisation zu reagieren.
- 17 Stellen Sie eine zugängliche Informationsquelle bereit, der geplante Ereignisse zu entnehmen sind. Stellen Sie Benachrichtigungen zu geplanten Ereignissen vom gleichen System bereit.
 - a. [AWS Systems Manager-Änderungskalender](#) kann verwendet werden, um Änderungszeitfenster zu erstellen, in denen Änderungen vorgenommen werden können. Dadurch werden Teammitglieder benachrichtigt, wann sie in sicherer Weise Änderungen vornehmen können.
- 18 Überwachen Sie Benachrichtigungen zu Schwachstellen und Patch-Informationen, um bestehende Schwachstellen und potenzielle Risiken im Zusammenhang mit den Komponenten Ihrer Workloads zu verstehen. Stellen Sie Benachrichtigungen für die Teammitglieder bereit, damit sie Maßnahmen ergreifen können.

- a. Sie können [AWS-Sicherheitsmitteilungen](#) abonnieren, um über Schwachstellen in AWS benachrichtigt zu werden.

19 Berücksichtigen unterschiedlicher Meinungen und Perspektiven: Ermutigen Sie alle dazu, Beiträge zu leisten. Geben Sie unterrepräsentierten Gruppen die Möglichkeit, sich in die Kommunikation einzubringen. Rotieren Sie die Rollen und Zuständigkeiten in Meetings.

- a. Erweitern von Rollen und Zuständigkeiten: Bieten Sie Teammitgliedern die Möglichkeit, Rollen zu übernehmen, die ihnen fremd sind. Auf diese Weise können sie Erfahrung sammeln und neue Perspektiven durch die Rolle und den resultierenden Austausch mit neuen Teammitgliedern gewinnen, zu denen sie möglicherweise andernfalls keinen Kontakt hätten. Nicht zuletzt können sie die neue Rolle und die Teammitglieder mit ihren Erfahrungen und Perspektiven bereichern. Mit zunehmender Erfahrung werden Sie aufkommende Geschäftsmöglichkeiten oder neue Verbesserungsmöglichkeiten identifizieren. Rotieren Sie allgemeine Aufgaben zwischen den Mitgliedern innerhalb eines Teams, die normalerweise anderen Tätigkeiten nachgehen, damit sie deren Anforderungen und Auswirkungen verstehen.
- b. Bereitstellen einer sicheren und freundlichen Umgebung: Richten Sie Richtlinien und Kontrollen zum Schutz der geistigen und physischen Sicherheit der Teammitglieder in Ihrer Organisation ein. Die Teammitglieder müssen ohne Angst vor Vergeltungsmaßnahmen zusammenarbeiten können. Wenn sich Teammitglieder sicher und willkommen fühlen, ist die Wahrscheinlichkeit höher, dass sie engagiert und produktiv bleiben. Je vielfältiger Ihre Organisation ist, desto besser verstehen Sie die Personen, die Sie unterstützen, einschließlich Ihrer Kunden. Wenn Ihre Teammitglieder zufrieden sind, ihre Meinung sagen können und sich ernst genommen fühlen, steigt die Wahrscheinlichkeit, dass sie wertvolle Erkenntnisse mitteilen (z. B. Marketingmöglichkeiten, erforderliche Maßnahmen zur Barrierefreiheit, unerschlossene Marktsegmente, unbehandelte Risiken in Ihrer Umgebung).
- c. Ermuntern von Teammitgliedern zu vollständigen Teilnahme: Stellen Sie die Ressourcen bereit, die Ihre Mitarbeiter benötigen, um alle arbeitsbezogenen Tätigkeiten auszuführen. Teammitglieder haben Fähigkeiten entwickelt, mit denen sie ihre täglichen Herausforderungen meistern. Diese einzigartigen Fähigkeiten können für Ihre Organisation von großem Vorteil sein. Wenn Sie die Teammitglieder mit den notwendigen Ressourcen ausstatten, können Sie den Nutzen ihrer Beiträge maximieren.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS03-BP01 Förderung durch die Geschäftsführung gewährleisten](#)

- [OPS07-BP03 Verwenden von Runbooks zur Durchführung von Verfahren](#)
- [OPS07-BP04 Verwenden von Playbooks zum Untersuchen von Problemen](#)

Zugehörige Dokumente:

- [AWS-Blog-Post | Rechenschaftspflicht und Befähigung sind der Schlüssel zu leistungsstarken agilen Organisationen](#)
- [AWS Executive Insights | Lernen Sie, Innovation statt Komplexität zu skalieren | Single-Threaded Leaders](#)
- [AWS-Sicherheitsberichte](#)
- [Offene CVE](#)
- [Support-App in Slack zur Verwaltung von Supportfällen](#)
- [Verwaltung von AWS-Ressourcen in Ihren Slack-Kanälen mit Amazon Q Developer in Chat-Anwendungen](#)

Zugehörige Services:

- [Amazon Q Developer in Chat-Anwendungen](#)
- [AWS Systems Manager-Änderungskalender](#)
- [AWS Systems Manager-Dokumente](#)

OPS03-BP05 Experimentieren wird empfohlen

Experimente können Katalysatoren für die Umsetzung von Ideen in Produkte und Funktionen sein. Sie beschleunigen Lernprozesse und halten Teammitglieder interessiert und engagiert. Team-Mitglieder sollten oft experimentieren, um Innovationen voranzubringen. Selbst nicht erwünschte Ergebnisse bieten den Vorteil, dass man dadurch weiß, wie man nicht vorgehen sollte. Teammitglieder werden nicht für erfolgreiche Experimente mit unerwünschten Ergebnissen bestraft.

Gewünschtes Ergebnis:

- Ihre Organisation ermutigt zum Experimentieren, um Innovationen voranzubringen.
- Experimente werden genutzt, um daraus zu lernen.

Typische Anti-Muster:

- Sie möchten einen A/B-Test durchführen, es gibt jedoch keinen Mechanismus für das Experiment. Sie stellen eine UI-Änderung bereit, ohne diese testen zu können. Dies beeinträchtigt den Kundenkomfort.
- Ihr Unternehmen verfügt nur über eine Staging- und eine Produktionsumgebung. Es gibt keine Sandbox-Umgebung zum Experimentieren mit neuen Funktionen oder Produkten, weshalb Sie in der Produktionsumgebung experimentieren müssen.

Vorteile der Nutzung dieser bewährten Methode:

- Experimente bringen Innovationen voran.
- Mithilfe von Experimenten können Sie schneller auf Feedback reagieren.
- Ihre Organisation entwickelt eine Lernkultur.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Experimente sollten in sicherer Weise durchgeführt werden. Nutzen Sie mehrere Umgebungen für Experimente, ohne dabei Produktionsressourcen in Gefahr zu bringen. Nutzen Sie A/B-Tests und Feature-Flags für Testexperimente. Geben Sie Teammitgliedern die Möglichkeit, Experimente in einer Sandbox-Umgebung durchzuführen.

Kundenbeispiel

AnyCompany Retail ermuntert seine Mitarbeiter zu Experimenten. Teammitglieder können 20 % ihrer wöchentlichen Arbeitszeit für Experimente oder zum Erlernen neuer Technologien nutzen. Es gibt eine Sandbox-Umgebung zum Ausprobieren von Innovationen. Für neue Funktionen werden A/B-Tests verwendet, um sie mit realem Benutzerfeedback zu prüfen.

Implementierungsschritte

1. Arbeiten Sie mit Führungskräften aus dem gesamten Unternehmen zusammen, um Experimente zu unterstützen. Teammitglieder sollten aufgefordert werden, Experimente in sicherer Weise durchzuführen.
2. Stellen Sie Ihren Teammitgliedern eine Umgebung zur Verfügung, in der sie in sicherer Weise experimentieren können. Sie müssen Zugriff auf eine Umgebung haben, die der Produktionsumgebung stark ähnelt.

- a. Sie können ein separates AWS-Konto verwenden, um eine Sandbox-Umgebung für Experimente zu erstellen. [AWS Control Tower](#) kann verwendet werden, um diese Konten bereitzustellen.
3. Verwenden Sie Feature-Flags und A/B-Tests, um in sicherer Weise zu experimentieren und Benutzer-Feedback einzuholen.
 - a. [AWS AppConfig Feature Flags](#) bietet die Möglichkeit, Feature-Flags zu erstellen.
 - b. Mithilfe von [AWS Lambda-Versionen](#) können Sie eine neue Version einer Funktion für Beta-Tests bereitstellen.

Aufwand für den Implementierungsplan: Hoch. Die Bereitstellung einer Umgebung für Teammitglieder, in der sie in sicherer Weise experimentieren können, kann erhebliche Investitionen erfordern. Möglicherweise muss auch der Anwendungscode modifiziert werden, um Feature-Flags verwenden oder A/B-Tests unterstützen zu können.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS11-BP02 Führen Sie eine Analyse nach dem Vorfall durch](#) – Das Lernen aus Vorfällen ist neben Experimenten ein wichtiger Motor für Innovation.
- [OPS11-BP03 Implementieren Sie Feedback-Schleifen](#) – Feedbackschleifen sind ein wichtiger Bestandteil von Experimenten.

Zugehörige Dokumente:

- [Ein Einblick in die Amazon-Kultur: Experimente, Misserfolge und Kundenorientierung](#)
- [Bewährte Methoden zur Erstellung und Verwaltung von Sandbox-Konten in AWS](#)
- [Schaffen Sie eine Kultur des Experimentierens, die durch die Cloud ermöglicht wird](#)
- [Ermöglichen von Experimenten und Innovationen in der Cloud bei SulAmérica Seguros](#)
- [Mehr experimentieren, weniger scheitern](#)
- [Organisieren Sie Ihre AWS-Umgebung mit mehreren Konten – Sandbox OE](#)
- [Verwenden von AWS AppConfig Feature Flags](#)

Zugehörige Videos:

- [AWS On Air mit Amazon CloudWatch Evidently | AWS Events](#)
- [AWS On Air San Fran Summit 2022 – Vorstellung der AWS AppConfig Feature Flags-Integration mit Jira](#)
- [AWS re:Invent 2022 – Eine Bereitstellung ist kein Release: Steuern Sie Ihre Einführungen mit Feature-Flags \(BOA305-R\)](#)
- [Programmgesteuerte AWS-Konto-Erstellung mit AWS Control Tower](#)
- [Einrichtung einer AWS-Umgebung mit mehreren Konten, in der bewährte Methoden für AWS Organizations genutzt werden](#)

Zugehörige Beispiele:

- [AWS Innovation Sandbox](#)
- [Umfassende Personalisierung für E-Commerce – Einführung](#)

Zugehörige Services:

- [Amazon CloudWatch Evidently](#)
- [AWS AppConfig](#)
- [AWS Control Tower](#)

OPS03-BP06 Teammitglieder werden ermutigt, ihre Fähigkeiten zu pflegen und zu erweitern

Teams müssen ihre Fähigkeiten ausbauen, um neue Technologien nutzen und mit veränderten Anforderungen und Aufgaben Ihrer Workloads umgehen zu können. Neue Fähigkeiten im Umgang mit neuen Technologien erhöhen oftmals die Zufriedenheit der Teammitglieder und ermöglichen Innovationen. Unterstützen Sie Ihre Teammitglieder beim Erlangen und Bewahren von Branchenzertifizierungen, mit denen ihre wachsenden Fähigkeiten bestätigt und anerkannt werden. Führen Sie funktionsübergreifende Trainings durch, um den Wissenstransfer zu fördern und das Risiko signifikanter Auswirkungen zu reduzieren, wenn Sie qualifizierte und erfahrene Teammitglieder mit kritischem Wissen verlieren. Schaffen Sie spezielle strukturierte Lernzeiten.

AWS stellt Ressourcen bereit, die Anleitungen, Beispiele und detaillierte Anleitungen zur Schulung Ihrer Teams bieten. Dazu zählen beispielsweise das [Erste Schritte-Ressourcencenter von AWS](#), [AWS-Blogs](#), [AWS Online Tech Talks](#), [AWS-Veranstaltungen und -Webinare](#) und die [AWS Well-Architected Labs](#).

Ressourcen wie [Support](#), [AWS re:Post](#), [Support Center](#) und die [AWS Dokumentation](#) helfen dabei, technische Hindernisse zu beseitigen und den Betrieb zu verbessern. Bei Fragen können Sie sich über das Support Center an den Support wenden.

AWS stellt in der [Amazon Builders' Library](#) auch bewährte Methoden und Muster vor, die wir durch den Betrieb von AWS gelernt haben. Außerdem finden Sie im [AWS-Blog](#) und im [offiziellen AWS-Podcast](#) eine Vielzahl weiterer nützlicher Lernmaterialien.

[AWS Training and Certification](#) bietet kostenlose Schulungen in Form von digitalen Kursen im Selbststudium sowie Lernpläne für spezifische Rollen oder Bereiche. Sie können sich auch für eine Schulung mit Kursleiter registrieren, um die AWS-Fähigkeiten Ihres Teams auszubauen.

Gewünschtes Ergebnis: Ihre Organisation bewertet ständig Qualifikationslücken und schließt sie mit strukturierten Budgets und Investitionen. Die Teams ermutigen und unterstützen ihre Mitglieder durch Weiterbildungsaktivitäten wie den Erwerb führender Branchenzertifizierungen. Die Teams nutzen spezielle Programme zum Wissensaustausch wie informelle Schulungen, Immersion Days, Hackathons und GameDays. Ihre Organisation hält ihre Wissenssysteme auf dem aktuellen Stand und relevant für die Schulung von Teammitgliedern, einschließlich Schulungen zur Einarbeitung neuer Mitarbeiter.

Typische Anti-Muster:

- Aufgrund eines fehlenden strukturierten Trainingsprogramms und Budgets entstehen in den Teams Unsicherheit und Zweifel, wenn sie versuchen, mit der technologischen Entwicklung Schritt zu halten, was letztlich zu einer erhöhten Personalabwanderung führt.
- Im Rahmen der Migration zu AWS weist Ihre Organisation Qualifikationslücken auf und die Teams verfügen über unterschiedlich starke Cloud-Kompetenzen. Aufgrund fehlender Fortbildungsprogramme sehen sich die Teams mit der veralteten und ineffizienten Verwaltung der Cloud-Umgebung überfordert, was zu einer Mehrbelastung der Mitarbeiter führt. Diese erschwerten Arbeitsbedingungen erhöhen die Unzufriedenheit der Mitarbeiter.

Vorteile der Nutzung dieser bewährten Methode: Wenn Ihre Organisation bewusst in die Verbesserung der Fähigkeiten der Teams investiert, trägt dies auch dazu bei, die Einführung und Optimierung der Cloud zu beschleunigen und zu skalieren. Gezielte Lernprogramme fördern Innovationen und stärken die operativen Fähigkeiten der Teams, um auf Ereignisse vorbereitet zu sein. Teams investieren bewusst in die Implementierung und Weiterentwicklung von bewährten Methoden. Die Arbeitsmoral im Team ist hoch und die Teammitglieder sind stolz auf ihren Beitrag zum Unternehmen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Investieren Sie kontinuierlich in die berufliche Weiterentwicklung Ihrer Teams, um neue Technologien einzuführen, Innovationen voranzutreiben und mit den Veränderungen der Anforderungen und Verantwortlichkeiten Schritt zu halten, um Ihre Workloads zu unterstützen.

Implementierungsschritte

1. Nutzen Sie strukturierte Cloud-Advocacy-Programme: [AWS Skills Guild](#) bietet beratende Schulungen an, um das Vertrauen in die eigenen Cloud-Fähigkeiten zu stärken und eine Kultur des kontinuierlichen Lernens zu fördern.
2. Bereitstellen von Ressourcen für die Weiterbildung: Sorgen Sie für eine spezielle strukturierte Lernzeit, Schulungsmaterialien und Laborressourcen. Unterstützen Sie die Teilnahme an Konferenzen und den Zugang zu professionellen Organisationen, die Möglichkeiten zum Lernen von Lehrenden und anderen Fachleuten bieten. Stellen Sie für Ihre Junior-Teammitglieder den Kontakt zu erfahreneren Teammitgliedern als Mentoren her oder ermöglichen Sie Junior-Teammitgliedern, ihnen bei der Arbeit zuzusehen, um sich mit ihren Methoden und Fähigkeiten vertraut zu machen. Ermutigen Sie dazu, auch etwas über Inhalte zu lernen, die nicht direkt mit der Arbeit zusammenhängen, um den Horizont zu erweitern.
3. Ermuntern Sie zur Nutzung von technischen Ressourcen für Experten: Nutzen Sie Ressourcen wie [AWS re:Post](#), um Zugang zu kuratiertem Wissen und einer lebendigen Community zu erhalten.
4. Aufbau und Pflege eines aktuellen Wissens-Repositorys: Nutzen Sie Plattformen für den Wissensaustausch wie Wikis und Runbooks. Erstellen Sie mit [AWS re:Post Private](#) Ihre eigene wiederverwendbare Quelle für Expertenwissen, um die Zusammenarbeit zu optimieren, die Produktivität zu steigern und das Onboarding von Mitarbeitern zu beschleunigen.
5. Teamschulung und teamübergreifende Zusammenarbeit: Planen Sie den Weiterbildungsbedarf Ihrer Teammitglieder ein. Schaffen Sie Gelegenheiten für die Teammitglieder, (vorübergehend oder dauerhaft) in anderen Teams zu arbeiten, damit sie untereinander Fähigkeiten und bewährte Methoden austauschen können, wovon letztendlich die gesamte Organisation profitiert.
6. Unterstützen beim Erlangen und Bewahren von Branchenzertifizierungen: Unterstützen Sie Ihre Teammitglieder beim Erlangen und Bewahren von Branchenzertifizierungen, durch die das Gelernte bestätigt wird und die Erfolge anerkannt werden.

Aufwand für den Implementierungsplan: Hoch

Ressourcen

Zugehörige bewährte Methoden:

- [OPS03-BP01 Förderung durch die Geschäftsführung gewährleisten](#)
- [OPS11-BP04 Wissensmanagement](#)

Zugehörige Dokumente:

- [AWS-Whitepaper | Cloud Adoption Framework: Die Mitarbeiterperspektive](#)
- [Investitionen in kontinuierliches Lernen, um die Zukunft Ihrer Organisation zu fördern](#)
- [AWS Skills Guild](#)
- [AWS Training and Certification](#)
- [Support](#)
- [AWS re:Post](#)
- [AWS-Ressourcencenter für erste Schritte](#)
- [AWS-Blogs](#)
- [AWS Cloud Compliance](#)
- [AWS-Dokumentation](#)
- [Offizieller AWS-Podcast.](#)
- [AWS Online Tech Talks](#)
- [AWS-Veranstaltungen und -Webinare](#)
- [AWS Well-Architected Labs](#)
- [Die Amazon Builders' Library](#)

Zugehörige Videos:

- [AWS re:Invent 2023 | Weiterbildung mit der Geschwindigkeit der Cloud: Aus Mitarbeitern Unternehmer machen](#)
- [WS re:Invent 2023 | Aufbau einer Kultur der Neugier durch Gamification](#)

OPS03-BP07 Ressourcenteams angemessen

Setzen Sie die richtige Anzahl kompetenter Teammitglieder ein und stellen Sie Tools und Ressourcen zur Verfügung, um Ihre Workload-Anforderungen zu erfüllen. Eine Überlastung der Teammitglieder erhöht das Risiko menschlicher Fehler. Investitionen in Tools und Ressourcen wie Automatisierung können die Effektivität Ihres Teams steigern und es dabei unterstützen, eine größere Anzahl von Workloads zu bewältigen, ohne zusätzliche Kapazitäten zu benötigen.

Gewünschtes Ergebnis:

- Sie haben Ihr Team entsprechend personell ausgestattet, um die Fähigkeiten zu erwerben, die es benötigt, um Workloads AWS gemäß Ihrem Migrationsplan zu verwalten. Da sich Ihr Team im Laufe Ihres Migrationsprojekts immer weiter vergrößert hat, hat es Kenntnisse in den AWS Kerntechnologien erworben, die das Unternehmen bei der Migration oder Modernisierung seiner Anwendungen einsetzen will.
- Sie haben Ihren Personalplan sorgfältig abgestimmt, um Ressourcen mithilfe von Automatisierung und Workflows effizient zu nutzen. Ein kleineres Team kann jetzt im Auftrag der Anwendungsentwicklungsteams mehr Infrastruktur verwalten.
- Angesichts sich ändernder betrieblicher Prioritäten werden Personalengpässe proaktiv erkannt, um den Erfolg von Geschäftsinitiativen zu sichern.
- Betriebsmetriken, die auf operative Schwierigkeiten (wie Ermüdung des Bereitschaftsdienstes oder übermäßiges Telefonieren) hinweisen, werden überprüft, um eine Überforderung der Mitarbeiter zu vermeiden.

Typische Anti-Muster:

- Ihre Mitarbeiter haben ihre AWS Fähigkeiten nicht erweitert, als Sie sich Ihrem mehrjährigen Cloud-Migrationsplan nähern, was die Unterstützung der Arbeitslast gefährdet und die Arbeitsmoral der Mitarbeiter beeinträchtigt.
- Ihre gesamte IT-Organisation stellt sich auf agile Arbeitsweisen um. Das Unternehmen priorisiert das Produktportfolio und legt Metriken dafür fest, welche Features zuerst entwickelt werden müssen. Ihr agiler Prozess erfordert nicht, dass Teams ihren Arbeitsplänen Story Points zuweisen. Daher ist es unmöglich zu wissen, welche Kapazitäten für den nächsten Arbeitsschritt erforderlich sind oder ob Sie über die dafür notwendigen Fähigkeiten verfügen.
- Sie beauftragen einen AWS Partner, Ihre Workloads zu migrieren, und Sie haben keinen Plan zur Umstellung auf den Support für Ihre Teams, sobald der Partner das Migrationsprojekt

abgeschlossen hat. Ihre Teams haben Schwierigkeiten, die Workloads effizient und effektiv zu unterstützen.

Vorteile der Nutzung dieser bewährten Methode: In Ihrer Organisation gibt es Teammitglieder, die für die Unterstützung der Workloads qualifiziert sind. Die Ressourcenzuweisung kann an sich ändernde Prioritäten angepasst werden, ohne die Leistung zu beeinträchtigen. Somit können die Teams die Workloads effizient unterstützen und gleichzeitig mehr Zeit mit Innovationen für Kunden aufwenden, was wiederum die Mitarbeiterzufriedenheit erhöht.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Die Ressourcenplanung für Ihre Cloud-Migration sollte auf einer Organisationsebene erfolgen, die Ihrem Migrationsplan sowie dem gewünschten Betriebsmodell entspricht, das zur Unterstützung Ihrer neuen Cloud-Umgebung implementiert wird. Dies erfordert nicht zuletzt ein umfassendes Verständnis, welche Cloud-Technologien für die Geschäfts- und Anwendungsentwicklungsteams eingesetzt werden. Die Infrastruktur- und Betriebsleitung sorgt für eine Analyse von Qualifikationslücken, Schulungen und die Rollendefinition für Ingenieure, die die Cloud-Einführung leiten.

Implementierungsschritte

1. Definieren Sie Erfolgskriterien für den Erfolg des Teams anhand relevanter Betriebsmetriken wie der Mitarbeiterproduktivität (z. B. Kosten für die Unterstützung einer Workload oder Arbeitsstunden, die Mitarbeiter bei Vorfällen aufgewendet haben).
2. Definieren Sie Mechanismen zur Planung und Überprüfung der Kapazität von Ressourcen, um sicherzustellen, dass bei Bedarf ausreichend qualifizierte Ressourcen verfügbar sind und deren Zahl im Laufe der Zeit angepasst werden kann.
3. Schaffen Sie Mechanismen (z. B. das Senden einer monatlichen Umfrage an Teams), um arbeitsbezogene Herausforderungen zu verstehen, die sich auf Teams auswirken (z. B. zunehmende Verantwortlichkeiten, technologische Veränderungen, Personalabwanderung oder wachsende Anzahl unterstützter Kunden).
4. Verwenden Sie diese Mechanismen, um mit Teams in Kontakt zu treten und Trends zu erkennen, die zu Problemen bei der Mitarbeiterproduktivität beitragen können. Wenn sich äußere Faktoren negativ auf Ihre Teams auswirken, bewerten Sie die Ziele neu und passen Sie sie entsprechend an. Identifizieren Sie Hindernisse für den Fortschritt Ihres Teams.

5. Prüfen Sie regelmäßig, ob Ihre derzeit vorhandenen Ressourcen noch ausreichen oder ob zusätzliche Ressourcen benötigt werden, und nehmen Sie entsprechende Anpassungen an den Support-Teams vor.

Aufwand für den Implementierungsplan: Mittel

Ressourcen

Zugehörige bewährte Methoden:

- [OPS03-BP06 Die Teammitglieder werden ermutigt, ihre Fähigkeiten beizubehalten und auszubauen](#)
- [OPS09-BP03 Überprüfen Sie die Betriebskennzahlen und priorisieren Sie Verbesserungen](#)
- [OPS10-BP01 Verwenden Sie einen Prozess für das Ereignis-, Vorfall- und Problemmanagement](#)
- [OPS10-BP07 Automatisieren Sie Reaktionen auf Ereignisse](#)

Zugehörige Dokumente:

- [AWS Cloud Adoptionsrahmen: Aus der Perspektive der Menschen](#)
- [Transformation zu einem zukunftsfähigen Unternehmen](#)
- [Priorisieren der Fähigkeiten Ihrer Mitarbeiter, um das Unternehmenswachstum voranzutreiben](#)
- [Leistungsstarke Organisation – das Zwei-Pizzen-Team von Amazon](#)
- [Wie Unternehmen mit umfassender Cloud-Erfahrung erfolgreich sind](#)

Vorbereitung

Fragen

- [OPS 4. Wie implementieren Sie die Beobachtbarkeit in Ihrer Workload?](#)
- [OPS 5. Wie können Sie Fehler reduzieren, die Fehlerbehebung erleichtern und den Ablauf bis zur Produktion verbessern?](#)
- [OPS 6. Wie können Sie Bereitstellungsrisiken eindämmen?](#)
- [OPS 7. Wie bringen Sie in Erfahrung, ob Sie für die Unterstützung eines Workloads bereit sind?](#)

OPS 4. Wie implementieren Sie die Beobachtbarkeit in Ihrer Workload?

Implementieren Sie Beobachtbarkeit in Ihre Workload, damit Sie deren Zustand verstehen und datengesteuerte Entscheidungen auf der Grundlage von Geschäftsanforderungen treffen können.

Best Practices

- [OPS04-BP01 Identifizieren Sie die wichtigsten Leistungsindikatoren](#)
- [OPS04-BP02 Implementieren Sie Anwendungstelemetrie](#)
- [OPS04-BP03 Implementieren Sie Benutzererlebnis-Telemetrie](#)
- [OPS04-BP04 Implementieren einer Abhängigkeitstelemetrie](#)
- [OPS04-BP05 Implementieren Sie verteiltes Tracing](#)

OPS04-BP01 Identifizieren Sie die wichtigsten Leistungsindikatoren

Die Implementierung von Beobachtbarkeit in Ihrer Workload beginnt damit, ihren Status zu verstehen und datengestützte Entscheidungen auf der Grundlage der geschäftlichen Anforderungen zu treffen. Eine der wirksamsten Methoden, um sicherzustellen, dass die Überwachungstätigkeiten und die Unternehmensziele aufeinander abgestimmt werden, ist die Definition und Überwachung zentraler Leistungsindikatoren (KPIs).

Gewünschtes Ergebnis: Effiziente Beobachtbarkeitspraktiken, die eng an den Geschäftszielen ausgerichtet sind und sicherstellen, dass die Überwachungsanstrengungen stets greifbaren Geschäftsergebnissen dienen.

Typische Anti-Muster:

- **UndefiniertKPIs:** Die Arbeit ohne klare Angaben KPIs kann dazu führen, dass zu viel oder zu wenig überwacht wird und wichtige Signale fehlen.
- **StatischKPIs:** Es wird nicht wiederholt oder verfeinert KPIs, wenn sich die Arbeitslast oder die Geschäftsziele ändern.
- **Fehlausrichtung:** Konzentration auf technische Metriken, die nicht direkt mit Geschäftsergebnissen korrelieren oder schwieriger mit realen Problemen zu korrelieren sind.

Vorteile der Nutzung dieser bewährten Methode:

- Einfache Identifizierung von Problemen: In Unternehmen KPIs treten Probleme häufig deutlicher zutage als bei technischen Kennzahlen. Ein Einbruch in einem Unternehmen KPI kann ein Problem effektiver lokalisieren als die Analyse zahlreicher technischer Kennzahlen.
- Geschäftsausrichtung: Es wird sichergestellt, dass die Überwachungsaktivitäten die Geschäftsziele direkt unterstützen.
- Effizienz: Es erfolgt eine Priorisierung der Ressourcen für die Überwachung und die Konzentration auf wichtige Metriken.
- Proaktivität: Probleme werden erkannt und gelöst, bevor sie weitreichende Auswirkungen auf das Geschäft haben.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Um die Arbeitslast effektiv zu definieren: KPIs

1. Beginnen Sie mit den Geschäftsergebnissen: Bevor Sie sich mit Metriken befassen, sollten Sie sich mit den gewünschten Geschäftsergebnissen vertraut machen. Sind es höhere Umsätze, mehr Benutzerinteraktionen oder schnellere Reaktionszeiten?
2. Stimmen Sie technische Metriken auf Geschäftsziele ab: Nicht alle technischen Metriken wirken sich direkt auf die Geschäftsergebnisse aus. Identifizieren Sie diejenigen, die dies tun, aber es ist oft einfacher, ein Problem mithilfe eines Unternehmens zu identifizieren KPI.
3. Verwenden Sie [Amazon CloudWatch](#): CloudWatch Employ, um Kennzahlen zu definieren und zu überwachen, die Ihren entsprechen KPIs.
4. Regelmäßige Überprüfung und Aktualisierung KPIs: Sorgen Sie dafür, dass Ihre Daten KPIs relevant sind, wenn sich Ihre Arbeitslast und Ihr Unternehmen weiterentwickeln.
5. Stakeholder einbeziehen: Beziehen Sie sowohl technische als auch geschäftliche Teams in die Definition und Überprüfung mit ein KPIs.

Aufwand für den Implementierungsplan: Mittel

Ressourcen

Zugehörige bewährte Methoden:

- [the section called “OPS04-BP02 Implementieren Sie die Anwendungstelemetrie”](#)

- [the section called “OPS04-BP03 Implementieren Sie Telemetrie für die Benutzererfahrung”](#)
- [the section called “OPS04-BP04 Implementieren einer Abhängigkeitstelemetrie”](#)
- [the section called “OPS04-BP05 Implementieren Sie die verteilte Ablaufverfolgung”](#)

Zugehörige Dokumente:

- [AWS Bewährte Verfahren zur Beobachtbarkeit](#)
- [CloudWatch Benutzerleitfaden](#)
- [AWS Kurs Observability Skill Builder](#)

Zugehörige Videos:

- [Entwicklung einer Beobachtbarkeitsstrategie](#)

Zugehörige Beispiele:

- [Workshop zur Beobachtbarkeit](#)

OPS04-BP02 Implementieren Sie Anwendungstelemetrie

Anwendungstelemetrie dient als Grundlage für die Beobachtbarkeit Ihres Workloads. Die ausgegebene Telemetrie muss unbedingt umsetzbare Erkenntnisse zum Status Ihrer Anwendung und zum Erreichen sowohl technischer als auch geschäftlicher Ergebnisse liefern. Von der Problembehebung über die Messung der Auswirkungen einer neuen Funktion bis hin zur Sicherstellung der Abstimmung mit den wichtigsten Unternehmensleistungsindikatoren (KPIs) — die Anwendungstelemetrie gibt Aufschluss darüber, wie Sie Ihre Workloads aufbauen, betreiben und weiterentwickeln.

Metriken, Protokolle und Traces bilden die drei wichtigsten Säulen der Beobachtbarkeit. Sie dienen als Diagnosetools, die den Status Ihrer Anwendung beschreiben. Im Laufe der Zeit helfen sie bei der Erstellung von Baselines und der Identifizierung von Anomalien. Um jedoch sicherzustellen, dass die Überwachungsaktivitäten und die Geschäftsziele aufeinander abgestimmt sind, ist es von entscheidender Bedeutung, diese zu definieren und zu überwachen. KPIs Unternehmen machen es KPIs oft einfacher, Probleme zu identifizieren als nur technische Kennzahlen.

Andere Telemetriearten, wie die Überwachung realer Benutzer (RUM) und synthetische Transaktionen, ergänzen diese primären Datenquellen. RUM bietet Einblicke in Benutzerinteraktionen

in Echtzeit, während synthetische Transaktionen potenzielles Benutzerverhalten simulieren und so helfen, Engpässe zu erkennen, bevor echte Benutzer darauf stoßen.

Gewünschtes Ergebnis: Sie erzielen umsetzbare Erkenntnisse zur Leistung Ihres Workloads. Diese Erkenntnisse ermöglichen es Ihnen, proaktive Entscheidungen zur Leistungsoptimierung zu treffen, eine höhere Workload-Stabilität zu erreichen, CI/CD-Prozesse zu rationalisieren und Ressourcen effektiv zu nutzen.

Typische Anti-Muster:

- Unvollständige Beobachtbarkeit: Wenn die Beobachtbarkeit nicht auf jeder Ebene der Workload berücksichtigt wird, führt dies zu blinden Flecken, die wichtige Erkenntnisse über Systemleistung und Verhalten verschleiern können.
- Fragmentierte Datenansicht: Wenn Daten über mehrere Tools und Systeme verteilt sind, wird es schwierig, einen ganzheitlichen Überblick über den Zustand und die Leistung Ihrer Workloads zu behalten.
- Von Benutzern gemeldete Probleme: Ein Zeichen dafür, dass eine proaktive Problemerkennung durch Telemetrie und KPI Unternehmensüberwachung fehlt.

Vorteile der Nutzung dieser bewährten Methode:

- Fundierte Entscheidungsfindung: Mit Erkenntnissen aus Telemetrie und Unternehmen können Sie KPIs datengestützte Entscheidungen treffen.
- Verbesserte betriebliche Effizienz: Datengesteuerte Ressourcennutzung führt zu Kosteneffektivität.
- Verbesserte Workload-Stabilität: Schnellere Erkennung und Lösung von Problemen führt zu einer verbesserten Verfügbarkeit.
- Optimierte CI/CD-Prozesse: Erkenntnisse aus Telemetriedaten erleichtern die Verfeinerung von Prozessen und sichern die Codebereitstellung.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Verwenden Sie AWS Dienste wie [Amazon CloudWatch](#) und, um Anwendungstelemetrie für Ihren Workload zu implementieren. [AWS X-Ray](#) Amazon CloudWatch bietet eine umfassende Suite von Überwachungstools, mit denen Sie Ihre Ressourcen und Anwendungen in AWS und vor Ort überwachen können. Der Service erfasst, verfolgt und analysiert Metriken, konsolidiert und überwacht

Protokolldaten und reagiert auf Änderungen in Ihren Ressourcen, wodurch Sie besser verstehen, wie Ihre Workload funktioniert. Gleichzeitig AWS X-Ray können Sie Ihre Anwendungen verfolgen, analysieren und debuggen, sodass Sie ein tiefes Verständnis des Verhaltens Ihrer Workloads erhalten. Mit Funktionen wie Service Maps, Latenzverteilungen und Trace-Zeitplänen AWS X-Ray bietet es Einblicke in die Leistung Ihres Workloads und die Engpässe, die sich darauf auswirken.

Implementierungsschritte

1. Identifizieren, welche Daten erfasst werden sollen: Ermitteln Sie die wichtigsten Metriken, Protokolle und Traces, die aussagekräftige Erkenntnisse zu Zustand, Leistung und Verhalten Ihres Workloads bieten.
2. Stellen Sie den [CloudWatchAgenten bereit: Der CloudWatch Agent](#) spielt eine wichtige Rolle bei der Beschaffung von System- und Anwendungsmetriken und Protokollen von Ihrem Workload und der zugrunde liegenden Infrastruktur. Der CloudWatch Agent kann auch verwendet werden, um Spuren zu sammeln OpenTelemetry oder zu röntgen und sie an X-Ray zu senden.
3. Implementieren Sie die Anomalieerkennung für Protokolle und Metriken: Verwenden Sie die Erkennung von [CloudWatch Protokollanomalien und die Erkennung](#) von [CloudWatchMetrikanomalien](#), um ungewöhnliche Aktivitäten im Betrieb Ihrer Anwendung automatisch zu identifizieren. Diese Tools verwenden Machine-Learning-Algorithmen, um Anomalien zu erkennen und sie zu melden. Dadurch werden Ihre Überwachungsfunktionen verbessert und die Reaktionszeit bei potenziellen Störungen oder Sicherheitsbedrohungen verkürzt. Richten Sie diese Features ein, um den Zustand und die Sicherheit von Anwendungen proaktiv zu verwalten.
4. Schützen Sie sensible Protokolldaten: Verwenden Sie den [Datenschutz von Amazon CloudWatch Logs](#), um vertrauliche Informationen in Ihren Protokollen zu maskieren. Dieses Feature trägt zur Wahrung von Datenschutz und Compliance bei, indem sensible Daten automatisch erkannt und maskiert werden, bevor auf sie zugegriffen wird. Implementieren Sie Datenmaskierung, um sensible Daten wie personenbezogene Daten sicher zu handhaben und zu schützen (PII).
5. Definieren und überwachen Sie Ihr GeschäftKPIs: Legen Sie [benutzerdefinierte Kennzahlen](#) fest, die auf Ihre [Geschäftsergebnisse](#) abgestimmt sind.
6. Instrumentieren Sie Ihre Anwendung mit AWS X-Ray: Neben der Bereitstellung des CloudWatch Agenten ist es wichtig, dass [Ihre Anwendung](#) so konfiguriert ist, dass sie Trace-Daten aussendet. Dieser Prozess kann weitere Erkenntnisse zum Verhalten und zur Leistung Ihrer Workload liefern.
7. Standardisierung der Datenerfassung in Ihrer gesamten Anwendung: Standardisieren Sie die Datenerfassungspraktiken für Ihre gesamte Anwendung. Einheitlichkeit hilft bei der Korrelation und Analyse von Daten und liefert einen umfassenden Überblick über das Verhalten Ihrer Anwendung.

8. Implementieren Sie kontenübergreifende Beobachtbarkeit: Verbessern Sie die Effizienz der Überwachung über mehrere Konten hinweg AWS-Konten mit der [CloudWatch kontenübergreifenden Observability von Amazon](#). Mit dieser Funktion können Sie Metriken, Protokolle und Alarmer von verschiedenen Konten in einer einzigen Ansicht konsolidieren, was die Verwaltung vereinfacht und die Reaktionszeiten bei identifizierten Problemen in der gesamten Unternehmensumgebung verbessert. AWS
9. Daten analysieren und entsprechend handeln: Sobald die Datenerfassung und Normalisierung abgeschlossen sind, können Sie [Amazon CloudWatch](#) für die Analyse von Kennzahlen und Protokollen sowie für die Trace-Analyse verwenden. [AWS X-Ray](#) Eine solche Analyse kann wichtige Erkenntnisse über den Zustand, die Leistung und das Verhalten Ihrer Workload liefern und so Ihren Entscheidungsprozess beeinflussen.

Aufwand für den Implementierungsplan: Hoch

Ressourcen

Zugehörige bewährte Methoden:

- [OPS04-BP01 Arbeitslast definieren KPIs](#)
- [OPS04-BP03 Implementieren Sie Telemetrie für Benutzeraktivitäten](#)
- [OPS04-BP04 Implementieren Sie Abhängigkeitstelemetrie](#)
- [OPS04-BP05 Implementieren Sie die Rückverfolgbarkeit von Transaktionen](#)

Zugehörige Dokumente:

- [Bewährte Methoden zur Beobachtbarkeit für AWS](#)
- [CloudWatch-Benutzerhandbuch](#)
- [AWS X-Ray Entwicklerhandbuch](#)
- [Instrumentieren verteilter Systeme für Einblicke in die Betriebsabläufe](#)
- [Skill Builder-Kurs zur Beobachtbarkeit in AWS](#)
- [Was ist neu bei Amazon CloudWatch](#)
- [Was ist neu bei AWS X-Ray](#)

Zugehörige Videos:

- [AWS re:Invent 2022 — Bewährte Methoden zur Beobachtbarkeit bei Amazon](#)
- [AWS re:Invent 2022 — Entwicklung einer Strategie zur Beobachtung](#)

Zugehörige Beispiele:

- [Workshop zur Beobachtbarkeit](#)
- [AWS Lösungsbibliothek: Anwendungsüberwachung mit Amazon CloudWatch](#)

OPS04-BP03 Implementieren Sie Benutzererlebnis-Telemetrie

Ein entscheidender Erfolgsfaktor besteht darin, tiefe Einblicke in die Erfahrung Ihrer Kunden und deren Interaktionen mit Ihrer Anwendung zu gewinnen. Echte Benutzerüberwachung (RUM) und synthetische Transaktionen dienen zu diesem Zweck als leistungsstarke Tools. RUM liefert Daten über echte Benutzerinteraktionen und ermöglicht so eine ungefilterte Perspektive der Nutzerzufriedenheit. Synthetische Transaktionen simulieren Benutzerinteraktionen und helfen so dabei, potenzielle Probleme zu erkennen, noch bevor sie sich auf echte Nutzer auswirken.

Gewünschtes Ergebnis: Eine ganzheitliche Ansicht des Kundenerlebnisses, die proaktive Erkennung von Problemen und die Optimierung der Benutzerinteraktionen, um nahtlos digitale Erfahrungen zu ermöglichen.

Typische Anti-Muster:

- Anwendungen ohne reale Benutzerüberwachung (RUM):
 - Verzögerte Problemerkennung: Ohne diese RUM Option werden Sie möglicherweise erst dann auf Leistungsengpässe oder -probleme aufmerksam, wenn sich Benutzer beschweren. Dieser reaktive Ansatz kann bei Ihren Kunden zu Unzufriedenheit führen.
 - Fehlende Einblicke in die Benutzererfahrung: Wenn Sie sie nicht verwenden RUM, verlieren Sie wichtige Daten, die zeigen, wie echte Benutzer mit Ihrer Anwendung interagieren, wodurch Ihre Möglichkeiten zur Optimierung der Benutzererfahrung eingeschränkt werden.
- Anwendungen ohne synthetische Transaktionen:
 - Fehlende Grenzfälle: Synthetische Transaktionen helfen Ihnen dabei, Pfade und Funktionen zu testen, die von den meisten Benutzern möglicherweise nicht häufig verwendet werden, aber für bestimmte Geschäftsfunktionen von entscheidender Bedeutung sind. Ohne sie könnten mögliche Fehler bei diesen Pfaden und Funktionen unbemerkt bleiben.

- Ausbleibende Überprüfung auf Probleme bei inaktiver Anwendung: Regelmäßige synthetische Tests können Situationen simulieren, in denen echte Benutzer nicht aktiv mit Ihrer Anwendung interagieren, wodurch sichergestellt wird, dass das System immer korrekt funktioniert.

Vorteile der Nutzung dieser bewährten Methode:

- Proaktive Problemerkennung: Identifizieren und beheben Sie potenzielle Probleme, bevor sie sich auf echte Benutzer auswirken.
- Optimierte Benutzererfahrung: Kontinuierliches Feedback von RUM hilft dabei, das allgemeine Benutzererlebnis zu verfeinern und zu verbessern.
- Erkenntnisse zur Geräte- und Browserleistung: Verstehen Sie, wie gut Ihre Anwendung auf verschiedenen Geräten und Browsern funktioniert, um weitere Optimierungen zu ermöglichen.
- Validierte Geschäftsabläufe: Regelmäßige synthetische Transaktionen stellen sicher, dass Kernfunktionen und kritische Pfade stets betriebsbereit und effizient bleiben.
- Verbesserte Anwendungsleistung: Nutzen Sie Erkenntnisse aus echten Benutzerdaten, um die Reaktionsfähigkeit und Zuverlässigkeit Ihrer Anwendungen zu verbessern.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

AWS [Bietet Dienste wie Amazon RUM und Amazon CloudWatch RUM Synthetics an, um synthetische Transaktionen für die Telemetrie von Benutzeraktivitäten zu nutzen. CloudWatch](#)

In Verbindung mit Daten zur Benutzeraktivität bieten Metriken, Protokolle und Traces einen umfassenden Überblick über den Betriebsstatus der Anwendung und die Benutzererfahrung zugleich.

Implementierungsschritte

1. Bereitstellen von Amazon CloudWatch RUM: Integrieren Sie Ihre Anwendung, CloudWatch RUM um echte Benutzerdaten zu sammeln, zu analysieren und zu präsentieren.
 - a. Verwenden Sie die [CloudWatch RUM JavaScript Bibliothek](#), um sie in Ihre Anwendung zu integrieren RUM.
 - b. Richten Sie Dashboards ein, um echte Benutzerdaten zu visualisieren und zu überwachen.
2. CloudWatch Synthetics konfigurieren: Erstellen Sie Canaries oder skriptbasierte Routinen, die Benutzerinteraktionen mit Ihrer Anwendung simulieren.
 - a. Definieren Sie kritische Anwendungsworkflows und -pfade.

- b. Entwerfen Sie Kanarienvögel mithilfe von [CloudWatch Synthetics-Skripten](#), um Benutzerinteraktionen für diese Pfade zu simulieren.
 - c. Planen und überwachen Sie Canaries so, dass sie in bestimmten Intervallen ausgeführt werden, und sorgen Sie so für einheitliche Leistungsprüfungen.
3. Daten analysieren und darauf reagieren: Nutzen Sie Daten aus RUM und synthetische Transaktionen, um Erkenntnisse zu gewinnen und Korrekturmaßnahmen zu ergreifen, wenn Anomalien entdeckt werden. Verwenden Sie CloudWatch Dashboards und Alarme, um auf dem Laufenden zu bleiben.

Aufwand für den Implementierungsplan: Mittel

Ressourcen

Zugehörige bewährte Methoden:

- [OPS04-BP01 Identifizieren Sie die wichtigsten Leistungsindikatoren](#)
- [OPS04-BP02 Implementieren Sie Anwendungstelemetrie](#)
- [OPS04-BP04 Implementieren einer Abhängigkeitstelemetrie](#)
- [OPS04-BP05 Implementieren Sie verteiltes Tracing](#)

Zugehörige Dokumente:

- [CloudWatch RUMAmazon-Leitfaden](#)
- [Leitfaden für Amazon CloudWatch Synthetics](#)

Zugehörige Videos:

- [Optimieren Sie Anwendungen mithilfe von Erkenntnissen für Endbenutzer mit Amazon CloudWatch RUM](#)
- [AWS auf Air ft. Echte Benutzerüberwachung für Amazon CloudWatch](#)

Zugehörige Beispiele:

- [Workshop zur Beobachtbarkeit](#)
- [Git-Repository für Amazon CloudWatch RUM Web Client](#)

- [Verwenden von Amazon CloudWatch Synthetics zur Messung der Seitenladezeit](#)

OPS04-BP04 Implementieren einer Abhängigkeitstelemetrie

Die Abhängigkeitstelemetrie ist für die Überwachung des Status und der Leistung der externen Services und Komponenten, auf die Ihre Workload angewiesen ist, unerlässlich. Sie liefert wertvolle Erkenntnisse zu Erreichbarkeit, Timeouts und anderen kritischen Ereignissen im Zusammenhang mit Abhängigkeiten wie DNS, Datenbanken oder APIs von Drittanbietern. Wenn Sie Ihre Anwendung so instrumentieren, dass sie Metriken, Protokolle und Traces zu diesen Abhängigkeiten ausgibt, gewinnen Sie ein besseres Verständnis von potenziellen Engpässen, Leistungsproblemen oder Ausfällen, die sich auf Ihren Workload auswirken könnten.

Gewünschtes Ergebnis: Sicherstellen, dass die Abhängigkeiten, auf die Ihre Workload angewiesen ist, erwartungsgemäß funktionieren, sodass Sie Probleme proaktiv angehen und eine optimale Workload-Leistung gewährleisten können.

Typische Anti-Muster:

- Nichtbeachtung externer Abhängigkeiten: sich nur auf interne Anwendungsmetriken konzentrieren und dabei Metriken im Zusammenhang mit externen Abhängigkeiten außer Acht lassen.
- Mangelnde proaktive Überwachung: warten, bis Probleme auftreten, statt den Status und die Leistung von Abhängigkeiten kontinuierlich zu überwachen.
- Isolierte Überwachung: Einsatz mehrerer, unterschiedlicher Überwachungstools, was zu fragmentierten und inkonsistenten Ansichten bezüglich des Überwachungsstatus führen kann.

Vorteile der Nutzung dieser bewährten Methode:

- Verbesserte Zuverlässigkeit der Workloads: sicherstellen, dass externe Abhängigkeiten kontinuierlich verfügbar sind und optimal funktionieren.
- Schnellere Problemerkennung und -lösung: proaktives Identifizieren und Beheben von Problemen mit Abhängigkeiten, bevor sie sich auf die Workload auswirken.
- Umfassender Überblick: Erhalt eines ganzheitlichen Überblicks über interne und externe Komponenten, die den Workload-Status beeinflussen.
- Verbesserte Skalierbarkeit der Workloads: Verständnis der Skalierbarkeitsgrenzen und Leistungsmerkmale externer Abhängigkeiten.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Implementieren Sie die Abhängigkeitstelemetrie, indem Sie zunächst die Services, Infrastrukturen und Prozesse identifizieren, von denen Ihre Workload abhängt. Quantifizieren Sie, wie gute Bedingungen aussehen, wenn diese Abhängigkeiten wie erwartet funktionieren, und bestimmen Sie dann, welche Daten zum Messen dieser Bedingungen benötigt werden. Mit diesen Informationen können Sie Dashboards und Warnmeldungen erstellen, die Ihren Operations-Teams Erkenntnisse zum Status dieser Abhängigkeiten liefern. Verwenden Sie AWS-Tools, um die Auswirkungen zu ermitteln und zu quantifizieren, wenn Abhängigkeiten nicht die gewünschten Resultate zeigen. Überarbeiten Sie Ihre Strategie kontinuierlich, um Änderungen der Prioritäten, Ziele und gewonnenen Erkenntnisse Rechnung zu tragen.

Implementierungsschritte

So implementieren Sie die Abhängigkeitstelemetrie auf effiziente Weise:

1. Identifizierung externer Abhängigkeiten: Arbeiten Sie mit Stakeholdern zusammen, um die externen Abhängigkeiten zu ermitteln, von denen Ihr Workload abhängt. Zu externen Abhängigkeiten zählen Services wie externe Datenbanken, APIs von Drittanbietern, Netzwerkverbindungsrouuten zu anderen Umgebungen und DNS-Services. Der erste Schritt zu einer effektiven Abhängigkeitstelemetrie besteht darin, auf ganzer Ebene zu verstehen, welche diese Abhängigkeiten sind.
2. Erstellung einer Überwachungsstrategie: Sobald Sie sich ein klares Bild von Ihren externen Abhängigkeiten verschafft haben, entwerfen Sie eine darauf zugeschnittene Überwachungsstrategie. Dazu müssen Sie die Wichtigkeit jeder Abhängigkeit, ihr erwartetes Verhalten und alle damit verbundenen Service Level Agreements oder -Ziele verstehen. Richten Sie proaktive Warnmeldungen ein, die Sie über Statusänderungen oder Leistungsabweichungen informieren.
3. Verwendung der [Netzwerküberwachung](#): Verwenden Sie die Tools [Internet Monitor](#) und [Network Monitor](#), die umfassende Einblicke in die globalen Internet- und Netzwerkbedingungen bieten. Diese Tools helfen Ihnen dabei, Ausfälle, Unterbrechungen oder Leistungseinbußen, die sich auf Ihre externen Abhängigkeiten auswirken, zu verstehen und darauf zu reagieren.
4. Bleiben Sie mit [AWS Health](#) auf dem Laufenden: AWS Health ist die maßgebliche Informationsquelle für den Zustand Ihrer AWS Cloud-Ressourcen. Ermöglicht AWS Health die Visualisierung und den Erhalt von Benachrichtigungen über aktuelle Serviceereignisse und bevorstehende Änderungen, z. B. geplante Lebenszyklusereignisse. So können Sie Maßnahmen ergreifen, um die Auswirkungen zu minimieren.

- a. [Erstellen Sie angepasste AWS Health-Ereignisbenachrichtigungen](#) für E-Mail- und Chat-Kanäle über [AWS-Benutzerbenachrichtigungen](#) und integrieren Sie diese programmgesteuert mit [Ihren Überwachungs- und Warnungstools über Amazon EventBridge](#) oder die [AWS Health-API](#).
 - b. Planen und verfolgen Sie den Fortschritt bei Gesundheitsereignissen, die Maßnahmen erfordern, indem Sie diese in Änderungsmanagement- oder ITSM-Tools (wie [Jira](#) oder [ServiceNow](#)) integrieren, die Sie möglicherweise bereits über Amazon EventBridge oder die AWS Health-API verwenden.
 - c. Wenn Sie AWS Organizations verwenden, aktivieren Sie die [Organisationsansicht für AWS Health](#), um AWS Health-Ereignisse über Konten hinweg zu aggregieren.
5. Instrumentierung Ihrer Anwendung mit [AWS X-Ray](#): AWS X-Ray stellt Ihnen Erkenntnisse zur Leistung von Anwendungen und zu den zugrunde liegenden Abhängigkeiten bereit. Verfolgen Sie Anfragen von Anfang bis Ende nach, um Engpässe oder Ausfälle bei den externen Services oder Komponenten zu identifizieren, auf die sich Ihre Anwendung stützt.
 6. Verwendung von [Amazon DevOps Guru](#): Dieser Machine Learning-gestützte Service identifiziert operative Probleme, prognostiziert das Auftreten kritischer Probleme und empfiehlt spezifische Maßnahmen. Dadurch ist er von unschätzbarem Wert, wenn es darum geht, Erkenntnisse zu Abhängigkeiten zu gewinnen und festzustellen, dass sie nicht die Ursache von operativen Problemen sind.
 7. Regelmäßige Überwachung: Überwachen Sie kontinuierlich alle Metriken und Protokolle, die sich auf externe Abhängigkeiten beziehen. Richten Sie Warnmeldungen ein, die Sie über unerwartetes Verhalten oder Leistungseinbußen informieren.
 8. Validierung nach Änderungen: Überprüfen Sie nach jeder Aktualisierung oder Änderung einer externen Abhängigkeit deren Leistung und Ausrichtung auf die Anforderungen Ihrer Anwendung.

Aufwand für den Implementierungsplan: Mittel

Ressourcen

Zugehörige bewährte Methoden:

- [OPS04-BP01 Definieren von Workload-KPIs](#)
- [OPS04-BP02 Implementieren einer Anwendungstelemetrie](#)
- [OPS04-BP03 Implementieren von Telemetrie für Benutzeraktivitäten](#)
- [OPS04-BP05 Implementierung einer Transaktionsverfolgung](#)
- [OP08-BP04 Erstellen umsetzbarer Warnmeldungen](#)

Zugehörige Dokumente:

- [Amazon Personal Health Dashboard-Benutzerhandbuch](#)
- [AWS Internet Monitor-Benutzerhandbuch](#)
- [AWS X-Ray-Entwicklerhandbuch](#)
- [AWS DevOps Guru-Benutzerhandbuch](#)

Zugehörige Videos:

- [Wie sich Internetprobleme auf die Leistung von Apps auswirken](#)
- [Einführung in Amazon DevOps Guru](#)
- [Ressourcenlebenszyklus-Ereignisse mit AWS Health in großem Umfang verwalten](#)

Zugehörige Beispiele:

- [AWS Health Aware](#)
- [Tag-basierte Filterung verwenden, um AWS Health-Überwachung und -Warnmeldungen in großem Umfang zu verwalten](#)

OPS04-BP05 Implementieren Sie verteiltes Tracing

Die verteilte Nachverfolgung bietet eine Möglichkeit, Anfragen zu überwachen und zu visualisieren, während sie verschiedene Komponenten eines verteilten Systems durchlaufen. Durch die Erfassung von Trace-Daten aus mehreren Quellen und deren Analyse in einer zentralen Ansicht können Teams besser verstehen, wie Anfragen ablaufen, wo Engpässe bestehen und worauf Optimierungsbemühungen abzielen sollten.

Gewünschtes Ergebnis: Sie verschaffen sich einen ganzheitlichen Überblick über die Anfragen, die durch Ihr verteiltes System fließen, und ermöglichen so präzises Debugging, optimierte Leistung und verbesserte Benutzererfahrungen.

Typische Anti-Muster:

- Inkonsistente Instrumentierung: Nicht alle Services in einem verteilten System sind für die Nachverfolgung instrumentiert.
- Latenz wird ignoriert: Sie konzentrieren sich nur auf Fehler und berücksichtigen nicht die Latenz oder allmähliche Leistungseinbußen.

Vorteile der Nutzung dieser bewährten Methode:

- **Umfassender Systemüberblick:** Visualisierung des gesamten Anfragenverlaufs, vom Eingang bis zum Ausgang.
- **Verbessertes Debugging:** Schnelle Identifizierung von Fehlern oder Leistungsproblemen.
- **Verbessertes Benutzererlebnis:** Überwachung und Optimierung auf der Grundlage von tatsächlichen Benutzerdaten, um sicherzustellen, dass das System den realen Anforderungen entspricht.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Identifizieren Sie zunächst alle Elemente Ihrer Workload, für die eine Instrumentierung erforderlich ist. Sobald alle Komponenten berücksichtigt sind, können Sie Tools wie AWS X-Ray und OpenTelemetry zum Sammeln von Trace-Daten für die Analyse mit Tools wie X-Ray und Amazon CloudWatch ServiceLens Map nutzen. Nehmen Sie regelmäßig an Besprechungen mit Entwicklern teil und ergänzen Sie diese Diskussionen mit Tools wie Amazon DevOps Guru, X-Ray Analytics und X-Ray Insights, um tiefere Erkenntnisse zu gewinnen. Richten Sie Warnmeldungen anhand von Trace-Daten ein, damit Sie benachrichtigt werden, wenn die im Workload-Überwachungsplan definierten Ergebnisse gefährdet sind.

Implementierungsschritte

So implementieren Sie die verteilte Nachverfolgung auf effektive Weise:

1. Verwendung von [AWS X-Ray](#): Integrieren Sie X-Ray in Ihre Anwendung, um Erkenntnisse zu ihrem Verhalten zu gewinnen, ihre Leistung zu verstehen und Engpässe zu lokalisieren. Nutzen Sie X-Ray Insights für die automatische Trace-Analyse.
2. Instrumentieren Sie Ihre Dienste: Stellen Sie sicher, dass jeder Service, von einer [AWS Lambda](#)-Funktion bis zu einer [EC2-Instance](#), Trace-Daten sendet. Je mehr Dienste Sie instrumentieren, desto klarer ist die end-to-end Sicht.
3. Integrieren Sie [CloudWatch Real User Monitoring](#) und [synthetisches Monitoring](#): Integrieren Sie Real User Monitoring (RUM) und synthetisches Monitoring mit X-Ray. Auf diese Weise können reale Benutzererfahrungen erfasst und Benutzerinteraktionen simuliert werden, um potenzielle Probleme zu identifizieren.
4. Verwenden Sie den [CloudWatch Agenten](#): Der Agent kann Spuren entweder von X-Ray oder senden OpenTelemetry, wodurch die Tiefe der gewonnenen Erkenntnisse erweitert wird.

5. Verwenden Sie [Amazon DevOps Guru](#): DevOps Guru verwendet Daten von X-Ray, CloudWatch, und AWS Config, AWS CloudTrail um umsetzbare Empfehlungen zu geben.
6. Analyse von Traces: Überprüfen Sie die Trace-Daten regelmäßig, um Muster, Anomalien oder Engpässe zu erkennen, die sich auf die Leistung Ihrer Anwendung auswirken könnten.
7. Warnmeldungen einrichten: Konfigurieren Sie Alarme [CloudWatch](#) für ungewöhnliche Muster oder längere Latenzen, sodass Probleme proaktiv behoben werden können.
8. Kontinuierliche Verbesserung: Überarbeiten Sie Ihre Tracing-Strategie, wenn Services hinzugefügt oder geändert werden, um alle relevanten Datenpunkte zu erfassen.

Aufwand für den Implementierungsplan: Mittel

Ressourcen

Zugehörige bewährte Methoden:

- [OPS04-BP01 Identifizieren Sie die wichtigsten Leistungsindikatoren](#)
- [OPS04-BP02 Implementieren Sie Anwendungstelemetrie](#)
- [OPS04-BP03 Implementieren Sie Benutzererlebnis-Telemetrie](#)
- [OPS04-BP04 Implementieren einer Abhängigkeitstelemetrie](#)

Zugehörige Dokumente:

- [AWS X-Ray Leitfaden für Entwickler](#)
- [Benutzerhandbuch für CloudWatch Amazon-Agenten](#)
- [Amazon DevOps Guru-Benutzerhandbuch](#)

Zugehörige Videos:

- [Nutzen Sie AWS X-Ray Insights](#)
- [AWS auf Air ft. Beobachtbarkeit: Amazon CloudWatch](#) und AWS X-Ray

Zugehörige Beispiele:

- [Instrumentierung Ihrer Anwendung für AWS X-Ray](#)

OPS 5. Wie können Sie Fehler reduzieren, die Fehlerbehebung erleichtern und den Ablauf bis zur Produktion verbessern?

Verwenden Sie Strategien, die die Übertragung von Änderungen auf die Produktionsumgebung verbessern und Faktorwechsel, schnelles Feedback zur Qualität sowie eine schnelle Fehlerbehebung ermöglichen. Dadurch fließen nützliche Änderungen schneller in die Produktion ein und es treten bei der Bereitstellung weniger Probleme auf. Zudem können Probleme, die durch Bereitstellungsaktivitäten verursacht werden, schnell aufgespürt und gelöst werden.

Best Practices

- [OPS05-BP01 Verwendung einer Versionskontrolle](#)
- [OPS05-BP02 Testen und Validieren von Änderungen](#)
- [OPS05-BP03 Einsatz von Systemen zur Konfigurationsverwaltung](#)
- [OPS05-BP04 Einsatz von Systemen zur Build- und Bereitstellungsverwaltung](#)
- [OPS05-BP05 Durchführen der Patch-Verwaltung](#)
- [OPS05-BP06 Designstandards teilen](#)
- [OPS05-BP07 Implementieren von Verfahren zur Verbesserung der Codequalität](#)
- [OPS05-BP08 Verwenden mehrerer Umgebungen](#)
- [OPS05-BP09 Nehmen Sie häufige, kleine, reversible Änderungen vor](#)
- [OPS05-BP10 Vollständige Automatisierung von Integration und Bereitstellung](#)

OPS05-BP01 Verwendung einer Versionskontrolle

Aktivieren Sie die Verfolgung von Änderungen und Releases mithilfe einer Versionskontrolle.

Viele AWS-Services bieten Versionskontrollfunktionen. Verwenden Sie ein Revisions- oder [Quellkontroll](#)-System wie [Git](#), um Code und andere Artefakte (z. B. versionsgesteuerte [AWS CloudFormation](#)-Vorlagen Ihrer Infrastruktur) zu verwalten.

Gewünschtes Ergebnis: Ihre Teams arbeiten gemeinsam am Code. Bei der Zusammenführung ist der Code einheitlich und es gehen keine Änderungen verloren. Fehler können durch korrekte Versionsverwaltung leicht behoben werden.

Typische Anti-Muster:

- Sie haben Ihren Code auf Ihrer Workstation entwickelt und gespeichert. Es ist ein Speicherfehler bei der Workstation aufgetreten, der nicht rückgängig gemacht werden kann, und Sie haben den Code verloren.
- Nachdem Sie den vorhandenen Code mit Ihren Änderungen überschrieben haben, starten Sie Ihre Anwendung neu, doch sie funktioniert nicht mehr. Sie können die Änderung nicht rückgängig machen.
- Sie arbeiten an einer Berichtsdatei, deshalb ist sie für alle anderen schreibgeschützt, doch ein anderer Benutzer möchte sie bearbeiten. Der Benutzer kontaktiert Sie und bittet darum, die Arbeit daran zu beenden, damit er seine Aufgabe erledigen kann.
- Ihr Forschungsteam arbeitet an einer detaillierten Analyse, die Ihre zukünftige Arbeit prägt. Jemand hat versehentlich den endgültigen Bericht mit seiner Einkaufsliste überschrieben. Sie können die Änderung nicht rückgängig machen und müssen den Bericht neu erstellen.

Vorteile der Nutzung dieser bewährten Methode: Durch die Verwendung von Versionskontrollfunktionen können Sie problemlos einen bekanntermaßen funktionierenden Status bzw. frühere Versionen wiederherstellen und so das Risiko von verlorenen Assets begrenzen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Bewahren Sie Ressourcen in Repositories mit Versionskontrolle auf. Dies ermöglicht die Nachvollziehung von Änderungen, die Bereitstellung neuer Versionen, die Erkennung von Änderungen an bestehenden Versionen und die Rückkehr zu vorherigen Versionen (zum Beispiel bei einem Fehler die Zurücksetzung auf einen bekanntermaßen funktionierenden Zustand). Integrieren Sie die Versionskontrollfunktionen Ihrer Konfigurationsverwaltungssysteme in Ihre Verfahren.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS05-BP04 Einsatz von Systemen zur Build- und Bereitstellungsverwaltung](#)

Zugehörige Videos:

- [AWS re:Invent 2023 – How Lockheed Martin builds software faster, powered by DevSecOps](#)
- [AWS re:Invent 2023 – How GitHub operationalizes AI for team collaboration and productivity](#)

OPS05-BP02 Testen und Validieren von Änderungen

Jede bereitgestellte Änderung muss getestet werden, um Fehler in der Produktion zu vermeiden. Diese bewährte Methode konzentriert sich auf das Testen von Änderungen von der Versionskontrolle bis zur Erstellung von Artefakten. Neben Änderungen am Anwendungscode sollten die Tests auch die Infrastruktur, die Konfiguration, die Sicherheitskontrollen und die Betriebsverfahren umfassen. Es gibt viele Formen des Testens, von Tests der Einheiten bis hin zur Softwarekomponentenanalyse (SCA). Wenn Tests im Softwareintegrations- und -bereitstellungsprozess weiter nach links verschoben werden, führt dies zu einer höheren Gewissheit der Artefaktqualität.

Ihr Unternehmen muss Teststandards für alle Software-Artefakte entwickeln. Automatisierte Tests verringern den Arbeitsaufwand und vermeiden manuelle Testfehler. In einigen Fällen können aber auch manuelle Tests notwendig sein. Entwickler müssen Zugang zu automatisierten Testergebnissen haben, um Feedback-Schleifen zur Verbesserung der Softwarequalität zu schaffen.

Gewünschtes Ergebnis: Ihre Softwareänderungen werden vor der Bereitstellung getestet. Die Entwickler haben Zugang zu den Testergebnissen und den Validierungen. Ihre Organisation hat einen Teststandard, der für alle Softwareänderungen gilt.

Typische Anti-Muster:

- Sie stellen eine neue Softwareänderung ohne jegliche Tests bereit. Sie kann in der Produktion nicht ausgeführt werden, was zu einem Ausfall führt.
- Es werden neue Sicherheitsgruppen mit AWS CloudFormation eingesetzt, ohne in einer Vorproduktionsumgebung getestet zu werden. Durch die Sicherheitsgruppen ist Ihre App für Ihre Kunden unerreichbar.
- Eine Methode wurde geändert, aber es gibt keine Tests der Einheiten. Die Software läuft nicht, wenn sie in der Produktion eingesetzt wird.

Vorteile der Nutzung dieser bewährten Methode: Die Fehlerquote bei Änderungen bei Softwarebereitstellungen wird reduziert. Die Qualität der Software wird verbessert. Die Entwickler haben ein größeres Bewusstsein für die Lebensfähigkeit ihres Codes. Sicherheitsrichtlinien können zuverlässig eingeführt werden, um die Compliance des Unternehmens zu unterstützen. Infrastrukturänderungen, wie automatische Aktualisierungen der Skalierungsrichtlinien, werden im Voraus getestet, um den Anforderungen des Datenverkehrs gerecht zu werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Alle Änderungen, vom Anwendungscode bis zur Infrastruktur, werden im Rahmen Ihrer kontinuierlichen Integrationspraxis getestet. Die Testergebnisse werden veröffentlicht, damit die Entwickler schnelles Feedback erhalten. Ihre Organisation hat einen Teststandard, den alle Änderungen erfüllen müssen.

Nutzen Sie die Leistungsfähigkeit generativer KI mit Amazon Q Developer, um die Entwicklerproduktivität und die Codequalität zu verbessern. Amazon Q Developer umfasst die Generierung von Codevorschlägen (basierend auf großen Sprachmodellen), die Erstellung von Komponententests (einschließlich Randbedingungen) und Verbesserungen der Codesicherheit durch die Erkennung und Behebung von Sicherheitsschwachstellen.

Kundenbeispiel

Als Teil der kontinuierlichen Integrationspipeline führt AnyCompany Retail verschiedene Arten von Tests für alle Software-Artefakte durch. Das Unternehmen praktiziert eine testgesteuerte Entwicklung, sodass die gesamte Software über Tests von Einheiten verfügt. Sobald das Artefakt erstellt ist, werden End-to-End-Tests durchgeführt. Nach Abschluss dieser ersten Testrunde wird ein statischer Anwendungssicherheitsscan durchgeführt, bei dem nach bekannten Schwachstellen gesucht wird. Die Entwickler erhalten Meldungen, sobald die einzelnen Prüfpunkte durchlaufen wurden. Sobald alle Tests abgeschlossen wurden, wird der Software-Artefakt in einem Artefakt-Repository gespeichert.

Implementierungsschritte

1. Arbeiten Sie mit den Beteiligten in Ihrem Unternehmen zusammen, um einen Teststandard für Software-Artefakte zu entwickeln. Welche Standardtests sollten alle Artefakte bestehen? Gibt es Compliance- oder Governance-Anforderungen, die bei der Testabdeckung berücksichtigt werden müssen? Müssen Sie die Qualität des Codes testen? Wer muss informiert werden, sobald die Tests abgeschlossen sind?
 1. Die [Referenzarchitektur für AWS-Bereitstellungs-Pipelines](#) enthält eine maßgebliche Liste von Testtypen, die als Teil einer Integrationspipeline an Software-Artefakten durchgeführt werden können.
2. Instrumentieren Sie Ihre Anwendung mit den erforderlichen Tests auf der Grundlage Ihres Software-Teststandards. Jeder Testreihe sollte in weniger als zehn Minuten abgeschlossen sein. Tests sollten im Rahmen einer Integrationspipeline durchgeführt werden.
 - a. Verwenden Sie [Amazon Q Developer](#), ein generatives KI-Tool, mit dem Sie Modultestfälle (einschließlich Randbedingungen) erstellen, Funktionen mithilfe von Code und Kommentaren generieren und bekannte Algorithmen implementieren können.

- b. Verwenden Sie [Amazon CodeGuru Reviewer](#), um Ihren Anwendungscode mithilfe von Tests auf Fehler zu prüfen.
- c. Sie können [AWS CodeBuild](#) verwenden, um Tests auf Software-Artefakten durchzuführen.
- d. [AWS CodePipeline](#) kann Ihre Softwaretests in eine Pipeline orchestrieren.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS05-BP01 Verwendung einer Versionskontrolle](#)
- [OPS05-BP06 Gemeinsame Design-Standards](#)
- [OPS05-BP07 Implementieren von Verfahren zur Verbesserung der Codequalität](#)
- [OPS05-BP10 Vollständige Automatisierung von Integration und Bereitstellung](#)

Zugehörige Dokumente:

- [Einen testgetriebenen Entwicklungsansatz verwenden](#)
- [Beschleunigen Ihres Softwareentwicklungszyklus mit Amazon Q](#)
- [Amazon Q Developer \(jetzt allgemein verfügbar\) enthält Vorschauen neuer Funktionen, mit denen Sie das Entwicklererlebnis neu gestalten können](#)
- [Der ultimative Spickzettel für den Einsatz von Amazon Q Developer in Ihrer IDE](#)
- [Shift-Left-Workload, Nutzung von KI für die Testerstellung](#)
- [Amazon Q Developer Center](#)
- [10 Möglichkeiten, Anwendungen mit Amazon CodeWhisperer schneller zu entwickeln](#)
- [Ein Blick über die Codeabdeckung hinaus – mit Amazon CodeWhisperer](#)
- [Bewährte Methoden für Prompt-Engineering mit Amazon CodeWhisperer](#)
- [Automatisierte AWS CloudFormation-Testpipeline mit TaskCat und CodePipeline](#)
- [Erstellen einer End-to-End-AWS DevSecOps-CI/CD-Pipeline mit Open-Source-SCA-, -SAST- und -DAST-Tools](#)
- [Erste Schritte beim Testen von Serverless-Anwendungen](#)
- [Meine CI/CD-Pipeline ist mein Release Captain](#)
- [Durchführung von Continuous Integration und Continuous Delivery in AWS \(Whitepaper\)](#)

Zugehörige Videos:

- [Implementieren einer API mit Amazon Q Developer-Agent für die Softwareentwicklung](#)
- [Installation, Konfiguration und Verwendung von Amazon Q Developer mit JetBrains-IDEs \(Anleitung\)](#)
- [Amazon CodeWhisperer meistern – YouTube-Playlist](#)
- [AWS re:Invent 2020: Testbare Infrastruktur: Integrationstests in AWS](#)
- [AWS Summit ANZ 2021 – Vorantreiben einer „Test-First“-Strategie mit CDK und testgesteuerter Entwicklung](#)
- [Testen Ihrer Infrastruktur als Code mit AWS CDK](#)

Zugehörige Ressourcen:

- [Referenzarchitektur für AWS-Bereitstellungs-Pipelines – Anwendung](#)
- [AWS Kubernetes DevSecOps-Pipeline](#)
- [Tests von Einheiten für eine Node.js-Anwendung aus GitHub mithilfe von AWS CodeBuild ausführen](#)
- [Serverspec für die testgesteuerte Entwicklung von Infrastrukturcode verwenden](#)

Zugehörige Services:

- [Amazon Q Developer](#)
- [Amazon CodeGuru Reviewer](#)
- [AWS CodeBuild](#)
- [AWS CodePipeline](#)

OPS05-BP03 Einsatz von Systemen zur Konfigurationsverwaltung

Verwenden Sie Systeme zur Konfigurationsverwaltung, um Änderungen vorzunehmen und zu verfolgen. Diese Systeme reduzieren Fehler aufgrund von manuellen Prozessen und verringern den Testaufwand.

Bei der statischen Konfigurationsverwaltung werden Werte festgelegt, wenn eine Ressource initialisiert wird, die erwartungsgemäß während der Lebensdauer der Ressource konsistent bleibt. Bei der dynamischen Konfigurationsverwaltung werden bei der Initialisierung Werte festgelegt, die sich

während der Lebensdauer einer Ressource ändern können oder voraussichtlich ändern werden. So können Sie zum Beispiel durch eine Konfigurationsänderung ein Feature in Ihrem Code aktivieren oder während eines Vorfalles den Detaillierungsgrad des Protokolls ändern.

Konfigurationen sollten in einem bekannten und konsistenten Zustand bereitgestellt werden. Sie sollten die automatisierte Inspektion verwenden, um die Ressourcenkonfigurationen in mehreren Umgebungen und Regionen kontinuierlich zu überwachen. Diese Kontrollen sollten als automatisierter Code und automatisierte Verwaltung definiert werden, um sicherzustellen, dass Regeln in allen Umgebungen einheitlich angewendet werden. Änderungen an Konfigurationen sollten im Rahmen vereinbarter Verfahren zur Kontrolle von Änderungen aktualisiert und konsistent angewendet werden, sodass die Versionskontrolle gewahrt bleibt. Die Anwendungskonfiguration sollte unabhängig vom Anwendungs- und Infrastrukturcode verwaltet werden. Dies ermöglicht eine konsistente Bereitstellung in mehreren Umgebungen. Konfigurationsänderungen führen nicht dazu, dass die Anwendung neu erstellt oder bereitgestellt wird.

Gewünschtes Ergebnis: Sie konfigurieren, validieren und implementieren als Teil Ihrer CI/CD-Pipeline (Continuous Integration, Continuous Delivery). Sie überwachen, um zu überprüfen, ob die Konfigurationen korrekt sind. Dadurch werden die Auswirkungen auf Endbenutzer und Kunden minimiert.

Typische Anti-Muster:

- Sie aktualisieren die Konfigurationen aller Webserver manuell und eine Reihe von Servern reagiert aufgrund von Updatefehlern nicht mehr.
- Sie aktualisieren Ihre Anwendungsserver mehrere Stunden lang auf manuelle Weise. Die Inkonsistenz der Konfiguration während der Änderung führt zu unerwarteten Verhaltensweisen.
- Jemand hat Ihre Sicherheitsgruppen aktualisiert und auf Ihre Webserver kann nicht mehr zugegriffen werden. Sie wissen nicht, was geändert wurde, und verbringen viel Zeit mit der Suche nach dem Problem – die Zeit bis zur Wiederherstellung nimmt zu.
- Sie übertragen eine Vorproduktionskonfiguration ohne Validierung über CI/CD in die Produktion. Sie setzen Benutzer und Kunden falschen Daten und Services aus.

Vorteile der Nutzung dieser bewährten Methode: Die Einführung von Konfigurationsverwaltungssystemen reduziert den Aufwand für die Durchführung und Nachverfolgung von Änderungen sowie die Häufigkeit der durch manuelle Verfahren verursachten Fehler. Konfigurationsverwaltungssysteme liefern Garantien in Bezug auf Governance, Compliance und regulatorische Anforderungen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Konfigurationsverwaltungssysteme werden verwendet, um Änderungen an Anwendungs- und Umgebungskonfigurationen zu verfolgen und zu implementieren. Konfigurationsverwaltungssysteme werden auch eingesetzt, um Fehler zu reduzieren, die durch manuelle Prozesse verursacht werden, Konfigurationsänderungen wiederholbar und überprüfbar zu machen und den Aufwand zu reduzieren.

In AWS können Sie Ihre AWS-Ressourcenkonfigurationen mit [AWS Config über Konten und Regionen hinweg](#) kontinuierlich überwachen. So können Sie den Konfigurationsverlauf besser verfolgen, nachvollziehen, wie sich eine Konfigurationsänderung auf andere Ressourcen auswirkt, und sie im Hinblick auf die erwarteten oder gewünschten Konfigurationen mithilfe von [AWS-Config-Regeln](#) und [AWS Config-Konformitätspaketen](#) prüfen.

Für dynamische Konfigurationen in Ihren Anwendungen, die auf Amazon EC2-Instances, AWS Lambda, Containern, mobilen Anwendungen oder IoT-Geräten ausgeführt werden, können Sie [AWS AppConfig](#) verwenden, um sie in Ihren Umgebungen zu konfigurieren, zu validieren, bereitzustellen und zu überwachen.

Implementierungsschritte

1. Identifizieren Sie die Verantwortlichen der Konfiguration.
 - a. Informieren Sie die Verantwortlichen der Konfigurationen über alle Compliance-, Governance- oder regulatorischen Anforderungen.
2. Identifizieren Sie Konfigurationselemente und Leistungen.
 - a. Konfigurationselemente sind alle Anwendungs- und Umgebungskonfigurationen, die von einer Bereitstellung innerhalb Ihrer CI/CD-Pipeline betroffen sind.
 - b. Zu den Leistungen gehören Erfolgskriterien, Validierung und was überwacht werden muss.
3. Wählen Sie Tools für die Konfigurationsverwaltung basierend auf Ihren Geschäftsanforderungen und Ihrer Bereitstellungspipeline aus.
4. Ziehen Sie für signifikante Konfigurationsänderungen gewichtete Bereitstellungen wie Canary-Bereitstellungen in Betracht, um die Auswirkungen falscher Konfigurationen zu minimieren.
5. Integrieren Sie Ihre Konfigurationsverwaltung in Ihre CI/CD-Pipeline.
6. Bestätigen Sie alle übermittelten Änderungen.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS06-BP01 Plan für erfolglose Änderungen](#)
- [OPS06-BP02 Testbereitstellungen](#)
- [OPS06-BP03 Einsetzen sicherer Bereitstellungsstrategien](#)
- [OPS06-BP04 Automatisieren Sie Tests und Rollback](#)

Zugehörige Dokumente:

- [AWS Control Tower](#)
- [AWS Landing Zone Accelerator](#)
- [AWS Config](#)
- [Was ist AWS Config?](#)
- [AWS AppConfig](#)
- [Was ist AWS CloudFormation?](#)
- [AWS-Entwicklungstools](#)
- [AWS CodeBuild](#)
- [AWS CodePipeline](#)
- [AWS CodeDeploy](#)

Zugehörige Videos:

- [AWS re:Invent 2022 – Proaktive Governance und Compliance für AWS-Workloads](#)
- [AWS re:Invent 2020: Mit AWS Config Compliance als Code erzielen](#)
- [Verwaltung und Bereitstellung von Anwendungskonfigurationen mit AWS AppConfig](#)

OPS05-BP04 Einsatz von Systemen zur Build- und Bereitstellungsverwaltung

Verwenden Sie Systeme zur Build- und Bereitstellungsverwaltung. Diese Systeme reduzieren Fehler aufgrund von manuellen Prozessen und verringern den Testaufwand.

In AWS können Sie mithilfe von Services wie [AWS-Entwicklertools](#) (z. B. [AWS CodeBuild](#), [AWS CodePipeline](#) und [AWS CodeDeploy](#)) Continuous Integration- und Continuous Deployment-Pipelines (CI/CD) erstellen.

Gewünschtes Ergebnis: Ihre Systeme zur Build- und Bereitstellungsverwaltung unterstützen das Continuous Integration Continuous Delivery (CI/CD)-System Ihrer Organisation, das Funktionen zur Automatisierung sicherer Rollouts mit den richtigen Konfigurationen bietet.

Typische Anti-Muster:

- Nachdem Sie Ihren Code auf Ihrem Entwicklungssystem kompiliert haben, kopieren Sie die ausführbare Datei auf Ihre Produktionssysteme und sie kann nicht gestartet werden. Die lokalen Protokolldateien zeigen an, dass die Ausführung aufgrund fehlender Abhängigkeiten fehlgeschlagen ist.
- Sie erstellen Ihre Anwendung erfolgreich mit neuen Funktionen in Ihrer Entwicklungsumgebung und stellen den Code der Quality Assurance (QA, Qualitätsprüfung) zur Verfügung. Die QA-Prüfung schlägt fehl, da statische Komponenten fehlen.
- Am Freitag haben Sie Ihre Anwendung nach großem Aufwand manuell in Ihrer Entwicklungsumgebung erstellt, einschließlich der neu geschriebenen Funktionen. Am Montag können Sie die Schritte, mit denen Sie Ihre Anwendung erfolgreich erstellen konnten, nicht wiederholen.
- Sie führen die Tests durch, die Sie für den neuen Release erstellt haben. Sie verbringen die nächste Woche damit, eine Testumgebung einzurichten und alle vorhandenen Integrationstests durchzuführen, gefolgt von den Leistungstests. Der neue Code bewirkt eine inakzeptable Leistungsbeeinträchtigung und muss neu entwickelt und dann erneut getestet werden.

Vorteile der Nutzung dieser bewährten Methode: Mithilfe von Mechanismen zur Verwaltung von Erstellungs- und Bereitstellungsaktivitäten reduzieren Sie den Aufwand für wiederholte Aufgaben, verschaffen Ihren Teammitgliedern die Zeit, sich auf ihre wichtigen Aufgaben zu konzentrieren, und begrenzen die Entstehung von Fehlern durch manuelle Verfahren.

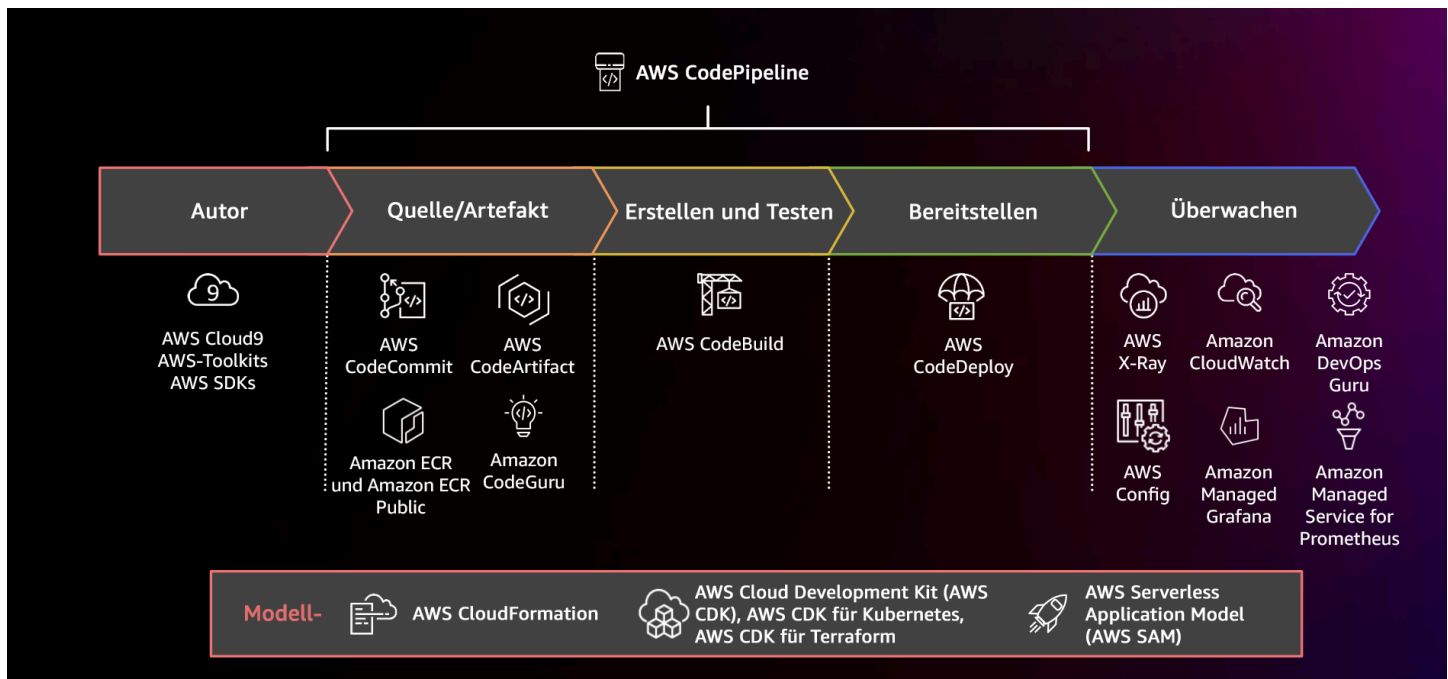
Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Systeme zur Build- und Bereitstellungsverwaltung werden verwendet, um Änderungen nachzuverfolgen und zu implementieren, Fehler zu reduzieren, die durch manuelle Prozesse verursacht werden, und den Aufwand für sichere Implementierungen zu minimieren. Nutzen Sie

eine vollständig automatisierte Integrations- und Bereitstellungs-Pipeline vom Einchecken des Codes über das Testen und die Bereitstellung bis hin zur Validierung. Dies reduziert die Vorlaufzeit, senkt die Kosten, ermöglicht häufigere Änderungen, minimiert den Aufwand und verbessert die Zusammenarbeit.

Implementierungsschritte



Diagramm, das eine CI/CD-Pipeline mit AWS CodePipeline und zugehörigen Services zeigt

1. Nutzen Sie ein Versionskontrollsystem zum Speichern und Verwalten von Ressourcen (z. B. Dokumente, Quellcode und Binärdateien).
2. Verwenden Sie CodeBuild, um den Quellcode zu kompilieren, Einheitentests auszuführen und Artefakte zu erzeugen, die direkt bereitgestellt werden können.
3. Verwenden Sie CodeDeploy als Bereitstellungsservice, der die Anwendungsbereitstellungen auf [Amazon EC2](#)-Instances, On-Premises-Instances, [Serverless AWS Lambda-Funktionen](#) oder [Amazon ECS](#) automatisiert.
4. Überwachen Sie Ihre Bereitstellungen.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS06-BP04 Automatisieren Sie Tests und Rollback](#)

Zugehörige Dokumente:

- [AWS-Entwicklungstools](#)
- [Was ist AWS CodeBuild?](#)
- [AWS CodeBuild](#)
- [Was ist AWS CodeDeploy?](#)

Zugehörige Videos:

- [AWS re:Invent 2022 – Bewährte AWS Well-Architected-Methoden für DevOps in AWS](#)

OPS05-BP05 Durchführen der Patch-Verwaltung

Führen Sie eine Patch-Verwaltung durch, um Funktionen zu erhalten, Probleme zu beheben und die Konformität mit der Governance zu gewährleisten. Automatisieren Sie die Patch-Verwaltung, um Fehler aufgrund manueller Prozesse zu reduzieren, zu skalieren und den Aufwand für die Installation von Patches zu verringern.

Patch- und Schwachstellenmanagement sind Teil Ihrer Vorteile- und Risikomanagement-Aktivitäten. Es ist vorzuziehen, unveränderliche Infrastrukturen zu haben und Workloads in verifizierten bekannten guten Zuständen bereitzustellen. Wenn dies nicht realisierbar ist, ist Patchen die verbleibende Option.

[AWS Health](#) ist die autoritative Informationsquelle für geplante Lebenszyklusereignisse und andere von Aktionen ausgelöste Ereignisse, die sich auf den Zustand Ihrer AWS Cloud-Ressourcen auswirken. Sie sollten über bevorstehende Änderungen und Aktualisierungen informiert sein, die durchgeführt werden sollten. Informationen zu wichtigen geplanten Lebenszyklusereignissen werden mindestens sechs Monate im Voraus bereitgestellt.

[Amazon EC2 Image Builder](#) stellt Pipelines zur Aktualisierung von Machine Images bereit. Als Teil des Patch-Managements sollten Sie [Amazon Machine Images](#) (AMIs) in Betracht ziehen, die eine [AMI-Image-Pipeline](#) oder Container-Images mit einer [Docker-Image-Pipeline](#) verwenden, während AWS Lambda Muster für [benutzerdefinierte Laufzeiten und zusätzliche Bibliotheken](#) bereitstellt, um Schwachstellen zu beseitigen.

Sie sollten Aktualisierungen für [Amazon Machine Images](#) für Linux- oder Windows Server-Images mit [Amazon EC2 Image Builder](#) verwalten. Sie können [Amazon Elastic Container Registry \(Amazon](#)

[ECR](#)) mit Ihrer bestehenden Pipeline verwenden, um Amazon ECS-Images und Amazon EKS-Images zu verwalten. Lambda enthält [Funktionen zur Versionsverwaltung](#).

Patches sollten nicht auf Produktionssystemen durchgeführt werden, ohne zuerst in einer sicheren Umgebung getestet zu werden. Patches sollten nur angewendet werden, wenn sie ein betriebliches oder geschäftliches Ergebnis unterstützen. In AWS können Sie [AWS Systems Manager Patch Manager](#) verwenden, um das Patchen verwalteter Systeme zu automatisieren und die Aktivität mithilfe von [Systems Manager-Wartungszeitfenstern](#) zu planen.

Gewünschtes Ergebnis: Ihre AMI und Container-Images sind gepatcht, aktuell und startbereit. Sie können den Status aller bereitgestellten Images nachverfolgen und wissen, dass die Patches konform sind. Sie können über den aktuellen Status berichten und verfügen über ein Verfahren, mit dem Sie Ihre Compliance-Anforderungen erfüllen können.

Typische Anti-Muster:

- Sie erhalten den Auftrag, alle neuen Sicherheits-Patches innerhalb von zwei Stunden anzuwenden, was zu mehreren Ausfällen aufgrund der Anwendungsinkompatibilität mit bestimmten Patches führt.
- Eine ungepatchte Bibliothek hat unbeabsichtigte Folgen, weil unbekannte Personen Schwachstellen darin ausnutzen, um auf Ihre Workload zuzugreifen.
- Sie patchen die Entwicklerumgebungen automatisch, ohne die Entwickler zu benachrichtigen. Sie erhalten mehrere Beschwerden von den Entwicklern, dass ihre Umgebung nicht mehr wie erwartet funktioniert.
- Sie haben die kommerziell im Handel erhältliche Software auf einer persistenten Instance nicht gepatcht. Als ein Problem mit der Software auftritt und Sie sich an den Anbieter wenden, werden Sie darüber informiert, dass die Version nicht unterstützt wird und Sie bestimmte Patches installieren müssen, um Unterstützung zu erhalten.
- Ein kürzlich veröffentlichter Patch für Ihre verwendete Verschlüsselungssoftware bietet signifikante Leistungsverbesserungen. Ihr ungepatchtes System weist Leistungsprobleme auf, die bestehen bleiben, weil es nicht gepatcht ist.
- Sie werden über eine Zero-Day-Schwachstelle informiert, die eine Notfalllösung erfordert, und Sie müssen alle Ihre Umgebungen manuell patchen.
- Sie sind über kritische Maßnahmen, die zur Wartung Ihrer Ressourcen erforderlich sind, nicht informiert, z. B. obligatorische Versionsupdates, weil Sie bevorstehende geplante Lebenszyklusevents und andere Informationen nicht überprüfen. Sie verlieren kritische Zeit für

Planung und Ausführung, was zu notfallmäßigen Änderungen für Ihre Teams und zu potenziellen Auswirkungen oder unerwarteten Ausfallzeiten führt.

Vorteile der Nutzung dieser bewährten Methode: Durch die Einrichtung eines Patch-Verwaltungsprozesses, einschließlich Ihrer Patching-Kriterien und Bereitstellungsmethodik für Ihre Umgebungen, können Sie die Patch-Ebenen skalieren und Berichte darüber erstellen. Das gibt Ihnen Sicherheit in Bezug auf Sicherheitspatches und gewährleistet einen klaren Überblick über den Status bekannter Problemlösungen. Dies wiederum fördert die Übernahme der gewünschten Merkmale und Funktionen, das Entfernen von Problemen und die kontinuierliche Compliance. Implementieren Sie Verwaltungssysteme und Automatisierung für Patches, um den Aufwand für die Bereitstellung von Patches zu reduzieren und Fehler zu begrenzen, die durch manuelle Prozesse verursacht werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Installieren Sie auf Ihren Systemen Patches zur Behebung von Problemen, zur Erlangung der gewünschten Funktionen oder Fähigkeiten sowie zur kontinuierlichen Einhaltung der Governance-Richtlinien und der Anforderungen des Lieferantensupport. Nehmen Sie in unveränderlichen Systemen eine Bereitstellung mit einer geeigneten Patch-Gruppe vor, um das gewünschte Ergebnis zu erzielen. Automatisieren Sie den Mechanismus der Patch-Verwaltung, um die Patch-Zeit zu verkürzen, Fehler aufgrund von manuellen Prozessen zu vermeiden und den Aufwand für die Installation von Patches zu verringern.

Implementierungsschritte

Für Amazon EC2 Image Builder:

1. Wenn Sie Amazon EC2 Image Builder verwenden, geben Sie die Pipeline-Details an:
 - a. Erstellen Sie eine Image-Pipeline und geben Sie ihr einen Namen.
 - b. Definieren Sie den Pipeline-Zeitplan und die Zeitzone.
 - c. Konfigurieren Sie alle Abhängigkeiten.
2. Wählen Sie ein Rezept:
 - a. Wählen Sie ein vorhandenes Rezept aus oder erstellen Sie ein neues.
 - b. Wählen Sie den Image-Typ aus.
 - c. Geben Sie Ihrem Rezept einen Namen und eine Versionsnummer.
 - d. Wählen Sie Ihr Basis-Image aus.

- e. Fügen Sie Build-Komponenten zur Zielregistrierung hinzu.
3. Optional: Definieren Sie Ihre Infrastrukturkonfiguration.
4. Optional: Definieren Sie die Konfigurationseinstellungen.
5. Prüfen Sie die Einstellungen.
6. Achten Sie regelmäßig auf die Rezepthygiene.

Für Systems Manager Patch Manager:

1. Erstellen Sie eine Patch-Baseline.
2. Wählen Sie eine Methode für Patching-Operationen aus.
3. Aktivieren Sie Compliance-Berichte und -Scans.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS06-BP04 Automatisieren Sie Tests und Rollback](#)

Zugehörige Dokumente:

- [Was ist Amazon EC2 Image Builder?](#)
- [Erstellen einer Image-Pipeline mit Amazon EC2 Image Builder](#)
- [Erstellen einer Container-Image-Pipeline](#)
- [AWS Systems Manager Patch Manager](#)
- [Arbeiten mit Patch Manager](#)
- [Arbeiten mit Patch-Compliance-Berichten](#)
- [AWS-Entwicklertools](#)

Zugehörige Videos:

- [CI/CD für Serverless-Anwendungen in AWS](#)
- [Design mit Blick auf die Ops](#)

Zugehörige Beispiele:

- [Anleitungen für AWS Systems Manager Patch Manager](#)

OPS05-BP06 Designstandards teilen

Tauschen Sie teamübergreifend bewährte Methoden aus, um das Bewusstsein zu schärfen und den Nutzen der Entwicklungsarbeit zu maximieren. Dokumentieren Sie sie und halten Sie sie auf dem neuesten Stand, wenn sich Ihre Architektur weiterentwickelt. Wenn gemeinsame Standards in Ihrem Unternehmen durchgesetzt werden, ist es wichtig, dass Mechanismen vorhanden sind, um Ergänzungen, Änderungen und Ausnahmen von Standards abzubilden. Ohne diese Option werden Standards zu einer Einschränkung der Innovation.

Gewünschtes Ergebnis: Designstandards werden von allen Teams in Ihren Organisationen gemeinsam genutzt. Sie werden dokumentiert und entsprechend der Weiterentwicklung der bewährten Verfahren aufbewahrt up-to-date.

Typische Anti-Muster:

- Zwei Entwicklerteams haben jeweils einen Service zur Authentifizierung von Benutzern erstellt. Ihre Benutzer müssen für jeden Teil des Systems, auf den sie zugreifen möchten, eigene Anmeldeinformationen verwenden.
- Jedes Team verwaltet seine eigene Infrastruktur. Eine neue Compliance-Anforderung erzwingt eine Änderung Ihrer Infrastruktur. Jedes Team implementiert sie auf andere Weise.

Vorteile der Nutzung dieser bewährten Methode: Die Verwendung gemeinsamer Standards unterstützt die Umsetzung bewährter Methoden und maximiert den Nutzen der Entwicklungsarbeit. Durch die Dokumentation und Aktualisierung von Designstandards wird Ihr Unternehmen stets up-to-date über bewährte Verfahren und Sicherheits- und Compliance-Anforderungen informiert.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Nutzen Sie bewährte Methoden, Designstandards, Checklisten, Arbeitsverfahren, Leitlinien und Governance-Anforderungen in allen Teams. Verwenden Sie Verfahren zur Anforderung von Änderungen, Ergänzungen und Ausnahmen von Designstandards, um Verbesserungen und Innovationen zu unterstützen. Stellen Sie sicher, dass die Teams über die veröffentlichten Inhalte informiert sind. Verfügen Sie über einen Mechanismus zur Beibehaltung von Designstandards up-to-date, wenn neue bewährte Verfahren auftauchen.

Kundenbeispiel

AnyCompany Der Einzelhandel verfügt über ein funktionsübergreifendes Architekturteam, das Softwarearchitekturmuster erstellt. Dieses Team entwickelt die Architektur mit integrierter Compliance und Governance. Teams, die diese gemeinsamen Standards anwenden, profitieren davon, dass Compliance und Governance bereits integriert sind. Sie können schnell auf dem Designstandard aufbauen. Das Architekturteam trifft sich vierteljährlich, um die Architekturmuster zu bewerten und sie gegebenenfalls zu aktualisieren.

Implementierungsschritte

1. Bestimmen Sie ein funktionsübergreifendes Team, das für die Entwicklung und Aktualisierung der Designstandards zuständig ist. Dieses Team sollte mit Stakeholdern in Ihrer gesamten Organisation zusammenarbeiten, um Designstandards, Arbeitsverfahren, Checklisten, Leitlinien und Governance-Anforderungen zu entwickeln. Dokumentieren Sie die Designstandards und geben Sie sie innerhalb Ihrer Organisation weiter.
 - a. Mit [AWS Service Catalog](#) können Sie Portfolios erstellen, die Designstandards als Infrastructure-as-Code abbilden. Sie können Portfolios über Konten hinweg gemeinsam nutzen.
2. Setzen Sie einen Mechanismus ein, um Designstandards beizubehalten, up-to-date sobald neue bewährte Verfahren identifiziert werden.
3. Wenn Designstandards zentral durchgesetzt werden, sollten Sie über ein Verfahren verfügen, um Änderungen, Aktualisierungen und Ausnahmen anzufordern.

Aufwand für den Implementierungsplan: Mittel. Die Entwicklung eines Prozesses zur Erstellung und gemeinsamen Nutzung von Designstandards kann die Koordination und Zusammenarbeit mit Stakeholdern in Ihrer gesamten Organisation erforderlich machen.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS01-BP03 Bewertung der Governance-Anforderungen](#) – Governance-Anforderungen beeinflussen Designstandards.
- [OPS01-BP04 Bewerten der Compliance-Anforderungen](#) – Compliance ist ein wichtiger Faktor bei der Erstellung von Designstandards.
- [OPS07-BP02 Sicherstellen einer konsistenten Prüfung der betrieblichen Bereitschaft](#) – Checklisten für die operative Einsatzbereitschaft sind ein Mechanismus zur Umsetzung von Designstandards bei der Gestaltung Ihrer Workload.

- [OPS11-BP01 Implementieren eines Prozesses für die kontinuierliche Verbesserung](#) – Die Aktualisierung von Designstandards ist ein Teil der kontinuierlichen Verbesserung.
- [OPS11-BP04 Wissensmanagement durchführen](#) – Als Teil Ihres Wissensmanagements sollten Sie Designstandards dokumentieren und weitergeben.

Zugehörige Dokumente:

- [Automatisieren Sie AWS Backup uns mit AWS Service Catalog](#)
- [AWS Service Catalog Ab Werk erweitertes Konto](#)
- [Wie die Expedia Group das Database-as-a-Service \(\) -Angebot \(\) DBaaS mithilfe von AWS Service Catalog](#)
- [Überblick über die Nutzung von Cloud-Architekturmustern](#)
- [Vereinfachen Sie die gemeinsame Nutzung Ihrer AWS Service Catalog Portfolios in einem Setup AWS Organizations](#)

Zugehörige Videos:

- [AWS Service Catalog — Erste Schritte](#)
- [AWS re:Invent 2020: Managen Sie Ihre AWS Service Catalog Portfolios wie ein Experte](#)

Zugehörige Beispiele:

- [AWS Service Catalog Referenzarchitektur](#)
- [AWS Service Catalog Werkstatt](#)

Zugehörige Services:

- [AWS Service Catalog](#)

OPS05-BP07 Implementieren von Verfahren zur Verbesserung der Codequalität

Implementieren Sie Verfahren zur Verbesserung der Codequalität und Minimierung von Fehlern. Einige Beispiele sind die testbasierte Entwicklung, Code-Reviews, die Einführung von Standards und Pair-Programming. Integrieren Sie diese Verfahren in Ihren Continuous-Integration- und Continuous-Delivery-Prozess.

Gewünschtes Ergebnis: Ihre Organisation setzt bewährte Methoden wie Code-Reviews oder Pair-Programming ein, um die Codequalität zu verbessern. Entwickler und operative Mitarbeiter nutzen bewährte Methoden zur Codequalität als Teil des Softwareentwicklungslebenszyklus.

Typische Anti-Muster:

- Sie führen ohne Code-Review Commits zum Main-Branch Ihrer Anwendung durch. Die Änderung wird automatisch in der Produktion bereitgestellt und verursacht einen Ausfall.
- Eine neue Anwendung wird ohne Unit-, End-to-End- oder Integrationstests entwickelt. Es gibt keine Möglichkeit, die Anwendung vor der Bereitstellung zu testen.
- Ihre Teams nehmen manuelle Änderungen in der Produktion vor, um Fehler zu beheben. Die Änderungen durchlaufen keine Tests oder Code-Reviews und werden nicht durch kontinuierliche Integrations- und Bereitstellungsprozesse erfasst oder protokolliert.

Vorteile der Nutzung dieser bewährten Methode: Durch die Umsetzung von Methoden zur Verbesserung der Codequalität können Sie die Anzahl der Probleme minimieren, die bei der Produktion noch vorhanden sind. Zu den bewährten Methoden für die Wahrung der Codequalität gehören Paarprogrammierung, Codeprüfung und Implementierung von KI-Produktivitätstools.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Implementieren Sie Verfahren zur Verbesserung der Codequalität, um vor der Bereitstellung Fehler zu minimieren. Nutzen Sie Verfahren wie die testbasierte Entwicklung, Code-Reviews und Pair-Programming, um die Qualität Ihrer Entwicklung zu verbessern.

Nutzen Sie die Leistungsfähigkeit generativer KI mit Amazon Q Developer, um die Entwicklerproduktivität und die Codequalität zu verbessern. Amazon Q Developer umfasst die Generierung von Codevorschlägen (basierend auf großen Sprachmodellen), die Erstellung von Komponententests (einschließlich Randbedingungen) und Verbesserungen der Codesicherheit durch die Erkennung und Behebung von Sicherheitsschwachstellen.

Kundenbeispiel

AnyCompany Retail wendet verschiedene Verfahren an, um die Codequalität zu verbessern. Die testbasierte Entwicklung ist der Standard für die Entwicklung von Anwendungen. Bei einigen neuen Funktionen arbeiten die Entwickler während eines Sprints zusammen. Jede Pull-Anforderung wird von einem erfahrenen Entwickler überprüft, bevor sie integriert und bereitgestellt wird.

Implementierungsschritte

1. Setzen Sie bei Ihrem kontinuierlichen Integrations- und Bereitstellungsprozess auf Code-Qualitätsverfahren wie die testbasierte Entwicklung, Code-Reviews und Pair-Programming. Nutzen Sie diese Techniken, um die Softwarequalität zu verbessern.
 - a. Verwenden Sie [Amazon Q Developer](#), ein generatives KI-Tool, mit dem Sie Modultestfälle (einschließlich Randbedingungen) erstellen, Funktionen mithilfe von Code und Kommentaren generieren, bekannte Algorithmen implementieren, Verstöße gegen Sicherheitsrichtlinien und Schwachstellen in Ihrem Code erkennen, Geheimnisse aufdecken, Infrastructure as Code (IaC) scannen, Code dokumentieren und Codebibliotheken von Drittanbietern schneller erlernen können.
 - b. [Amazon CodeGuru Reviewer](#) kann Machine-Learning-Programmierempfehlungen für Java- und Python-Code bereitstellen.

Aufwand für den Implementierungsplan: Mittel. Es gibt viele Möglichkeiten zur Umsetzung dieser bewährten Methode. Es kann jedoch schwierig sein, die Akzeptanz im Unternehmen zu erreichen.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS05-BP02 Testen und Validieren von Änderungen](#)
- [OPS05-BP06 Gemeinsame Design-Standards](#)

Zugehörige Dokumente:

- [Einen testgestützten Entwicklungsansatz verwenden](#)
- [Beschleunigen Ihres Softwareentwicklungszyklus mit Amazon Q](#)
- [Amazon Q Developer \(jetzt allgemein verfügbar\) enthält Vorschauen neuer Funktionen, mit denen Sie das Entwicklererlebnis neu gestalten können](#)
- [Der ultimative Spickzettel für den Einsatz von Amazon Q Developer in Ihrer IDE](#)
- [Shift-Left-Workload, Nutzung von KI für die Testerstellung](#)
- [Amazon Q Developer Center](#)
- [10 Möglichkeiten, Anwendungen mit Amazon CodeWhisperer schneller zu entwickeln](#)
- [Ein Blick über die Codeabdeckung hinaus – mit Amazon CodeWhisperer](#)

- [Bewährte Methoden für Prompt-Engineering mit Amazon CodeWhisperer](#)
- [Leitfaden für agile Software](#)
- [Meine CI/CD-Pipeline ist mein Release Captain](#)
- [Automatisieren von Code-Reviews mit Amazon CodeGuru Reviewer](#)
- [Einen testgestützten Entwicklungsansatz verwenden](#)
- [So entwickelt DevFactory mit Amazon CodeGuru bessere Anwendungen](#)
- [Über Pair-Programming](#)
- [RENGA Inc. automatisiert Code-Reviews mit Amazon CodeGuru](#)
- [Die Kunst der agilen Entwicklung: Testbasierte Entwicklung](#)
- [Warum Code-Reviews wichtig sind \(und tatsächlich Zeit sparen!\)](#)

Zugehörige Videos:

- [Implementieren einer API mit Amazon Q Developer-Agent für die Softwareentwicklung](#)
- [Installation, Konfiguration und Verwendung von Amazon Q Developer mit JetBrains-IDEs \(Anleitung\)](#)
- [Amazon CodeWhisperer meistern – YouTube-Playlist](#)
- [AWS re:Invent 2020: Kontinuierliche Verbesserung der Codequalität mit Amazon CodeGuru](#)
- [AWS Summit ANZ 2021 – Vorantreiben einer „Test-First“-Strategie mit CDK und testgesteuerter Entwicklung](#)

Zugehörige Services:

- [Amazon Q Developer](#)
- [Amazon CodeGuru Reviewer](#)
- [Amazon CodeGuru Profiler](#)

OPS05-BP08 Verwenden mehrerer Umgebungen

Verwenden Sie mehrere Umgebungen, um Ihre Workload auszuprobieren, zu entwickeln und zu testen. Verwenden Sie zunehmende Kontrollstufen, wenn Umgebungen sich der Produktion nähern, um sicherzustellen, dass Ihre Workload bei der Bereitstellung wie beabsichtigt funktioniert.

Gewünschtes Ergebnis: Sie verfügen über mehrere Umgebungen, die Ihre Compliance- und Governance-Anforderungen widerspiegeln. Auf Ihrem Weg zur Produktion testen und promoten Sie Code in Umgebungen.

1. Ihr Unternehmen tut dies durch die Einrichtung einer Landing Zone, die für Steuerung, Kontrollen, Kontoautomatisierung, Networking, Sicherheit und betriebliche Beobachtbarkeit sorgt. Verwalten Sie diese Landing Zone-Funktionen, indem Sie mehrere Umgebungen verwenden. Ein typisches Beispiel ist eine Sandbox-Organisation für die Entwicklung und Erprobung von Änderungen an einer [AWS Control Tower](#)-basierten Landing Zone, zu der auch [AWS IAM Identity Center](#) und Richtlinien wie [Service-Kontrollrichtlinien \(SCPs\)](#) gehören. Alle diese Elemente können den Zugang zur Landing Zone und den Betrieb von AWS-Konten innerhalb der Landing Zone erheblich beeinträchtigen.
2. Zusätzlich zu diesen Services erweitern Ihre Teams die Funktionen der Landing Zones mit Lösungen, die von AWS- und AWS-Partnern veröffentlicht werden, oder als maßgeschneiderte Lösungen in Ihrem Unternehmen entwickelt wurden. Zu den von AWS veröffentlichten Lösungen gehören beispielsweise [Customizations for AWS Control Tower \(CfCT\)](#) und [AWS Control TowerAccount Factory for Terraform \(AFT\)](#).
3. Ihr Unternehmen wendet dieselben Prinzipien beim Testen, Promoten von Code und Richtlinienänderungen für die Landing Zone durch Umgebungen auf Ihrem Weg zur Produktion an. Diese Strategie bietet eine stabile und sichere Landing Zone-Umgebung für Ihre Anwendungs- und Workload-Teams.

Typische Anti-Muster:

- Sie führen die Entwicklung in einer gemeinsamen Entwicklungsumgebung durch und ein weiterer Entwickler überschreibt Ihre Codeänderungen.
- Die restriktiven Sicherheitskontrollen Ihrer gemeinsamen Entwicklungsumgebung verhindern, dass Sie mit neuen Services und Funktionen experimentieren können.
- Sie führen Belastungstests auf Ihren Produktionssystemen durch und verursachen einen Ausfall für Ihre Benutzer.
- In der Produktion ist ein kritischer Fehler aufgetreten, der zum Verlust von Daten geführt hat. In Ihrer Produktionsumgebung versuchen Sie, die Bedingungen, die zum Datenverlust geführt haben, nachzustellen, damit Sie die Ursache feststellen und beseitigen können. Um einen weiteren Datenverlust während des Testens zu verhindern, müssen Sie die Anwendung für Ihre Benutzer deaktivieren.

- Sie betreiben einen Mehrmandanten-Service und können eine Kundenanfrage nach einer eigenen Umgebung nicht erfüllen.
- Möglicherweise testen Sie nicht immer, aber wenn Sie dies tun, testen Sie in Ihrer Produktionsumgebung.
- Sie glauben, dass die Einfachheit einer einzelnen Umgebung die Auswirkungen von Änderungen innerhalb der Umgebung ausgleicht.
- Sie erweitern eine wichtige Landing Zone-Funktion, aber die Änderung beeinträchtigt die Fähigkeit Ihres Teams, Konten entweder für neue Projekte oder für Ihre bestehenden Workloads zu bereitstellen.
- Sie wenden neue Kontrollen auf Ihre AWS-Konten an, aber die Änderung wirkt sich auf die Fähigkeit Ihres Workload-Teams aus, Änderungen innerhalb seiner AWS-Konten vorzunehmen.

Vorteile der Nutzung dieser bewährten Methode: Wenn Sie mehrere Umgebungen bereitstellen, können Sie gleichzeitig mehrere Entwicklungs-, Test- und Produktionsumgebungen unterstützen, ohne Konflikte zwischen Entwicklern oder Benutzer-Communitys zu verursachen. Bei komplexen Funktionen wie Landing Zones wird das Risiko von Änderungen erheblich reduziert, der Verbesserungsprozess vereinfacht und das Risiko kritischer Aktualisierungen der Umgebung verringert. Organisationen, die Landing Zones verwenden, profitieren in natürlicher Weise von mehreren Konten in ihrer AWS-Umgebung mit Kontostruktur, Governance, Netzwerk- und Sicherheitskonfigurationen. Im Laufe der Zeit kann sich die Landing Zone mit dem Wachstum Ihres Unternehmens weiterentwickeln, um Ihre Workloads und Ressourcen zu sichern und zu organisieren.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Verwenden Sie mehrere Umgebungen und stellen Sie den Entwicklern Sandbox-Umgebungen mit weniger Kontrollen zur Verfügung, in denen sie experimentieren können. Richten Sie individuelle Entwicklungsumgebungen ein, damit parallele Arbeit möglich ist. Dadurch steigern Sie die Agilität der Entwicklung. Implementieren Sie strengere Kontrollen erst in den Umgebungen, die kurz vor der Produktionsaufnahme stehen, damit Entwickler Innovationen schaffen können. Nutzen Sie die Infrastruktur als Code sowie Konfigurationsverwaltungssysteme, um Umgebungen bereitzustellen, die mit den in der Produktion vorhandenen Kontrollen einheitlich konfiguriert sind. Auf diese Weise können Sie sicherstellen, dass die Systeme bei der Bereitstellung wie erwartet funktionieren. Wenn Umgebungen nicht in Gebrauch sind, schalten Sie sie ab, um Kosten für ungenutzte Ressourcen zu vermeiden (z. B. Entwicklungssysteme am Abend und am Wochenende). Stellen Sie beim

Belastungstest produktionsgleiche Umgebungen bereit, um die Gültigkeit der Ergebnisse zu verbessern.

Teams aus den Bereichen Plattformentwicklung, Netzwerke und Sicherheit verwalten ihre Fähigkeiten häufig auf Organisationsebene mit unterschiedlichen Anforderungen. Eine Trennung der Konten allein reicht nicht aus, um separate Umgebungen für Experimente, Entwicklung und Tests bereitzustellen und aufrechtzuerhalten. Erstellen Sie in solchen Fällen separate Instanzen von AWS Organizations.

Ressourcen

Zugehörige Dokumente:

- [Instance Scheduler in AWS](#)
- [Was ist AWS CloudFormation?](#)
- [Organisation Ihrer AWS-Umgebung mit mehreren Konten – Mehrere Organisationen – Testen von Änderungen an Ihrer gesamten AWS-Umgebung](#)
- [AWS Control Tower-Handbuch](#)

OPS05-BP09 Nehmen Sie häufige, kleine, reversible Änderungen vor

Häufige, kleine und reversible Änderungen verringern den Umfang und die Auswirkung einer Änderung. In Verbindung mit Change-Management-Systemen, Systemen zur Konfigurationsverwaltung und Build- und Liefersystemen reduzieren häufige, kleine und reversible Änderungen den Umfang und die Auswirkungen einer Änderung. Dies macht die Fehlersuche effizienter und ermöglicht eine schnellere Korrektur, da die Möglichkeit besteht, Änderungen zurückzusetzen.

Typische Anti-Muster:

- Sie stellen vierteljährlich eine neue Version Ihrer Anwendung mit einem Änderungsfenster bereit, was bedeutet, dass ein zentraler Dienst ausgeschaltet wird.
- Sie nehmen häufig Änderungen an Ihrem Datenbankschema vor, ohne Änderungen in Ihren Managementsystemen nachzuverfolgen.
- Sie führen direkte manuelle Updates durch, überschreiben damit bestehende Installationen und Konfigurationen und haben keinen klaren Rollback-Plan.

Vorteile der Nutzung dieser bewährten Methode: Sie profitieren schneller von den Entwicklungsarbeiten, wenn Sie häufig kleine Änderungen bereitstellen. Wenn die Änderungen klein sind, ist es viel einfacher zu erkennen, ob sie unbeabsichtigte Folgen haben, und sie lassen sich leichter rückgängig machen. Wenn die Änderungen rückgängig gemacht werden können, ist die Implementierung mit geringeren Risiken verbunden, da die Wiederherstellung einfacher ist. Der Änderungsprozess hat ein geringeres Risiko und die Auswirkungen einer fehlgeschlagenen Änderung werden reduziert.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Niedrig

Implementierungsleitfaden

Machen Sie häufige, kleine und reversible Änderungen und verringern Sie dadurch den Umfang und die Auswirkung einer Änderung. Dies erleichtert die Fehlersuche, trägt zur Beschleunigung der Fehlerbehebung bei und bietet die Möglichkeit, eine Änderung zurückzusetzen. Außerdem profitiert Ihr Unternehmen schneller von neuen Entwicklungen.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS05-BP03 Einsatz von Systemen zur Konfigurationsverwaltung](#)
- [OPS05-BP04 Einsatz von Systemen zur Build- und Bereitstellungsverwaltung](#)
- [OPS06-BP04 Automatisieren Sie Tests und Rollback](#)

Zugehörige Dokumente:

- [Implementierung von Microservices auf AWS](#)
- [Microservices – Beobachtbarkeit](#)

OPS05-BP10 Vollständige Automatisierung von Integration und Bereitstellung

Automatisieren Sie den Aufbau, die Bereitstellung und die Tests der Workloads. Dadurch werden Fehler aufgrund von manuellen Prozessen und der Aufwand für die Bereitstellung von Änderungen verringert.

Wenden Sie Metadaten mithilfe von [Ressourcen-Tags](#) und [AWS -Ressourcengruppen](#) nach einer konsistenten [Tagging-Strategie](#) an, um die Identifizierung Ihrer Ressourcen zu erleichtern. Versehen

Sie Ihre Ressourcen mit Tags für Organisation, Kostenkalkulation, Zugriffssteuerung und Zielrichtung der Ausführung von automatisierten Betriebsaktivitäten.

Gewünschtes Ergebnis: Entwickler verwenden Tools, um Code bereitzustellen und bis zur Produktion zu unterstützen. Entwickler müssen sich nicht bei der AWS-Managementkonsole anmelden, um Updates bereitzustellen. Es gibt einen vollständigen Audit Trail für Änderungen und Konfigurationen, der die Governance- und Compliance-Anforderungen erfüllt. Prozesse sind wiederholbar und teamübergreifend standardisiert. Entwickler sind in der Lage, sich auf die Entwicklung und Code-Pushs zu konzentrieren, sodass die Produktivität steigt.

Typische Anti-Muster:

- Am Freitag schließen Sie die Erstellung des neuen Codes für Ihren Feature-Zweig ab. Am Montag, nach dem Ausführen Ihrer Skripts für die Codequalitätstests und einzelnen Komponententests, überprüfen Sie Ihren Code für den nächsten geplanten Release.
- Sie erhalten die Aufgabe, eine Korrektur für ein kritisches Problem zu schreiben, das sich auf eine große Anzahl von Kunden in der Produktion auswirkt. Nachdem Sie die Korrektur getestet haben, übermitteln Sie Ihren Code und fordern beim Änderungsmanagement die Bereitstellungsgenehmigung zur Produktion an.
- Als Entwickler melden Sie sich bei der AWS-Managementkonsole an, um eine neue Entwicklungsumgebung mit nicht standardmäßigen Methoden und Systemen zu erstellen.

Vorteile der Nutzung dieser bewährten Methode: Durch die Implementierung automatisierter Build- und Bereitstellungsverwaltungssysteme reduzieren Sie Fehler aus manuellen Prozessen und den Aufwand für die Bereitstellung von Änderungen, sodass sich Ihre Teammitglieder auf die Wertschöpfung konzentrieren können. Sie erhöhen die Liefergeschwindigkeit auf Ihrem Weg zur Produktion.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Niedrig

Implementierungsleitfaden

Verwenden Sie Systeme zur Build- und Bereitstellungsverwaltung für die Verfolgung und Implementierung von Änderungen, die Reduzierung von Fehlern, die durch manuelle Prozesse entstehen, sowie zur Verringerung des Aufwands. Nutzen Sie eine vollständig automatisierte Integrations- und Bereitstellungs-Pipeline vom Einchecken des Codes über das Testen und die Bereitstellung bis hin zur Validierung. Dies reduziert die Vorlaufzeit, fördert häufigere Änderungen,

reduziert den Aufwand, beschleunigt die Markteinführung, führt zu einer höheren Produktivität und erhöht die Sicherheit Ihres Codes bis hin zur Produktion.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS05-BP03 Einsatz von Systemen zur Konfigurationsverwaltung](#)
- [OPS05-BP04 Einsatz von Systemen zur Build- und Bereitstellungsverwaltung](#)

Zugehörige Dokumente:

- [Was ist AWS CodeBuild?](#)
- [Was ist AWS CodeDeploy?](#)

Zugehörige Videos:

- [AWS re:Invent 2022 – Bewährte AWS Well-Architected-Methoden für DevOps in AWS](#)

OPS 6. Wie können Sie Bereitstellungsrisiken eindämmen?

Verwenden Sie Ansätze, die schnelles Feedback zur Qualität liefern und eine schnelle Wiederherstellung bei Änderungen ermöglichen, die nicht zu den gewünschten Ergebnissen führen. Mit diesen Verfahren können Sie die Auswirkung von Problemen eindämmen, die durch Änderungen entstehen.

Best Practices

- [OPS06-BP01 Plan für erfolglose Änderungen](#)
- [OPS06-BP02 Testbereitstellungen](#)
- [OPS06-BP03 Einsetzen sicherer Bereitstellungsstrategien](#)
- [OPS06-BP04 Automatisieren Sie Tests und Rollback](#)

OPS06-BP01 Plan für erfolglose Änderungen

Planen Sie Maßnahmen für die Rückkehr zu einem bekanntermaßen funktionierenden Zustand oder die Korrektur in der Produktionsumgebung ein, falls bei der Bereitstellung ein nicht erwünschtes Ergebnis auftritt. Eine Richtlinie zur Festlegung eines solchen Plans hilft allen Teams, Strategien

zum Umgang mit fehlgeschlagenen Änderungen zu entwickeln. Einige Beispiele für Strategien sind Bereitstellungs- und Rollback-Schritte, Änderungsrichtlinien, Feature-Flags sowie die Isolierung und Verlagerung von Datenverkehr. Ein einzelner Release kann mehrere zusammengehörige Komponentenänderungen enthalten. Die Strategie sollte die Möglichkeit bieten, dem Ausfall einer Komponentenänderung standzuhalten oder sich danach zu regenerieren.

Gewünschtes Ergebnis: Sie haben einen detaillierten Wiederherstellungsplan für Ihre Änderung erstellt, falls diese nicht erfolgreich sein sollte. Darüber hinaus haben Sie die Größe Ihres Releases reduziert, um die potenziellen Auswirkungen auf andere Workload-Komponenten zu minimieren. Infolgedessen haben Sie die Auswirkungen auf Ihr Unternehmen verringert, indem Sie die potenziellen Ausfallzeiten aufgrund einer fehlgeschlagenen Änderung reduziert und die Flexibilität und Effizienz der Wiederherstellungszeiten erhöht haben.

Typische Anti-Muster:

- Sie haben Code bereitgestellt und Ihre Anwendung ist instabil geworden, aber es befinden sich aktive Benutzer im System. Sie müssen entscheiden, ob Sie die Änderung rückgängig machen und Auswirkungen auf die aktiven Benutzer in Kauf nehmen möchten, oder ob Sie die Änderung erst später rückgängig machen möchten, wodurch möglicherweise trotzdem Auswirkungen auf die Benutzer entstehen könnten.
- Nachdem Sie eine Routineänderung vorgenommen haben, kann auf Ihre neuen Umgebungen zugegriffen werden, aber eines Ihrer Subnetze ist nicht mehr erreichbar. Sie müssen entscheiden, ob Sie die gesamte Änderung rückgängig machen oder versuchen, die Nichtverfügbarkeit des Subnetzes zu beheben. Während Sie diese Entscheidung abwägen, bleibt das Subnetz nicht erreichbar.
- Ihre Systeme sind nicht so konzipiert, dass sie mit kleineren Releases aktualisiert werden können. Daher haben Sie Schwierigkeiten, die Bulk-Änderungen während einer fehlgeschlagenen Bereitstellung rückgängig zu machen.
- Sie verwenden nicht Infrastructure as Code (IaC) und Sie haben manuelle Aktualisierungen an Ihrer Infrastruktur vorgenommen, die zu einer unerwünschten Konfiguration geführt haben. Sie sind nicht in der Lage, die manuellen Änderungen effektiv zu verfolgen und rückgängig zu machen.
- Da Sie die erhöhte Häufigkeit Ihrer Bereitstellungen nicht gemessen haben, hat Ihr Team keinen Anreiz, den Umfang seiner Änderungen zu reduzieren und seine Rollback-Pläne für jede Änderung zu verbessern. Dies führt zu höheren Risiken und höheren Ausfallraten.
- Sie messen nicht die Gesamtdauer eines Ausfalls, der durch erfolglose Änderungen verursacht wird. Ihr Team ist nicht in der Lage, den Bereitstellungsprozess und die Effektivität des Wiederherstellungsplans zu priorisieren und zu verbessern.

Vorteile der Einführung dieser bewährten Methode: Wenn Sie einen Plan für die Wiederherstellung nach erfolglosen Änderungen haben, wird die durchschnittliche Wiederherstellungszeit (MTTR) minimiert und die Auswirkungen auf Ihr Unternehmen verringert.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Mithilfe einer konsistenten, dokumentierten Richtlinie und Praxis, die von den Release-Teams angewendet wird, kann ein Unternehmen planen, was bei nicht erfolgreichen Änderungen passieren soll. Unter bestimmten Umständen sollte die Richtlinie ein Forward-Fixing berücksichtigen. In allen Fällen sollte ein Fix-Forward- oder Rollback-Plan vor der Bereitstellung in der Live-Produktion gut dokumentiert und getestet werden, um die benötigte Zeit zum Rückgängigmachen einer Änderung zu minimieren.

Implementierungsschritte

1. Dokumentieren Sie die Richtlinien, nach denen Teams über wirksame Pläne verfügen müssen, wie Änderungen innerhalb eines bestimmten Zeitraums rückgängig gemacht werden können.
 - a. In den Richtlinien sollte festgelegt sein, wann eine Fix-Forward-Situation zulässig ist.
 - b. Erfordern Sie einen dokumentierten Rollback-Plan, auf den alle Beteiligten zugreifen können.
 - c. Geben Sie die Anforderungen für das Rollback an (z. B. wenn festgestellt wird, dass nicht autorisierte Änderungen vorgenommen wurden).
2. Analysieren Sie den Grad der Auswirkungen aller Änderungen für jede Komponente einer Workload.
 - a. Ermöglichen Sie die Standardisierung, Vorlagenerstellung und Vorautorisierung wiederholbarer Änderungen, sofern sie einem konsistenten Workflow folgen, der Änderungsrichtlinien durchsetzt.
 - b. Reduzieren Sie die potenziellen Auswirkungen jeder Änderung, indem Sie den Umfang der Änderung verringern, damit die Wiederherstellung weniger Zeit in Anspruch nimmt und weniger Auswirkungen auf das Unternehmen hat.
 - c. Stellen Sie sicher, dass die Rollback-Verfahren den Code in einen bekannt funktionierenden Zustand zurückversetzen, um Zwischenfälle nach Möglichkeit zu vermeiden.
3. Integrieren Sie Tools und Workflows, um Ihre Richtlinien programmgesteuert durchzusetzen.
4. Machen Sie Daten zu Änderungen für andere Workload-Besitzer sichtbar, um die Diagnose bei fehlgeschlagenen Änderungen, für die kein Rollback möglich ist, zu beschleunigen.

- a. Messen Sie den Erfolg dieser Methode anhand sichtbarer Änderungsdaten und identifizieren Sie iterative Verbesserungen.
5. Verwenden Sie Überwachungstools, um den Erfolg oder Misserfolg einer Bereitstellung zu überprüfen und so die Entscheidungsfindung beim Rollback zu beschleunigen.
6. Messen Sie die Dauer des Ausfalls bei einer erfolglosen Änderung, um Ihre Wiederherstellungspläne kontinuierlich zu verbessern.

Aufwand für den Implementierungsplan: Mittel

Ressourcen

Zugehörige bewährte Methoden:

- [OPS06-BP04 Automatisieren Sie Tests und Rollback](#)

Zugehörige Dokumente:

- [AWS Builders Library | Gewährleistung der Rollback-Sicherheit bei Bereitstellungen](#)
- [AWS Whitepaper | Änderungsmanagement in der Cloud](#)

Zugehörige Videos:

- [re:Invent 2019 | Der Amazon-Ansatz für die Hochverfügbarkeitsbereitstellung](#)

OPS06-BP02 Testbereitstellungen

Testen Sie Release-Verfahren in der Vorproduktion, indem Sie dieselbe Bereitstellungsconfiguration, dieselben Sicherheitskontrollen, Schritte und Verfahren wie in der Produktion verwenden. Stellen Sie sicher, dass alle bereitgestellten Schritte wie erwartet abgeschlossen wurden, z. B. das Überprüfen von Dateien, Konfigurationen und Services. Testen Sie alle Änderungen darüber hinaus mit Funktions-, Integrations- und Auslastungstests sowie Überwachungsverfahren, z. B. Zustandsprüfungen. Durch diese Tests können Sie Bereitstellungsprobleme frühzeitig erkennen und haben die Möglichkeit, sie vor der Produktion einzuplanen und zu beheben.

Sie können temporäre parallele Umgebungen erstellen, um jede Änderung zu testen. Automatisieren Sie die Bereitstellung der Testumgebungen mithilfe von Infrastructure as Code (IaC), um den

Arbeitsaufwand zu reduzieren und Stabilität, Konsistenz und schnellere Feature-Bereitstellung zu gewährleisten.

Gewünschtes Ergebnis: Ihr Unternehmen führt eine testgestützte Entwicklungskultur ein, die Testbereitstellungen einschließt. Dadurch wird sichergestellt, dass sich die Teams darauf konzentrieren, Werte für das Unternehmen zu schaffen, anstatt Releases zu verwalten. Die Teams werden bei der Identifizierung von Bereitstellungsrisiken frühzeitig einbezogen, um die geeigneten Maßnahmen zur Risikominderung festzulegen.

Typische Anti-Muster:

- Während Produktionseinführungen führen ungetestete Bereitstellungen häufig zu Problemen, die eine Fehlerbehebung und Eskalation erfordern.
- Ihr Release enthält Infrastructure as Code (IaC), wodurch vorhandene Ressourcen aktualisiert werden. Sie sind sich nicht sicher, ob IaC erfolgreich ausgeführt wird oder ob es Auswirkungen auf die Ressourcen gibt.
- Sie stellen ein neues Feature für Ihre Anwendung bereit. Sie funktioniert nicht wie beabsichtigt und dies fällt erst auf, als sie von betroffenen Benutzern gemeldet wird.
- Sie aktualisieren Ihre Zertifikate. Sie installieren versehentlich die Zertifikate für die falschen Komponenten, was unentdeckt bleibt und Auswirkungen auf Website-Benutzer hat, da keine sichere Verbindung zur Website hergestellt werden kann.

Vorteile der Nutzung dieser bewährten Methode: Durch umfangreiche Tests der Bereitstellungsverfahren und der durch sie eingeführten Änderungen in der Vorproduktion werden die potenziellen Auswirkungen der Bereitstellungsschritte auf die Produktion minimiert. Dies erhöht das Vertrauen bei der Produktionseinführung und minimiert den Support während des Betriebs, ohne die bereitgestellten Änderungen zu verlangsamen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Das Testen Ihres Bereitstellungsprozesses ist genauso wichtig wie das Testen der Änderungen, die sich aus der Bereitstellung ergeben. Dies kann erreicht werden, indem Sie Ihre Bereitstellungsschritte in einer Vorproduktionsumgebung testen, die die Produktion so genau wie möglich widerspiegelt. Häufig auftretende Probleme, z. B. unvollständige oder falsche Bereitstellungsschritte oder Fehlkonfigurationen, können so vor der Bereitstellung in der Produktionsumgebung erkannt werden. Darüber hinaus können Sie Ihre Wiederherstellungsschritte testen.

Kundenbeispiel

Im Rahmen seiner CI/CD-Pipeline (Continuous Integration and Continuous Delivery) führt AnyCompany Retail die definierten Schritte durch, die zur Veröffentlichung von Infrastruktur- und Softwareupdates für seine Kunden in einer Produktionsumgebung erforderlich sind. Die Pipeline besteht aus Vorabprüfungen zur Erkennung von Abweichungen (Erkennung von Änderungen an Ressourcen, die außerhalb von IaC vorgenommen wurden) bei Ressourcen vor der Bereitstellung sowie zur Validierung der Aktionen, die von IaC bei der Initiierung ausgeführt werden. Vor der erneuten Registrierung beim Load Balancer werden Bereitstellungsschritte validiert und z. B. sichergestellt, dass bestimmte Dateien und Konfigurationen vorhanden sind und Services ausgeführt werden und korrekt auf Zustandsprüfungen auf dem lokalen Host reagieren. Darüber hinaus führen alle Änderungen zu einer Reihe automatisierter Tests wie Funktions-, Sicherheits-, Regressions-, Integrations- und Auslastungstests.

Implementierungsschritte

1. Führen Sie Prüfungen vor der Installation durch, um die Vorproduktionsumgebung in der Produktionsumgebung zu spiegeln.
 - a. Verwenden Sie die [Drift-Erkennung](#), um zu erkennen, wann Ressourcen außerhalb von geändert wurden. CloudFormation
 - b. Verwenden Sie [Änderungssätze](#), um zu überprüfen, ob die Absicht eines Stack-Updates mit den Aktionen übereinstimmt, die CloudFormation bei der Initiierung des Änderungssatzes ausgeführt werden.
2. Dadurch wird ein manueller Genehmigungsschritt in [AWS CodePipeline](#) ausgelöst, um die Bereitstellung in der Vorproduktionsumgebung zu autorisieren.
3. Verwenden Sie Bereitstellungs-konfigurationen wie [AWS CodeDeploy AppSpec](#)-Dateien, um Bereitstellungs- und Validierungsschritte zu definieren.
4. [Integrieren Sie AWS CodeDeploy gegebenenfalls andere AWS Dienste](#) oder [integrieren Sie AWS CodeDeploy in Produkte und Services von Partnern](#).
5. [Überwachen Sie Bereitstellungen](#) mithilfe von Amazon CloudWatch, AWS CloudTrail, und SNS Amazon-Ereignisbenachrichtigungen.
6. Führen Sie nach der Bereitstellung automatisierte Tests durch, einschließlich Funktions-, Sicherheits-, Regressions-, Integrations- und Auslastungstests.
7. Führen Sie die [Fehlersuche](#) bei Problemen mit der Bereitstellung aus.
8. Eine erfolgreiche Validierung der zuvor genannten Schritte sollte einen manuellen Genehmigungsworkflow initiieren, um die Bereitstellung in der Produktion zu autorisieren.

Aufwand für den Implementierungsplan: Hoch

Ressourcen

Zugehörige bewährte Methoden:

- [OPS05-BP02 Testen und Validieren von Änderungen](#)

Zugehörige Dokumente:

- [AWS Builders' Library | Automatisieren von sicheren, automatischen Bereitstellungen | Testbereitstellungen](#)
- [AWS Whitepaper | Praktische Umsetzung von Continuous Integration und Continuous Delivery am AWS](#)
- [Die Geschichte von Apollo – Die Deployment Engine von Amazon](#)
- [Wie können Sie AWS CodeDeploy lokal testen und debuggen, bevor Sie Ihren Code versenden](#)
- [Integration von Netzwerkkonnektivitätstests in die Bereitstellung der Infrastruktur](#)

Zugehörige Videos:

- [re:Invent 2020 | Testen von Software und Systemen bei Amazon](#)

Zugehörige Beispiele:

- [Tutorial | Bereitstellen und ECS Amazon-Service mit einem Validierungstest](#)

OPS06-BP03 Einsetzen sicherer Bereitstellungsstrategien

Sichere Produktionseinführungen steuern den Fluss vorteilhafter Änderungen mit dem Ziel, die von den Kunden wahrgenommenen Auswirkungen dieser Änderungen zu minimieren. Die Sicherheitskontrollen bieten Prüfmechanismen, um die gewünschten Ergebnisse zu validieren und den Umfang der Auswirkungen von Fehlern zu begrenzen, die durch die Änderungen oder durch Fehler bei der Bereitstellung verursacht werden. Zu sicheren Rollouts können Strategien wie Feature-Flags, One-Box, Rolling (Canary-Releases), Immutable, Aufteilung des Datenverkehrs und Blau/Grün-Bereitstellungen gehören.

Gewünschtes Ergebnis: Ihr Unternehmen verwendet ein CI/CD-System (Continuous Integration/Continuous Delivery, kontinuierliche Integration/kontinuierliche Bereitstellung), das Funktionen

zur Automatisierung sicherer Rollouts bietet. Die Teams müssen angemessene sichere Rollout-Strategien anwenden.

Typische Anti-Muster:

- Sie stellen eine nicht erfolgreiche Änderung für die gesamte Produktion gleichzeitig bereit. Infolgedessen sind alle Kunden gleichzeitig betroffen.
- Ein Fehler, der bei einer gleichzeitigen Bereitstellung in allen Systemen auftritt, erfordert ein Notfall-Release. Die Korrektur für alle Kunden dauert mehrere Tage.
- Die Verwaltung der Produktionseinführung erfordert die Planung und Beteiligung mehrerer Teams. Dies schränkt Ihre Fähigkeit ein, Features für Ihre Kunden häufig zu aktualisieren.
- Sie führen eine veränderbare Bereitstellung durch, indem Sie Ihre vorhandenen Systeme ändern. Nachdem Sie festgestellt haben, dass die Änderung nicht erfolgreich war, müssen Sie die Systeme erneut ändern, um die alte Version wiederherzustellen, was die Wiederherstellungsdauer verlängert.

Vorteile der Nutzung dieser bewährten Methode: Automatisierte Bereitstellungen sorgen für ein ausgewogenes Verhältnis zwischen der Geschwindigkeit der Bereitstellungen und der konsistenten Bereitstellung nützlicher Änderungen für die Kunden. Die Begrenzung der Auswirkungen verhindert kostspielige Bereitstellungsfehler und maximiert die Fähigkeit der Teams, effizient auf Ausfälle zu reagieren.

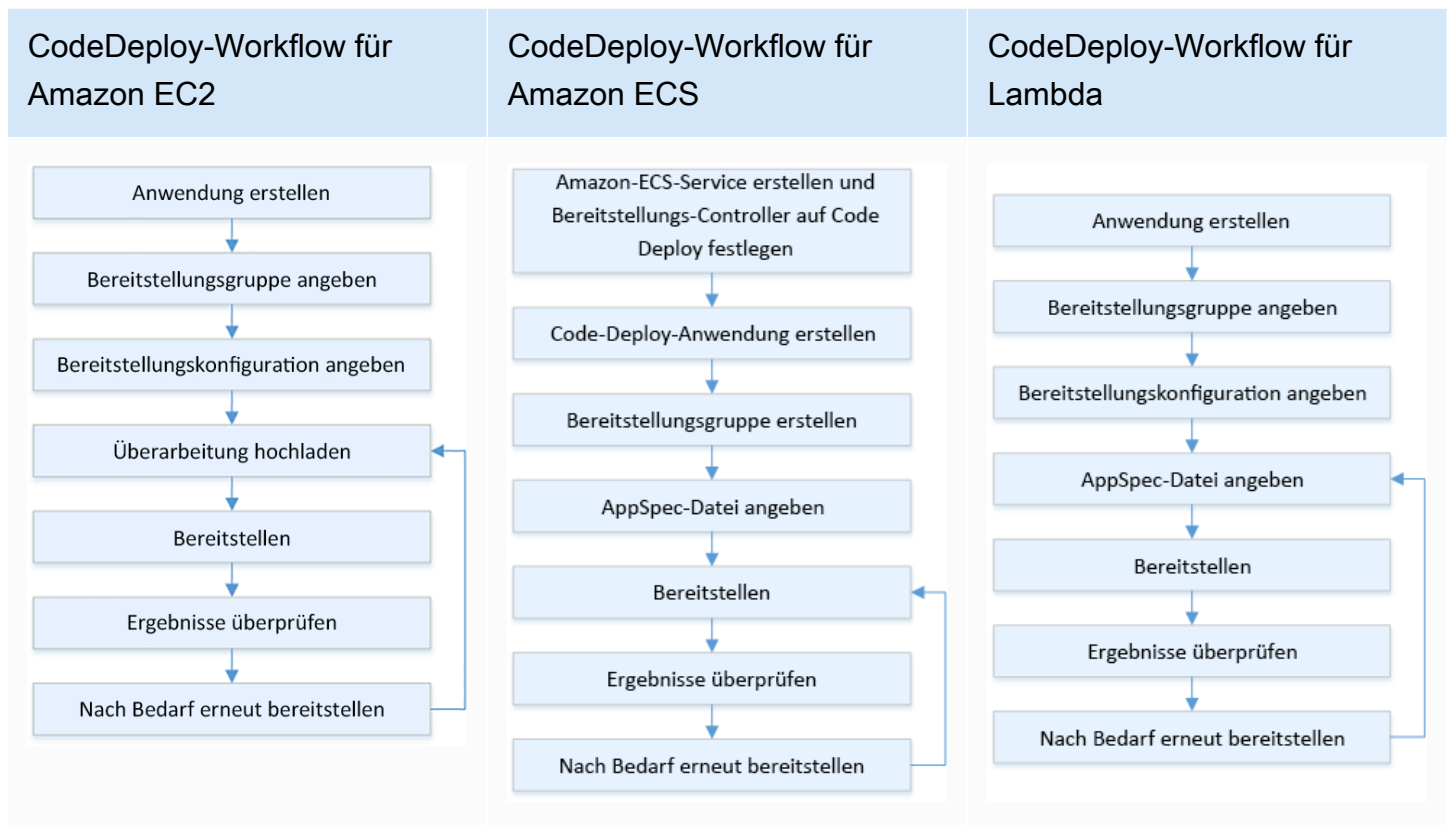
Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Ausfälle bei der kontinuierlichen Bereitstellung können zu einer verringerten Serviceverfügbarkeit und schlechten Kundenerfahrungen führen. Um die Anzahl erfolgreicher Implementierungen zu maximieren, sollten Sie im gesamten Release-Prozess Sicherheitskontrollen zur Minimierung von Bereitstellungsfehlern implementieren. Das Ziel sollte dabei sein, dass keine Bereitstellungsfehler auftreten.

Kundenbeispiel

AnyCompany Retail möchte Bereitstellungen mit minimalen bis gar keinen Ausfallzeiten erreichen, d. h. es soll während der Bereitstellung keine spürbaren Auswirkungen für die Benutzer geben. Um dies zu erreichen, hat das Unternehmen Bereitstellungsmuster festgelegt, z. B. fortlaufende und Blau/Grün-Bereitstellungen (siehe nachfolgendes Workflow-Diagramm). Alle Teams übernehmen eines oder mehrere dieser Muster in ihre CI/CD-Pipeline.



Implementierungsschritte

1. Verwenden Sie einen Genehmigungsworkflow, um die Reihenfolge der Produktionseinführungsschritte nach der Beförderung zur Produktion einzuleiten.
2. Verwenden Sie ein automatisiertes Bereitstellungssystem wie [AWS CodeDeploy](#). Die [Bereitstellungsoptionen](#) von AWS CodeDeploy schließen lokale Bereitstellungen für EC2/On-Premises und Blau/Grün-Bereitstellungen für EC2/On-Premises, AWS Lambda und Amazon ECS ein (siehe vorhergehendes Workflow-Diagramm).
 - a. [Integrieren Sie AWS CodeDeploy gegebenenfalls in andere AWS-Services](#) oder [integrieren Sie AWS CodeDeploy in Produkte und Services von Partnern](#).
3. Verwenden Sie Blau/Grün-Bereitstellungen für Datenbanken wie [Amazon Aurora](#) und [Amazon RDS](#).
4. [Überwachen Sie Bereitstellungen](#) mithilfe von Amazon CloudWatch, AWS CloudTrail und Amazon Simple Notification Service (Amazon SNS)-Ereignisbenachrichtigungen.
5. Führen Sie nach der Bereitstellung automatisierte Tests durch, einschließlich Funktions-, Sicherheits-, Regressions-, Integrations- und Auslastungstests.
6. Führen Sie die [Fehlersuche](#) bei Problemen mit der Bereitstellung aus.

Aufwand für den Implementierungsplan: Mittel

Ressourcen

Zugehörige bewährte Methoden:

- [OPS05-BP02 Testen und Validieren von Änderungen](#)
- [OPS05-BP09 Nehmen Sie häufige, kleine, reversible Änderungen vor](#)
- [OPS05-BP10 Vollständige Automatisierung von Integration und Bereitstellung](#)

Zugehörige Dokumente:

- [AWS Builders Library | Automatisierung sicherer, vollautomatischer Bereitstellungen | Produktionsbereitstellungen](#)
- [AWS Builders Library | Meine CI/CD-Pipeline ist mein Release Captain | Sichere, automatische Produktionseinführungen](#)
- [AWS-Whitepaper | Durchführung von Continuous Integration und Continuous Delivery in AWS | Bereitstellungsmethoden](#)
- [AWS CodeDeploy Benutzerhandbuch](#)
- [Arbeiten mit Bereitstellungsconfigurationen in AWS CodeDeploy](#)
- [Einrichten einer API Gateway-Canary-Release-Bereitstellung](#)
- [Amazon-ECS-Bereitlungstypen](#)
- [Vollständig verwaltete Blau/Grün-Bereitstellungen in Amazon Aurora und Amazon RDS](#)
- [Blau/Grün-Bereitstellungen mit AWS Elastic Beanstalk](#)

Zugehörige Videos:

- [re:Invent 2020 | Vollständige Automatisierung: Automatisieren der Pipelines für kontinuierliche Bereitstellung bei Amazon](#)
- [re:Invent 2019 | Der Amazon-Ansatz für die Hochverfügbarkeitsbereitstellung](#)

Zugehörige Beispiele:

- [Testen einer Blau/Grün-Beispielbereitstellung in AWS CodeDeploy](#)
- [Workshop | Entwicklung von CI/CD-Pipelines für Lambda-Canary-Bereitstellungen mit AWS CDK](#)

- [Workshop | Entwicklung Ihrer ersten DevOps-Blue/Green-Pipeline mit Amazon ECS](#)
- [Workshop | Entwicklung Ihrer ersten DevOps-Blue/Green-Pipeline mit Amazon EKS](#)
- [Workshop | EKS GitOps mit ArgoCD](#)
- [Workshop | CI/CD in AWS Workshop](#)
- [Implementierung von kontenübergreifendem CI/CD mit AWS SAM für containerbasierte Lambda-Funktionen](#)

OPS06-BP04 Automatisieren Sie Tests und Rollback

Um die Geschwindigkeit, Zuverlässigkeit und Sicherheit Ihres Bereitstellungsprozesses zu erhöhen, sollten Sie eine Strategie für automatisierte Test- und Rollback-Funktionen in Vorproduktions- und Produktionsumgebungen entwickeln. Automatisieren Sie Tests bei der Bereitstellung in der Produktion, um Interaktionen zwischen Mensch und System zu simulieren und die bereitgestellten Änderungen zu überprüfen. Automatisieren Sie das Rollback, um schnell zu einem als funktionierend bekannten Zustand zurückkehren zu können. Das Rollback sollte unter vordefinierten Bedingungen automatisch eingeleitet werden, z. B. wenn das gewünschte Ergebnis einer Änderung nicht erreicht wird oder wenn der automatisierte Test fehlschlägt. Die Automatisierung dieser beiden Aktivitäten verbessert Ihre Erfolgsquote bei Bereitstellungen, minimiert die Wiederherstellungszeit und reduziert die potenziellen Auswirkungen auf das Unternehmen.

Gewünschtes Ergebnis: Ihre automatisierten Tests und Rollback-Strategien sind in Ihre CI/CD-Pipeline (Continuous Integration/Continuous Delivery, kontinuierliche Integration/kontinuierliche Bereitstellung) integriert. Ihre Überwachung kann Validierungen anhand Ihrer Erfolgskriterien ausführen und bei einem Fehler ein automatisches Rollback einleiten. Dadurch werden die Auswirkungen auf Endbenutzer und Kunden minimiert. Wenn beispielsweise alle Testergebnisse den Anforderungen entsprechen, übertragen Sie Ihren Code in die Produktionsumgebung, wo automatisierte Regressionstests unter Verwendung derselben Testfälle eingeleitet werden. Wenn die Ergebnisse der Regressionstests nicht den Erwartungen entsprechen, wird im Pipeline-Workflow ein automatisiertes Rollback eingeleitet.

Typische Anti-Muster:

- Ihre Systeme sind nicht so konzipiert, dass sie mit kleineren Releases aktualisiert werden können. Daher haben Sie Schwierigkeiten, die Bulk-Änderungen während einer fehlgeschlagenen Bereitstellung rückgängig zu machen.
- Ihr Bereitstellungsprozess besteht aus einer Reihe manueller Schritte. Nachdem Sie Änderungen an Ihrer Workload bereitgestellt haben, beginnen Sie mit den Tests nach der Bereitstellung.

Danach bemerken Sie, dass Ihre Workload nicht mehr funktioniert und die Verbindung der Kunden getrennt wird. Sie starten das Rollback zur vorherigen Version. All diese manuellen Schritte verzögern die allgemeine Systemwiederherstellung und wirken sich nachhaltig auf Ihre Kunden aus.

- Sie haben Zeit dafür aufgewendet, automatisierte Testfälle für Funktionen zu entwickeln, die in Ihrer Anwendung nicht häufig verwendet werden. Dadurch amortisiert sich die Investition in Ihre automatisierten Testfunktionen nur schlecht.
- Ihre Version besteht aus Anwendungs-, Infrastruktur-, Patch- und Konfigurations-Updates, die voneinander unabhängig sind. Sie haben jedoch nur eine CI/CD-Pipeline, die alle Änderungen gleichzeitig bereitstellt. Ein Fehler in einer Komponente zwingt Sie, alle Änderungen rückgängig zu machen, wodurch Ihr Rollback komplex und ineffizient wird.
- Ihr Team schließt die Programmierarbeiten im ersten Sprint ab und beginnt mit dem zweiten Sprint, aber Ihr Plan sieht Tests erst im dritten Sprint vor. Deshalb haben automatisierte Tests Fehler aus dem ersten Sprint aufgedeckt, die behoben werden müssen, bevor mit dem Testen der Ergebnisse von Sprint zwei begonnen werden kann. Der gesamte Release verzögert sich, wodurch der Wert Ihrer automatisierten Tests erheblich verringert wird.
- Ihre automatisierten Regressionstestfälle für die Produktionsversion sind abgeschlossen, aber Sie überwachen den Zustand der Workloads nicht. Da Sie nicht sehen können, ob der Dienst neu gestartet wurde oder nicht, sind Sie sich nicht sicher, ob ein Rollback erforderlich ist oder bereits stattgefunden hat.

Vorteile der Nutzung dieser bewährten Methode: Automatisierte Tests erhöhen die Transparenz Ihres Testprozesses und Ihre Fähigkeit, mehr Funktionen in kürzerer Zeit abzudecken. Durch das Testen und Validieren von Änderungen in der Produktionsphase können Sie Probleme sofort identifizieren. Die Verbesserung der Konsistenz mit automatisierten Testtools ermöglicht eine bessere Fehlererkennung. Durch das automatische Rollback zur vorherigen Version werden die Auswirkungen für Ihre Kunden minimiert. Ein automatisiertes Rollback sorgt letztendlich für mehr Vertrauen in Ihre Bereitstellungsfunktionen, da es die Auswirkungen auf Ihr Unternehmen verringert. Insgesamt verringern time-to-delivery sich diese Fähigkeiten bei gleichbleibender Qualität.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Automatisieren Sie die Tests von bereitgestellten Umgebungen, um schneller die gewünschten Ergebnisse zu erreichen. Automatisieren Sie den Rollback zu einem bekanntermaßen funktionierenden vorherigen Zustand, wenn die zuvor definierten Ergebnisse nicht erzielt werden.

So können Sie die Wiederherstellungszeit minimieren und verringern Fehler, die durch manuelle Prozesse entstehen. Integrieren Sie Testtools in Ihren Pipeline-Workflow, um manuelle Eingaben konsistent zu testen und zu minimieren. Priorisieren Sie die Automatisierung von Testfällen, z. B. Tests, die die größten Risiken minimieren und die bei jeder Änderung häufig durchgeführt werden müssen. Automatisieren Sie außerdem das Rollback auf Grundlage bestimmter Bedingungen, die in Ihrem Testplan vordefiniert sind.

Implementierungsschritte

1. Richten Sie einen Testlebenszyklus für Ihren Entwicklungslebenszyklus ein, in dem jede Phase des Testprozesses definiert wird. Dies reicht von der Anforderungsplanung über die Testfallentwicklung, die Toolkonfiguration, das automatisierte Testen bis hin zum Abschluss des Testfalls.
 - a. Erstellen Sie anhand Ihrer gesamten Teststrategie einen Workload-spezifischen Testansatz.
 - b. Ziehen Sie eine Strategie für kontinuierliche Tests während des gesamten Entwicklungszyklus in Erwägung.
2. Wählen Sie in Abhängigkeit von Ihren Geschäftsanforderungen und Pipeline-Investitionen automatisierte Tools für Tests und Rollbacks aus.
3. Entscheiden Sie, welche Testfälle Sie automatisieren möchten und welche manuell durchgeführt werden sollen. Dies kann auf Grundlage des geschäftlichen Nutzens des getesteten Features definiert werden. Informieren Sie alle Teammitglieder über diesen Plan und legen Sie fest, wer für die Durchführung manueller Tests verantwortlich ist.
 - a. Wenden Sie automatisierte Testfunktionen auf bestimmte Testfälle an, die für die Automatisierung sinnvoll sind, z. B. wiederholbare oder häufig ausgeführte Fälle, Fälle, die sich wiederholende Aufgaben erfordern, oder solche, die für mehrere Konfigurationen erforderlich sind.
 - b. Definieren Sie Skripts für die Testautomatisierung sowie die Erfolgskriterien im Automatisierungstool, sodass eine kontinuierliche Workflow-Automatisierung initiiert werden kann, wenn bei bestimmten Fällen Fehler auftreten.
 - c. Definieren Sie spezifische Fehlerkriterien für das automatisierte Rollback.
4. Priorisieren Sie die Testautomatisierung, um konsistente Ergebnisse mit einer gründlichen Testfallentwicklung zu erzielen, bei der Komplexität und menschliche Interaktion ein höheres Ausfallrisiko darstellen.
5. Integrieren Sie Ihre automatisierten Test- und Rollback-Tools in Ihre CI/CD-Pipeline.
 - a. Entwickeln Sie klare Erfolgskriterien für Ihre Änderungen.

- b. Überwachen und beobachten Sie Ihre Umgebung, um diese Kriterien zu erkennen und Änderungen automatisch rückgängig zu machen, wenn bestimmte Rollback-Kriterien erfüllt werden.
6. Führen Sie verschiedene Arten automatisierter Produktionstests durch, z. B.:
- a. A/B-Tests zur Anzeige von Ergebnissen im Vergleich zur aktuellen Version zwischen zwei Benutzertestgruppen.
 - b. Canary-Tests, mit denen Sie Ihre Änderung für eine Untergruppe von Benutzern bereitstellen können, bevor Sie sie für alle freigeben.
 - c. Testen mit Feature-Flags, wobei jeweils eine einzelne Funktion der neuen Version außerhalb der Anwendung ein- und ausgeschaltet werden kann, sodass alle neuen Funktionen einzeln validiert werden können.
 - d. Regressionstests zur Überprüfung neuer Funktionen mit bestehenden, miteinander verbundenen Komponenten.
7. Überwachen Sie die betrieblichen Aspekte der Anwendung, Transaktionen und Interaktionen mit anderen Anwendungen und Komponenten. Entwickeln Sie Berichte, um den Erfolg von Änderungen nach Workload aufzuzeigen, sodass Sie erkennen können, welche Teile der Automatisierung und des Workflows weiter optimiert werden können.
- a. Entwickeln Sie Testergebnisberichte, anhand derer Sie schnell entscheiden können, ob Rollback-Verfahren eingeleitet werden sollten oder nicht.
 - b. Implementieren Sie eine Strategie, die ein automatisiertes Rollback auf Grundlage vordefinierter Fehlerbedingungen ermöglicht, die sich aus einer oder mehreren Ihrer Testmethoden ergeben.
8. Entwickeln Sie Ihre automatisierten Testfälle so, dass sie bei zukünftigen wiederholbaren Änderungen wiederverwendet werden können.

Aufwand für den Implementierungsplan: Mittel

Ressourcen

Zugehörige bewährte Methoden:

- [OPS06-BP01 Plan für erfolglose Änderungen](#)
- [OPS06-BP02 Testbereitstellungen](#)

Zugehörige Dokumente:

- [AWS Builders Library | Gewährleistung der Rollback-Sicherheit bei Bereitstellungen](#)
- [Eine Bereitstellung erneut bereitstellen und rückgängig machen mit AWS CodeDeploy](#)
- [8 bewährte Methoden für die Automatisierung Ihrer Bereitstellungen mit AWS CloudFormation](#)

Zugehörige Beispiele:

- [Testen von Benutzeroberflächen ohne Server mit Selenium, AWS Lambda, und Developer Tools AWS FargateAWS](#)

Zugehörige Videos:

- [re:Invent 2020 | Vollständige Automatisierung: Automatisieren der Pipelines für kontinuierliche Bereitstellung bei Amazon](#)
- [re:Invent 2019 | Der Amazon-Ansatz für die Hochverfügbarkeitsbereitstellung](#)

OPS 7. Wie bringen Sie in Erfahrung, ob Sie für die Unterstützung eines Workloads bereit sind?

Bewerten Sie die Betriebsbereitschaft Ihrer Workloads, von Prozessen und Verfahren sowie Ihrer Mitarbeiter, damit Sie die betrieblichen Risiken im Zusammenhang mit Ihrer Workload genau kennen.

Best Practices

- [OPS07-BP01 Sicherstellen des Know-hows der Mitarbeiter](#)
- [OPS07-BP02 Sicherstellen einer konsistenten Prüfung der betrieblichen Bereitschaft](#)
- [OPS07-BP03 Verwenden von Runbooks zur Durchführung von Verfahren](#)
- [OPS07-BP04 Verwenden von Playbooks zum Untersuchen von Problemen](#)
- [OPS07-BP05 Treffen fundierter Entscheidungen für die Bereitstellung von Systemen und Änderungen](#)
- [OPS07-BP06 Erstellen von Supportplänen für Produktions-Workloads](#)

OPS07-BP01 Sicherstellen des Know-hows der Mitarbeiter

Stellen Sie einen Mechanismus bereit, mit dem Sie prüfen können, ob Sie über ausreichend trainierte Mitarbeiter zur Unterstützung der Workload verfügen. Sie müssen für die Plattform und die Services, die Ihre Workload ausmachen, trainiert sein. Stellen Sie ihnen die Informationen zur

Verfügung, die sie zum Betrieb des Workloads benötigen. Sie müssen über genügend geschulte Mitarbeiter verfügen, um den normalen Betrieb der Workload zu unterstützen und auftretende Probleme zu beheben. Sorgen Sie für genügend Mitarbeiter, sodass Sie Bereitschaftsdienste und Urlaubsvertretungen abwechseln können, um Burnouts zu vermeiden.

Gewünschtes Ergebnis:

- Es gibt genügend trainierte Mitarbeiter, um die Workload im Rahmen des Verfügbarkeitszeitraums zu unterstützen.
- Sie trainieren Ihre Mitarbeiter für die Software und Services, die Ihre Workload ausmachen.

Typische Anti-Muster:

- Bereitstellen einer Workload ohne Teammitglieder, die für den Betrieb der Plattform und der genutzten Services trainiert sind.
- Sie haben nicht genug Mitarbeiter, um wechselnde Bereitschaftsdienste oder Urlaubszeiten abzudecken.

Vorteile der Nutzung dieser bewährten Methode:

- Wenn Sie über qualifizierte Teammitglieder verfügen, können diese Ihre Workload effektiv unterstützen.
- Mit einer ausreichenden Anzahl von Teammitgliedern können Sie den Workload und die Rotation der Bereitschaftsdienste unterstützen und gleichzeitig das Risiko eines Burnouts verringern.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Validieren Sie, ob ausreichend trainierte Mitarbeiter für den Support des Workloads vorhanden sind. Vergewissern Sie sich, dass Sie über genügend Teammitglieder verfügen, um die normalen operativen Aktivitäten, einschließlich Einsatzbereitschaftsdienste, abzudecken.

Kundenbeispiel

AnyCompany Retail sorgt dafür, dass die Teams für die Workload angemessen besetzt und trainiert sind. Es gibt genügend Ingenieure, um wechselnde Bereitschaftsdienste zu unterstützen. Die Mitarbeiter erhalten Training, um die Software und die Workload-Plattform zu nutzen. Sie werden

außerdem ermutigt, Zertifizierungen zu erwerben. Es gibt so viele Mitarbeiter, dass Urlaub möglich ist, ohne dass die Abdeckung der Workload und der rotierenden Bereitschaftsdienste unterbrochen werden muss.

Implementierungsschritte

1. Weisen Sie eine ausreichende Anzahl von Mitarbeitern zur Ausführung und Unterstützung Ihres Workloads zu, einschließlich Bereitschaftsdiensten, Sicherheitsproblemen und Lebenszyklusereignissen, z. B. Supportende und Zertifikatrotation.
2. Trainieren Sie die Mitarbeiter im Umgang mit der Software und den Plattformen, aus denen Ihr Workload besteht.
 - a. [AWS Training and Certification](#) bietet eine Bibliothek mit Kursen zu AWS. Es gibt kostenlose und kostenpflichtige Kurse – online und vor Ort.
 - b. [AWS organisiert Veranstaltungen und Webinare](#), bei denen Sie von AWS-Experten lernen.
3. Führen Sie Folgendes regelmäßig aus.
 - Bewerten Sie regelmäßig Größe und Kompetenzen des Teams, wenn sich operative Bedingungen und Workloads verändern.
 - Passen Sie die Größe und Fähigkeiten des Teams an die operativen Anforderungen an.
 - Überprüfen Sie, ob die nötigen Fähigkeiten und Kapazitäten vorhanden sind, um [geplante Lebenszyklusereignisse](#), ungeplante Sicherheitsprobleme und operative Benachrichtigungen mittels AWS Health zu behandeln.

Aufwand für den Implementierungsplan: Hoch. Das Einstellen und Trainieren eines Teams zur Unterstützung einer Workload kann einen erheblichen Aufwand darstellen, bietet aber langfristig einen bedeutenden Nutzen.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS11-BP04 Wissensmanagement durchführen](#) – Die Teammitglieder müssen über die notwendigen Informationen verfügen, um die Workload zu betreiben und zu unterstützen. Der Schlüssel dazu ist das Wissensmanagement.

Zugehörige Dokumente:

- [AWS-Veranstaltungen und -Webinare](#)

- [AWS Training and Certification](#)

OPS07-BP02 Sicherstellen einer konsistenten Prüfung der betrieblichen Bereitschaft

Verwenden Sie Operational Readiness Reviews (ORRs, Überprüfungen der Einsatzbereitschaft), um zu prüfen, ob Sie Ihre Workload betreiben können. ORR ist ein bei Amazon entwickelter Mechanismus zur Prüfung, ob Teams ihre Workloads in sicherer Weise betreiben können. ORR bezeichnet einen Prüfungs- und Inspektionsprozess anhand einer Checkliste mit Anforderungen. Dies ist ein Selfservicevorgang, mit dem Teams ihre Workloads zertifizieren. ORRs beinhalten bewährte Methoden aus unseren jahrelangen Erfahrungen bei der Erstellung von Software.

Eine ORR-Checkliste besteht aus Architekturempfehlungen, betrieblichen Prozessen, Ereignismanagement und Freigabequalität. Unser Correction of Error (CoE)-Prozess ist dafür eine sehr wichtige Grundlage. Ihre eigene Analyse nach einem Vorfall sollte die Weiterentwicklung Ihrer eigenen ORR unterstützen. Bei einer ORR geht es nicht nur um die Umsetzung bewährter Methoden, sondern auch darum, das erneute Auftreten von Ereignissen zu verhindern. Schließlich können auch Sicherheit, Governance und Compliance zu einer ORR gehören.

Führen Sie eine ORR durch, bevor eine Workload zur allgemeinen Verfügbarkeit gestartet wird, und anschließend während des gesamten Softwareentwicklungslebenszyklus. Die Durchführung der ORR vor dem Start verbessert Ihre Fähigkeit zum sicheren Betrieb der Workload. Führen Sie die ORR auf der Workload regelmäßig erneut durch, um Abweichungen von bewährten Methoden zu erkennen. Sie können ORR-Checklisten für neue Serviceeinführungen oder für regelmäßige Prüfungen haben. So bleiben Sie hinsichtlich der neuen bewährten Methoden auf dem Laufenden und können Erfahrungen aus Analysen nach Vorfällen einarbeiten. Wenn Sie mit der Cloud immer vertrauter werden, können Sie ORR-Anforderungen als Standardelemente in Ihre Architektur einbauen.

Gewünschtes Ergebnis: Sie haben eine ORR-Checkliste mit bewährten Methoden für Ihre Organisation. ORRs werden vor dem Start von Workloads durchgeführt. ORRs werden im Laufe des Workload-Lebenszyklus regelmäßig durchgeführt.

Typische Anti-Muster:

- Sie starten eine Workload, ohne zu wissen, ob Sie diese betreiben können.
- Governance- und Sicherheitsanforderungen gehören nicht zur Zertifizierung einer Workload für den Start.
- Workloads werden nicht regelmäßig erneut bewertet.
- Workloads werden gestartet, ohne dass erforderliche Verfahren eingerichtet sind.

- Sie erleben die Wiederholung von Ausfällen mit der gleichen Ursache bei mehreren Workloads.

Vorteile der Nutzung dieser bewährten Methode:

- Ihre Workloads beinhalten bewährte Methoden für Architektur, Prozess und Management.
- Erkenntnisse werden in Ihren ORR-Prozess integriert.
- Workloads werden gestartet, wenn erforderliche Verfahren eingerichtet sind.
- ORRs werden über den gesamten Softwarelebenszyklus Ihrer Workloads hinweg ausgeführt.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Eine ORR ist zweierlei: ein Verfahren und eine Checkliste. Ihr ORR-Verfahren sollte von ihrer Organisation übernommen und von der Unternehmensleitung unterstützt werden. ORRs müssen mindestens durchgeführt werden, bevor Workloads zur allgemeinen Verfügbarkeit gestartet werden. Führen Sie die ORR während des gesamten Lebenszyklus der Softwareentwicklung durch, um ihn bei bewährten Methoden oder neuen Anforderungen aktuell zu halten. Die ORR-Checkliste sollte Konfigurationselemente, Sicherheits- und Governance-Elemente sowie bewährte Methoden aus Ihrer Organisation enthalten. Mit der Zeit können Sie Services wie [AWS Config](#), [AWS Security Hub CSPM](#) und [Integritätsschutz von AWS Control Tower](#) verwenden, um bewährte Methoden aus der ORR in den Integritätsschutz für die automatische Erkennung optimaler Verfahrensweisen aufzunehmen.

Kundenbeispiel

Nach mehreren Produktionsvorfällen entschied sich AnyCompany Retail, einen ORR-Prozess zu implementieren. Das Unternehmen erstellte eine Checkliste mit bewährten Methoden sowie Governance- und Complianceanforderungen und Erfahrungen aus früheren Ausfällen. Für neue Workloads werden vor dem Start ORRs durchgeführt. Für jede Workload wird eine jährliche ORR mit einer Teilmenge der bewährten Methoden durchgeführt, um neue bewährte Methoden und Anforderungen umzusetzen, die der ORR-Checkliste hinzugefügt werden. Mit der Zeit verwendete AnyCompany Retail [AWS Config](#) zur Aufdeckung einiger bewährter Methoden, was den ORR-Prozess beschleunigte.

Implementierungsschritte

Weitere Informationen zu ORRs finden Sie im [Whitepaper zur Überprüfung der betrieblichen Bereitschaft \(ORR\)](#). Hier finden Sie ausführliche Informationen zur Geschichte des ORR-Verfahrens,

zum Aufbau Ihrer eigenen ORR-Praxis und zur Erstellung Ihrer ORR-Checkliste. Die folgenden Schritte sind eine verkürzte Version dieses Dokuments. Für ein vertieftes Verständnis des ORR-Konzepts und der Erstellung eigener ORRs empfehlen wir, das Whitepaper zu lesen.

1. Bringen Sie die wichtigsten Stakeholder zusammen, darunter auch Vertreter aus den Bereichen Sicherheit, Operations und Entwicklung.
2. Lassen Sie alle Stakeholder mindestens eine Anforderung beisteuern. Versuchen Sie für den ersten Durchgang die Anzahl der Elemente auf höchstens dreißig zu beschränken.
 - [Anhang B: Beispielfragen für ORRs](#) aus dem ORR-Whitepaper enthält Beispielfragen, die Ihnen beim Start helfen können.
3. Fassen Sie Ihre Anforderungen in einer Tabelle zusammen.
 - Sie können [benutzerdefinierte Linsen](#) im [AWS Well-Architected Tool](#) verwenden, um Ihre ORR zu entwickeln und an Ihre Konten und die AWS-Organisation weiterzugeben.
4. Identifizieren Sie eine Workload für die ORR. Ideal ist dafür eine Pre-Launch-Workload oder eine interne Workload.
5. Gehen Sie die ORR-Checkliste durch und notieren Sie alle Erkenntnisse. Erkenntnisse sind möglicherweise akzeptabel, wenn eine Behebung vorhanden ist. Fügen Sie alle Erkenntnisse ohne Behebung Ihrer Liste hinzu und implementieren Sie die Behebungen vor dem Start.
6. Fügen Sie Ihrer ORR-Checkliste stets weitere bewährte Methoden und Anforderungen hinzu.

Support-Kunden mit Enterprise Support können den [Workshop zur Überprüfung der betrieblichen Bereitschaft](#) bei ihrem Technical Account Manager anfordern. Der Workshop ist eine interaktive Working Backwards-Sitzung zur Entwicklung Ihrer eigenen ORR-Checkliste.

Aufwand für den Implementierungsplan: Hoch. Die Einführung einer ORR-Praxis in Ihrer Organisation erfordert die Unterstützung durch Führungskräfte und alle Stakeholder. Erstellen und aktualisieren Sie die Checkliste mit Beiträgen aus der gesamten Organisation.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS01-BP03 Bewertung der Governance-Anforderungen](#) – Governance-Anforderungen passen perfekt zu einer ORR-Checkliste.
- [OPS01-BP04 Bewerten der Compliance-Anforderungen](#) – Complianceanforderungen werden manchmal auf ORR-Checklisten berücksichtigt. Ansonsten sind sie ein separater Prozess.

- [OPS03-BP07 Ressourcenteams angemessen](#) – Die Teamkapazität ist ein guter Kandidat für eine ORR-Anforderung.
- [OPS06-BP01 Plan für erfolglose Änderungen](#) – Vor dem Start Ihrer Workload muss ein Rollback- oder Rollforward-Plan eingerichtet werden.
- [OPS07-BP01 Sicherstellen des Know-hows der Mitarbeiter](#) – Zur Unterstützung einer Workload benötigen Sie das erforderliche Personal.
- [SEC01-BP03 Identifizieren und Validieren von Kontrollzielen](#) – Sicherheitskontrollziele sind hervorragende ORR-Anforderungen.
- [REL13-BP01 Definieren von Wiederherstellungszielen bei Ausfällen und Datenverlusten](#) – Notfallwiederherstellungspläne sind eine gute ORR-Anforderung.
- [COST02-BP01 Entwickeln von Richtlinien auf Basis Ihrer Organisationsanforderungen](#) – Kostenmanagementrichtlinien sind für Ihre ORR-Checkliste gut geeignet.

Zugehörige Dokumente:

- [AWS Control Tower – Integritätsschutz in AWS Control Tower](#)
- [AWS Well-Architected Tool – Fokusbereiche](#)
- [Operational Readiness Review Template von Adrian Hornsby](#)
- [Whitepaper zur Überprüfung der betrieblichen Bereitschaft \(ORR\)](#)

Zugehörige Videos:

- [AWS Supports You | Entwickeln einer effektiven Überprüfung der betrieblichen Bereitschaft \(ORR\)](#)

Zugehörige Beispiele:

- [Sample Operational Readiness Review \(ORR\)-Fokusbereich](#)

Zugehörige Services:

- [AWS Config](#)
- [AWS Control Tower](#)
- [AWS Security Hub CSPM](#)
- [AWS Well-Architected Tool](#)

OPS07-BP03 Verwenden von Runbooks zur Durchführung von Verfahren

Ein Runbook ist ein dokumentierter Prozess für das Erreichen eines bestimmten Ergebnisses. Runbooks bestehen aus einer Reihe von Schritten, die befolgt werden sollen, um ein Ergebnis zu erzielen. Runbooks werden schon seit den frühen Tagen der Luftfahrt verwendet. Im Cloud-Bereich werden Runbooks verwendet, um die Risiken zu reduzieren und die gewünschten Ergebnisse zu erzielen. In der einfachsten Form ist ein Runbook eine Checkliste für die Durchführung einer Aufgabe.

Runbooks stellen einen kritischen Teil der Ausführung Ihrer Workload dar. Vom Onboarding eines neuen Teammitglieds bis zur Bereitstellung einer Hauptversion – Runbooks stellen kodifizierte Prozesse dar, mit denen unabhängig von der ausführenden Person konsistente Ergebnisse erzielt werden können. Runbooks sollten an einer zentralen Stelle veröffentlicht werden. Wenn sich der Prozess verändert, sollten sie aktualisiert werden; dies stellt eine zentrale Komponente des Änderungsmanagements dar. Sie sollten auch Anleitungen für Fehlerbehandlung, Tools, Berechtigungen, Ausnahmen und Eskalationen enthalten, falls ein Problem auftritt.

Wenn sich Ihre Organisation entwickelt, sollten Sie mit der Automatisierung von Runbooks beginnen. Sie sollten zunächst Runbooks automatisieren, die kurz sind und häufig verwendet werden. Verwenden Sie Skriptsprachen, um Schritte zu automatisieren oder ihre Ausführung zu vereinfachen. Nach der Automatisierung der ersten Runbooks können Sie komplexere Runbooks automatisieren. Mit der Zeit sollten die meisten Ihrer Runbooks auf die eine oder andere Art automatisiert werden.

Gewünschtes Ergebnis: Ihr Team besitzt eine Sammlung von Schritt-für-Schritt-Anleitungen für die Ausführung von Workload-Aufgaben. Die Runbooks enthalten Angaben zum gewünschten Ergebnis sowie zu notwendigen Tools und Berechtigungen. Darüber hinaus stellen sie Anleitungen für die Fehlerbehandlung bereit. Sie werden an einem zentralen Ort (Versionskontrollsystem) gespeichert und regelmäßig aktualisiert. Ihre Runbooks bieten Ihren Teams beispielsweise die Möglichkeit, AWS Health-Ereignisse für kritische Konten bei Anwendungsalarmen, Betriebsproblemen und geplanten Lebenszykluseignissen zu überwachen, zu kommunizieren und darauf zu reagieren.

Typische Anti-Muster:

- Verlassen auf das Gedächtnis, um die einzelnen Schritte in einem Prozess durchzuführen.
- Manuelle Bereitstellung von Änderungen ohne Checkliste.
- Verschiedene Teammitglieder führen den gleichen Prozess aus, aber mit unterschiedlichen Schritten oder Ergebnissen.
- Runbooks sind nicht mehr mit Systemänderungen und Automatisierungen synchronisiert.

Vorteile der Nutzung dieser bewährten Methode:

- Reduzierung der Fehlerquoten für manuelle Aufgaben.
- Prozesse werden konsistent ausgeführt.
- Neue Teammitglieder können schneller mit der Ausführung von Aufgaben beginnen.
- Runbooks können automatisiert werden, um den Aufwand zu reduzieren.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Runbooks können verschiedene Formen annehmen, abhängig vom Entwicklungsstand Ihrer Organisation. Sie sollten mindestens aus einem Schritt-für-Schritt-Textdokument bestehen. Das gewünschte Ergebnis sollte klar angegeben werden. Dokumentieren Sie klar die notwendigen Berechtigungen oder Tools. Stellen Sie für den Fall, dass etwas nicht funktioniert, detaillierte Anleitungen für Fehlerbehandlung und Eskalation bereit. Nennen Sie die Person, die für das Runbook verantwortlich ist, und veröffentlichen Sie es an einer zentralen Stelle. Validieren Sie das Runbook, nachdem Sie es dokumentiert haben, indem Sie es von einem Teammitglied ausführen lassen. Mit der weiteren Entwicklung der Verfahren sollten Sie Ihre Runbooks entsprechend Ihrem Prozess für das Änderungsmanagement aktualisieren.

Ihre textbasierten Runbooks sollten mit zunehmender Entwicklung Ihrer Organisation automatisiert werden. Mit Services wie [AWS-Systems-Manager-Automatisierungen](#) können Sie Textdateien zu Automatisierungen transformieren, die Sie für Ihre Workload ausführen können. Diese Automatisierungen können als Reaktion auf Ereignisse ausgeführt werden, was den operativen Aufwand für die Wartung der Workload reduziert. AWS Systems Manager Automation bietet außerdem ein [visuelles Low-Code-Design](#), um Automatisierungs-Runbooks einfacher zu erstellen.

Kundenbeispiel

AnyCompany Retail muss während Softwarebereitstellungen die Datenbankschemata aktualisieren. Das Cloud Operations-Team entwickelt gemeinsam mit dem Datenbankverwaltungsteam ein Runbook für die manuelle Bereitstellung dieser Änderungen. In diesem Runbook werden die einzelnen Prozessschritte in Form einer Checkliste aufgelistet. Es enthält für den Fall, dass es ein Problem gibt, auch einen Abschnitt zur Fehlerbehandlung. Das Runbook wird wie die übrigen Runbooks im internen Wiki veröffentlicht. Das Cloud Operations-Team plant, das Runbook in der Zukunft zu automatisieren.

Implementierungsschritte

Wenn Sie noch kein Dokumenten-Repository besitzen, dann ist ein Repository für die Versionskontrolle hervorragend als Grundlage für Ihre Runbook-Bibliothek geeignet. Sie können Ihre Runbooks mithilfe von Markdown erstellen. Wir haben eine Runbook-Beispielvorlage bereitgestellt, die Sie für die Erstellung von Runbooks verwenden können.

```
# Runbook Title
## Runbook Info
| Runbook ID | Description | Tools Used | Special Permissions | Runbook Author | Last
Updated | Escalation POC |
|-----|-----|-----|-----|-----|-----|-----|
| RUN001 | What is this runbook for? What is the desired outcome? | Tools | Permissions
| Your Name | 2022-09-21 | Escalation Name |
## Steps
1. Step one
2. Step two
```

1. Wenn Sie noch kein Dokumentations-Repository oder -Wiki besitzen, sollten Sie in Ihrem Versionskontrollsystem ein neues Versionskontroll-Repository erstellen.
2. Identifizieren Sie einen Prozess, für den es kein Runbook gibt. Ein idealer Prozess hierfür ist ein Prozess, der halbregelmäßig ausgeführt wird, nur wenige Schritte enthält und bei Fehlern nur geringe Auswirkungen hat.
3. Erstellen Sie in Ihrem Dokument-Repository ein neues Markdown-Entwurfsdokument auf der Basis der Vorlage. Füllen Sie den Runbook-Titel und die Pflichtfelder unter Runbook-Informationen aus.
4. Füllen Sie ab dem ersten Schritt den Abschnitt Schritte im Runbook aus.
5. Geben Sie das Runbook einem Teammitglied. Lassen Sie das Teammitglied das Runbook ausführen, um die Schritte zu validieren. Aktualisieren Sie das Runbook, wenn etwas fehlt oder unklar ist.
6. Veröffentlichen Sie das Runbook in Ihrem internen Dokumentationsspeicher. Informieren Sie Ihr Team und die übrigen Stakeholder über das Runbook, nachdem es veröffentlicht wurde.
7. Mit der Zeit entsteht dadurch eine Bibliothek von Runbooks. Beginnen Sie mit der Automatisierung von Runbooks, wenn diese Bibliothek wächst.

Aufwand für den Implementierungsplan: Niedrig. Eine Schritt-für-Schritt-Anleitung in Textform ist der Mindeststandard für ein Runbook. Die Automatisierung von Runbooks kann den Implementierungsaufwand erhöhen.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS02-BP02 Prozesse und Verfahren haben feste Besitzer](#)
- [OPS07-BP04 Verwenden von Playbooks zum Untersuchen von Problemen](#)
- [OPS10-BP01 Verwenden eines Prozesses für die Bewältigung von Ereignissen, Vorfällen und Problemen](#)
- [OPS10-BP02 Implementieren eines Prozesses für jede Warnmeldung](#)
- [OPS11-BP04 Wissensmanagement](#)

Zugehörige Dokumente:

- [Operative Kompetenz durch automatisierte Playbooks und Runbooks](#)
- [AWS Systems Manager: Working with runbooks](#)
- [Migrations-Playbook für große AWS-Migrationen – Aufgabe 4: Verbesserung Ihrer Migrations-Runbooks](#)
- [Use AWS Systems Manager Automation runbooks to resolve operational tasks](#)

Zugehörige Videos:

- [AWS re:Invent 2019: DIY-Leitfaden für Runbooks, Vorfälleberichte und Vorfällereaktion](#)
- [How to automate IT Operations on AWS | Amazon Web Services](#)
- [Integrate Scripts into AWS Systems Manager](#)

Zugehörige Beispiele:

- [Well-Architected Labs: Automatisieren von Vorgängen mit Playbooks und Runbooks](#)
- [AWS Blogbeitrag: Aufbau einer Cloud-Automatisierungspraxis für operative Exzellenz: Bewährte Methoden von AWS Managed Services](#)
- [AWS Systems Manager: Automation walkthroughs](#)
- [AWS Systems Manager: Restore a root volume from the latest snapshot runbook](#)
- [Building an AWS incident response runbook using Jupyter notebooks and CloudTrail Lake](#)
- [Gitlab – Runbooks](#)

- [Rubix – eine Python-Bibliothek für die Erstellung von Runbooks in Jupyter Notebooks](#)
- [Verwenden von Document Builder zum Erstellen eines benutzerdefinierten Runbooks](#)

Zugehörige Services:

- [AWS Systems Manager Automation](#)

OPS07-BP04 Verwenden von Playbooks zum Untersuchen von Problemen

Playbooks sind Schritt-für-Schritt-Anleitungen zur Untersuchung von Vorfällen. Wenn Vorfälle auftreten, werden Playbooks verwendet, um sie zu untersuchen, die Auswirkungen abzuschätzen und Ursachen zu identifizieren. Playbooks werden für verschiedene Szenarien eingesetzt, von fehlgeschlagenen Bereitstellungen bis hin zu Sicherheitsvorfällen. In vielen Fällen identifizieren Playbooks Ursachen, die dann mithilfe eines Runbooks beseitigt werden. Playbooks sind eine sehr wichtige Komponente der Vorfallreaktionspläne Ihrer Organisation.

Ein gutes Playbook weist einige zentrale Merkmale auf. Es leitet den Benutzer Schritt für Schritt durch den Erkennungsprozess. Welche Schritte sollten befolgt werden, um einen Vorfall zu diagnostizieren? Legen Sie im Playbook klar fest, ob bestimmte Tools oder erhöhte Berechtigungen benötigt werden. Ein wichtiger Teil ist ein Kommunikationsplan, um alle Stakeholder über den Status der Untersuchung zu informieren. Für den Fall, dass die eigentliche Ursache des Vorfalls nicht identifiziert werden kann, sollte das Playbook einen Eskalationsplan enthalten. Wenn die Ursache identifiziert wurde, sollte das Playbook auf ein Runbook verweisen, das beschreibt, wie die Ursache zu beheben ist. Playbooks sollten zentral gespeichert und regelmäßig gepflegt werden. Wenn Playbooks für bestimmte Warnungsmeldungen verwendet werden, sollte Ihr Team in den Warnungsmeldungen auf das Playbook verwiesen werden.

Im Zuge der Weiterentwicklung Ihrer Organisation sollten Sie Ihre Playbooks automatisieren. Beginnen Sie mit Playbooks für Vorfälle mit geringem Risikograd. Automatisieren Sie die Erkennungsschritte mit Skripten. Stellen Sie sicher, dass Sie über begleitende Runbooks für die Behebung typischer Ursachen verfügen.

Gewünschtes Ergebnis: Ihre Organisation verfügt über Playbooks für typische Vorfälle. Die Playbooks werden an einem zentralen Ort gespeichert und sind für Ihre Teammitglieder verfügbar. Playbooks werden häufig aktualisiert. Für alle bekannten Ursachen werden begleitende Runbooks erstellt.

Typische Anti-Muster:

- Es gibt kein Standardverfahren für die Untersuchung von Vorfällen.

- Teammitglieder verlassen sich auf ihr Gedächtnis oder allgemein vorhandenes Wissen, um eine fehlgeschlagene Bereitstellung zu beheben.
- Neue Teammitglieder lernen die Untersuchung von Problemen durch Ausprobieren.
- Es werden keine bewährten Methoden für die Untersuchung von Problemen zwischen Teams ausgetauscht.

Vorteile der Nutzung dieser bewährten Methode:

- Playbooks verbessern Ihre Fähigkeit zum Umgang mit Vorfällen.
- Verschiedene Teammitglieder können dasselbe Playbook verwenden, um Ursachen in konsistenter Weise zu ermitteln.
- Für bekannte Ursachen können Runbooks entwickelt werden, um die Wiederherstellungszeit zu verkürzen.
- Mit Playbooks können Teammitglieder schneller Beiträge leisten.
- Mit wiederholbaren Playbooks können Teams ihre Prozesse skalieren.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Wie Sie Ihre Playbooks aufbauen und verwenden, hängt vom Reifegrad Ihrer Organisation ab. Wenn Sie noch neu in der Cloud sind, erstellen Sie Playbooks in Textform in einem zentralen Dokumenten-Repository. Wenn sich Ihre Organisation weiterentwickelt, können Playbooks mit Skriptsprachen wie Python teilweise automatisiert werden. Diese Skripts können zur Beschleunigung der Untersuchung in einem Jupyter Notebook ausgeführt werden. Fortgeschrittene Organisationen haben vollständig automatisierte Playbooks für häufig auftretende Probleme, die dann mit Runbooks automatisch behoben werden.

Beginnen Sie die Arbeit an Ihren Playbooks mit der Auflistung typischer Vorfälle bei Ihren Workloads. Wählen Sie Playbooks zunächst für Vorfälle mit geringem Risiko, bei denen die Ursache eingegrenzt werden kann. Wenn Sie über Playbooks für einfachere Szenarien verfügen, gehen Sie zu Szenarien mit höheren Risiken oder zu Szenarien über, bei denen die Ursache nicht vollständig klar ist.

Ihre textbasierten Playbooks sollten mit zunehmender Entwicklung Ihrer Organisation automatisiert werden. Mit Services wie [AWS-Systems-Manager-Automatisierungen](#) können Textdateien in Automatisierungen transformiert werden. Diese Automatisierungen können dann für Ihre Workload

ausgeführt werden, um die Untersuchungen zu beschleunigen. Sie können in Reaktion auf Ereignisse aktiviert werden, wodurch sich der durchschnittliche Zeitaufwand für die Untersuchung und Behebung von Vorfällen reduziert.

Kunden können [AWS Systems Manager Incident Manager](#) zur Reaktion auf Vorfälle verwenden. Dieser Service bietet eine einzige Oberfläche für die Untersuchung von Vorfällen, die Information der Stakeholder über Untersuchung und Abhilfemaßnahmen und die Zusammenarbeit während des gesamten Vorgangs. Er verwendet AWS-Systems-Manager-Automatisierungen zur Beschleunigung von Untersuchung und Wiederherstellung.

Kundenbeispiel

Ein Produktionsvorfall hat Auswirkungen auf AnyCompany Retail. Der zuständige Techniker untersuchte das Problem mithilfe eines Playbooks. Im Zuge der einzelnen Schritte wurden die Stakeholder, die im Playbook festgelegt waren, auf dem Laufenden gehalten. Der Techniker ermittelte einen Race-Zustand in einem Backend-Service als Ursache für den Vorfall. Mithilfe eines Runbooks startete er den Service neu und brachte AnyCompany Retail so wieder online.

Implementierungsschritte

Wenn Sie noch kein Dokumenten-Repository besitzen, dann sollten Sie ein Versionskontroll-Repository für Ihre Playbook-Bibliothek erstellen. Sie können Ihre Playbooks mit Markdown erstellen, das mit den meisten Playbook-Automatisierungssystemen kompatibel ist. Wenn Sie neu beginnen, verwenden Sie die folgende Beispielvorlage für ein Playbook.

```
# Playbook Title
## Playbook Info
| Playbook ID | Description | Tools Used | Special Permissions | Playbook Author | Last
Updated | Escalation POC | Stakeholders | Communication Plan |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| RUN001 | What is this playbook for? What incident is it used for? | Tools |
Permissions | Your Name | 2022-09-21 | Escalation Name | Stakeholder Name | How will
updates be communicated during the investigation? |
## Steps
1. Step one
2. Step two
```

1. Wenn Sie noch kein Dokumenten-Repository oder -Wiki besitzen, sollten Sie in Ihrem Versionskontrollsystem ein neues Versionskontroll-Repository für Ihre Playbooks erstellen.

2. Identifizieren Sie ein typisches Problem, das eine Untersuchung erfordert. Dies sollte ein Szenario sein, bei dem die Ursache auf wenige Probleme eingegrenzt werden kann und das Risiko insgesamt niedrig ist.
3. Füllen Sie mithilfe der Markdown-Vorlage den Abschnitt Playbook-Name und die Felder unter Playbook-Informationen aus.
4. Geben Sie die Schritte zur Fehlerbehebung ein. Benennen Sie die zu treffenden Maßnahmen bzw. die zu untersuchenden Bereiche so klar wie möglich.
5. Geben Sie das Playbook einem Teammitglied zur Prüfung. Wenn darin etwas fehlt oder nicht klar ist, aktualisieren Sie das Playbook.
6. Veröffentlichen Sie Ihr Playbook in Ihrem Dokumenten-Repository und informieren Sie Ihr Team und alle Stakeholder darüber.
7. Diese Playbook-Bibliothek wächst mit der Zeit an. Sobald Sie mehrere Playbooks haben, beginnen Sie mithilfe von Tools wie AWS-Systems-Manager-Automatisierungen mit ihrer Automatisierung, um die Automatisierung und die Playbooks synchron zu halten.

Aufwand für den Implementierungsplan: Niedrig. Ihre Playbooks sollten an einem zentralen Ort gespeicherte Textdokumente sein. Ausgereiftere Organisationen gehen zu automatisierten Playbooks über.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS02-BP02 Prozesse und Verfahren haben feste Besitzer](#)
- [OPS07-BP03 Verwenden von Runbooks zur Durchführung von Verfahren](#)
- [OPS10-BP01 Verwenden eines Prozesses für die Bewältigung von Ereignissen, Vorfällen und Problemen](#)
- [OPS10-BP02 Implementieren eines Prozesses für jede Warnmeldung](#)
- [OPS11-BP04 Wissensmanagement](#)

Zugehörige Dokumente:

- [Operative Kompetenz durch automatisierte Playbooks und Runbooks](#)
- [AWS Systems Manager: Working with runbooks](#)
- [Use AWS Systems Manager Automation runbooks to resolve operational tasks](#)

Zugehörige Videos:

- [AWS re:Invent 2019: DIY-Leitfaden für Runbooks, Vorfälleberichte und Vorfällereaktion \(SEC318-R1\)](#)
- [AWS Systems Manager Incident Manager - AWS Virtual Workshops](#)
- [Integrate Scripts into AWS Systems Manager](#)

Zugehörige Beispiele:

- [AWS Customer Playbook Framework](#)
- [AWS Systems Manager: Automation walkthroughs](#)
- [Building an AWS incident response runbook using Jupyter notebooks and CloudTrail Lake](#)
- [Rubix – Eine Python-Bibliothek für die Erstellung von Runbooks in Jupyter Notebooks](#)
- [Verwenden von Document Builder zum Erstellen eines benutzerdefinierten Runbooks](#)

Zugehörige Services:

- [AWS Systems Manager Automation](#)
- [AWS Systems Manager Incident Manager](#)

OPS07-BP05 Treffen fundierter Entscheidungen für die Bereitstellung von Systemen und Änderungen

Sorgen Sie dafür, dass Prozesse für erfolgreiche und nicht erfolgreiche Änderungen an Ihrer Workload vorhanden sind. Eine Pre-mortem-Übung ist eine Übung, bei der ein Team einen Fehler simuliert, um Strategien zur Behebung zu entwickeln. Nutzen Sie diese „Pre-mortems“, um Fehlern vorzubeugen und legen Sie, wo erforderlich, entsprechende Abläufe fest. Bewerten Sie den Nutzen und die Risiken der Bereitstellung von Änderungen an Ihrer Workload. Überprüfen Sie, ob alle Änderungen mit der Governance übereinstimmen.

Gewünschtes Ergebnis:

- Sie treffen bei der Bereitstellung von Änderungen an Ihrer Workload fundierte Entscheidungen.
- Änderungen entsprechen der Governance.

Typische Anti-Muster:

- Sie stellen eine Änderung an Ihrer Workload bereit, ohne einen Prozess für die Verarbeitung einer fehlgeschlagenen Bereitstellung zu haben.
- Sie nehmen Änderungen an Ihrer Produktionsumgebung vor, die nicht mit den Governance-Anforderungen vereinbar sind.
- Sie stellen eine neue Version Ihrer Workload bereit, ohne eine Baseline für die Ressourcenauslastung zu erstellen.

Vorteile der Nutzung dieser bewährten Methode:

- Sie sind auf fehlgeschlagene Änderungen an Ihrer Workload vorbereitet.
- Änderungen an Ihrer Workload sind konform mit den Governance-Richtlinien.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Niedrig

Implementierungsleitfaden

Verwenden Sie Pre-mortem-Übungen, um Prozesse für fehlgeschlagene Änderungen zu entwickeln. Dokumentieren Sie Ihre Prozesse für fehlgeschlagene Änderungen. Stellen Sie sicher, dass alle Änderungen mit der Governance übereinstimmen. Evaluieren Sie die Vorteile und Risiken der Bereitstellung von Änderungen an Ihrer Workload.

Kundenbeispiel

AnyCompany Retail führt regelmäßig Pre-Mortems durch, um die Prozesse für fehlgeschlagene Änderungen zu validieren. Die Prozesse werden in einem gemeinsamen Wiki dokumentiert und regelmäßig aktualisiert. Alle Änderungen entsprechen den Governance-Anforderungen.

Implementierungsschritte

1. Treffen Sie fundierte Entscheidungen, wenn Sie Änderungen an Ihrer Workload bereitstellen. Legen Sie Kriterien für eine erfolgreiche Bereitstellung fest und überprüfen Sie diese. Entwickeln Sie Szenarien oder Kriterien, die ein Rollback einer Änderung auslösen würden. Wägen Sie den Nutzen der Bereitstellung von Änderungen gegen die Risiken einer fehlgeschlagenen Änderung ab.
2. Überprüfen Sie, ob alle Änderungen mit den Governance-Richtlinien übereinstimmen.
3. Planen Sie anhand von Pre-Mortems fehlgeschlagene Änderungen und dokumentieren Sie Strategien zur Schadensbegrenzung. Führen Sie eine Table-Top-Übung durch, um eine fehlgeschlagene Änderung zu modellieren und Rollback-Verfahren zu validieren.

Aufwand für den Implementierungsplan: Mittel. Die Einführung von Pre-Mortems erfordert die Koordination und den Einsatz aller Stakeholder in Ihrer gesamten Organisation

Ressourcen

Zugehörige bewährte Methoden:

- [OPS01-BP03 Bewertung der Governance-Anforderungen](#) – Governance-Anforderungen sind ein Schlüssel bei der Entscheidung zur Bereitstellung einer Änderung.
- [OPS06-BP01 Plan für erfolglose Änderungen](#) – Erstellen Sie Pläne zur Eindämmung einer fehlgeschlagenen Bereitstellung und verwenden Sie Pre-Mortems, um diese zu validieren.
- [OPS06-BP02 Testbereitstellungen](#) – Jede Softwareänderung sollte vor der Bereitstellung ordnungsgemäß getestet werden, um Fehler in der Produktion zu reduzieren.
- [OPS07-BP01 Sicherstellen des Know-hows der Mitarbeiter](#) – Ausreichend trainierte Mitarbeiter zur Unterstützung der Workload sind unerlässlich, um eine fundierte Entscheidung über die Bereitstellung einer Systemänderung zu treffen.

Zugehörige Dokumente:

- [Amazon Web Services: Risiko und Compliance](#)
- [AWS-Modell der übergreifenden Verantwortlichkeit](#)
- [Governance in der AWS Cloud: Die richtige Balance zwischen Agilität und Sicherheit](#)

OPS07-BP06 Erstellen von Supportplänen für Produktions-Workloads

Aktivieren Sie Support für sämtliche Software und Services, auf denen Ihre Produktions-Workload basiert. Wählen Sie ein geeignetes Support-Level für Ihre Servicelevel-Anforderungen in der Produktion. Supportpläne für diese Abhängigkeiten sind wichtig für den Fall von Serviceunterbrechungen oder Softwareproblemen. Dokumentieren Sie Supportpläne sowie die Verfahren zur Anfrage nach Support bei allen Service- und Softwareanbietern. Implementieren Sie Mechanismen zur Prüfung, ob Support-Kontaktpunkte stets aktuell sind.

Gewünschtes Ergebnis:

- Implementieren Sie Supportpläne für Software und Services, auf denen Ihre Produktions-Workloads basieren.
- Wählen Sie einen geeigneten Supportplan auf der Grundlage Ihrer Service-Level-Anforderungen.

- Dokumentieren Sie die Supportpläne, die Supportlevels und die Vorgehensweise bei Supportanfragen.

Typische Anti-Muster:

- Sie haben keinen Supportplan für einen kritischen Softwareanbieter. Dies beeinflusst Ihre Workload und Sie haben keine Möglichkeit, schnell einen Fix oder rechtzeitige Updates von dem Anbieter zu erhalten.
- Ein Entwickler, der der primäre Ansprechpartner bei einem Softwareanbieter war, hat das Unternehmen verlassen. Sie können den Support des Anbieters nicht direkt erreichen. Sie müssen Zeit aufwenden, um sich durch generische Kontaktsysteme zu arbeiten, was die Reaktionszeiten verlängert.
- Bei einem Softwareanbieter ereignet sich ein Produktionsausfall. Es gibt keine Dokumentation dazu, wie ein Supportfall einzureichen ist.

Vorteile der Nutzung dieser bewährten Methode:

- Mit dem richtigen Supportlevel können Sie schnell eine Reaktion erhalten, die dem Service-Level entspricht.
- Als Kunde mit Support stehen Ihnen bei Produktionsproblemen Eskalationsmöglichkeiten zur Verfügung.
- Software- und Serviceanbieter können Ihnen bei Vorfällen Unterstützung bei der Fehlerbehebung bieten.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Niedrig

Implementierungsleitfaden

Aktivieren Sie die Supportpläne für sämtliche Software- und Serviceanbieter, von denen Ihre Produktions-Workload abhängt. Richten Sie geeignete Supportpläne ein, um Service-Level einhalten zu können. Für AWS-Kunden bedeutet dies die Aktivierung von AWS Business Support oder einer höheren Stufe für alle Konten mit Produktions-Workloads. Treffen Sie sich regelmäßig mit Supportanbietern, um Neues zu Supportangeboten, -prozessen und -ansprechpartnern zu erfahren. Dokumentieren Sie das Supportverfahren bei Software- und Serviceanbietern, einschließlich der Eskalationsmöglichkeiten bei Ausfällen. Implementieren Sie Mechanismen, um die Supportkontakte stets auf aktuellem Stand zu halten.

Kundenbeispiel

Bei AnyCompany Retail gibt es für alle kommerziellen Software- und Service-Abhängigkeiten Supportpläne. Beispielsweise hat das Unternehmen AWS Enterprise Support für alle Konten mit Produktions-Workloads. Jeder Entwickler kann bei einem Problem einen Supportfall auslösen. Es gibt eine Wiki-Seite mit Informationen zum Verfahren bei Supportanfragen, zu den Ansprechpartnern und zu bewährten Methoden dafür.

Implementierungsschritte

1. Arbeiten Sie mit den Stakeholdern in Ihrer Organisation, um Software- und Serviceanbieter zu identifizieren, von denen Ihre Workload abhängt. Dokumentieren Sie diese Abhängigkeiten.
2. Legen Sie die Service-Level-Anforderungen für Ihre Workload fest. Wählen Sie einen Supportplan, der dazu passt.
3. Richten Sie für kommerzielle Software und Services einen Supportplan bei den Anbietern ein.
 - a. Ein Abonnement von AWS Business Support oder höher für alle Produktionskonten bietet schnellere Reaktionszeiten von AWS Support und wird dringend empfohlen. Wenn Sie keinen Premium-Support haben, benötigen Sie einen Aktionsplan für den Umgang mit Problemen, bei denen Hilfe von AWS Support erforderlich ist. AWS Support stellt Ihnen verschiedenste Tools und Technologien, Fachpersonal und Programme zur Verfügung, die Sie proaktiv bei der Performance-Optimierung, Kostensenkung und schnelleren Entwicklung neuer Innovationen unterstützen. Darüber hinaus bietet AWS Business Support zusätzliche Vorteile, darunter API-Zugriff auf AWS Trusted Advisor und AWS Health für die programmgesteuerte Integration mit Ihren Systemen sowie weitere Zugriffsmethoden wie die AWS-Managementkonsole und Amazon EventBridge-Kanäle.
4. Dokumentieren Sie den Supportplan in Ihrem Wissensmanagement-Tool. Berücksichtigen Sie dabei, wie eine Supportanfrage durchgeführt wird, wer in einem solchen Fall zu benachrichtigen ist und wie Vorfälle eskaliert werden können. Ein Wiki ist ein gutes Hilfsmittel, das allen Beteiligten ermöglicht, erforderliche Aktualisierungen der Dokumentation vorzunehmen, wenn ihnen Änderungen bei Supportprozessen oder Ansprechpartnern bekannt werden.

Aufwand für den Implementierungsplan: Niedrig. Die meisten Software- und Serviceanbieter bieten Opt-in-Supportpläne an. Durch die Dokumentation und die Weitergabe bewährter Supportmethoden in Ihrem Wissensmanagementsystem können Sie sicherstellen, dass Ihr Team weiß, was bei einem Produktionsproblem zu tun ist.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS02-BP02 Prozesse und Verfahren haben feste Besitzer](#)

Zugehörige Dokumente:

- [AWS Support-Pläne](#)

Zugehörige Services:

- [AWS Business Support](#)
- [AWS Enterprise Support](#)

Betrieb

Fragen

- [OPS 8. Wie nutzen Sie die Beobachtbarkeit von Workloads in Ihrer Organisation?](#)
- [OPS 9. Wie können Sie den Zustand Ihrer Operationen beurteilen?](#)
- [OPS 10. Wie bewältigen Sie Workload- und operationsspezifische Ereignisse?](#)

OPS 8. Wie nutzen Sie die Beobachtbarkeit von Workloads in Ihrer Organisation?

Sorgen Sie für einen optimalen Zustand der Workload, indem Sie die Beobachtbarkeit nutzen. Nutzen Sie relevante Metriken, Protokolle und Ablaufverfolgungen, um sich einen umfassenden Überblick über die Leistung Ihrer Workload zu verschaffen und Probleme effizient zu beheben.

Best Practices

- [OPS08-BP01 Analysieren Sie Workload-Metriken](#)
- [OPS08-BP02 Analysieren Sie Workload-Protokolle](#)
- [OPS08-BP03 Analysieren Sie Workload-Traces](#)
- [OPS08-BP04 Erstellen umsetzbarer Warnmeldungen](#)
- [OPS08-BP05 Erstellen von Dashboards](#)

OPS08-BP01 Analysieren Sie Workload-Metriken

Analysieren Sie nach der Implementierung der Anwendungstelemetrie regelmäßig die gesammelten Metriken. Latenz, Anfragen, Fehler und Kapazität (oder Kontingente) liefern zwar Erkenntnisse zur Systemleistung, es ist jedoch wichtig, die Überprüfung der Metriken zu Geschäftsergebnissen zu priorisieren. Dadurch wird sichergestellt, dass Sie datengestützte Entscheidungen treffen, die auf Ihre Geschäftsziele abgestimmt sind.

Gewünschtes Ergebnis: Präzise Erkenntnisse zur Workload-Leistung, die als Grundlage für datengestützte Entscheidungen dienen und die Abstimmung mit den Geschäftszielen sicherstellen.

Typische Anti-Muster:

- Isolierte Analyse von Metriken, ohne deren Auswirkungen auf die Geschäftsergebnisse zu berücksichtigen.
- Übermäßiges Vertrauen in technische Metriken, während Geschäftsmetriken ignoriert werden.
- Seltene Überprüfung von Metriken, Entscheidungsmöglichkeiten in Echtzeit werden verpasst.

Vorteile der Nutzung dieser bewährten Methode:

- Verbessertes Verständnis des Zusammenhangs zwischen technischer Leistung und Geschäftsergebnissen.
- Verbesserter Entscheidungsprozess auf der Grundlage von Echtzeitdaten.
- Proaktive Identifizierung und Minderung von Problemen, bevor sie sich auf die Geschäftsergebnisse auswirken.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Nutzen Sie Tools wie Amazon CloudWatch , um metrische Analysen durchzuführen. AWS Dienste wie CloudWatch Anomalieerkennung und Amazon DevOps Guru können zur Erkennung von Anomalien verwendet werden, insbesondere wenn statische Schwellenwerte unbekannt sind oder wenn Verhaltensmuster besser für die Erkennung von Anomalien geeignet sind.

Implementierungsschritte

1. Analysieren und überprüfen: Überprüfen Sie regelmäßig Ihre Workload-Metriken und werten Sie sie aus.

- a. Priorisieren Sie Metriken zu Geschäftsergebnissen gegenüber rein technischen.
 - b. Machen Sie sich mit der Bedeutung von Spitzen, Rückgängen oder Mustern in Ihren Daten vertraut.
2. Nutzen Sie Amazon CloudWatch: Verwenden Sie Amazon CloudWatch für eine zentrale Ansicht und detaillierte Analysen.
- a. Konfigurieren Sie CloudWatch Dashboards, um Ihre Kennzahlen zu visualisieren und sie im Laufe der Zeit zu vergleichen.
 - b. Verwenden Sie [Perzentile](#), CloudWatch um sich einen klaren Überblick über die Verteilung der Metriken zu verschaffen. Dies kann dazu beitragen, Ausreißer zu definieren SLAs und zu verstehen.
 - c. Richten Sie die [Erkennung von CloudWatch Anomalien](#) ein, um ungewöhnliche Muster zu identifizieren, ohne sich auf statische Schwellenwerte verlassen zu müssen.
 - d. Implementieren Sie [CloudWatch kontenübergreifende Beobachtbarkeit](#), um Anwendungen zu überwachen und Fehler zu beheben, die sich über mehrere Konten innerhalb einer Region erstrecken.
 - e. Verwenden Sie [CloudWatch Metric Insights](#), um Kennzahlen konten- und regionsübergreifend abzufragen und zu analysieren und Trends und Anomalien zu identifizieren.
 - f. Wenden Sie [CloudWatch Metric Math](#) an, um Ihre Kennzahlen zu transformieren, zu aggregieren oder zu berechnen, um tiefere Einblicke zu erhalten.
3. Nutzen Sie Amazon DevOps Guru: Integrieren Sie [Amazon DevOps Guru](#) für die durch maschinelles Lernen erweiterte Anomalieerkennung, um frühe Anzeichen von Betriebsproblemen Ihrer serverlosen Anwendungen zu erkennen und diese zu beheben, bevor sie sich auf Ihre Kunden auswirken.
4. Optimieren Sie auf der Grundlage von Erkenntnissen: Treffen Sie fundierte Entscheidungen auf der Grundlage Ihrer Metrikanalyse, um Ihre Workloads anzupassen und zu verbessern.

Aufwand für den Implementierungsplan: Mittel

Ressourcen

Zugehörige bewährte Methoden:

- [OPS04-BP01 Identifizieren Sie die wichtigsten Leistungsindikatoren](#)
- [OPS04-BP02 Implementieren Sie Anwendungstelemetrie](#)

Zugehörige Dokumente:

- [The Wheel Blog – Die Bedeutung der kontinuierlichen Überprüfung von Metriken](#)
- [Perzentile sind wichtig](#)
- [Verwenden AWS Cost Anomaly Detection](#)
- [CloudWatch kontenübergreifende Beobachtbarkeit](#)
- [Fragen Sie Ihre Metriken mit Metrics Insights ab CloudWatch](#)

Zugehörige Videos:

- [Kontoübergreifende Observability in Amazon aktivieren CloudWatch](#)
- [Einführung in Amazon DevOps Guru](#)
- [Analysieren Sie kontinuierlich Metriken mit AWS Cost Anomaly Detection](#)

Zugehörige Beispiele:

- [Workshop zur Beobachtbarkeit](#)
- [AIOpsMit Amazon DevOps Guru Einblicke in den Betrieb gewinnen](#)

OPS08-BP02 Analysieren Sie Workload-Protokolle

Die regelmäßige Analyse von Workload-Protokollen ist unerlässlich, um ein tieferes Verständnis der operativen Aspekte Ihrer Anwendung zu erlangen. Durch effizientes Durchsuchen, Visualisieren und Interpretieren von Protokolldaten können Sie die Leistung und Sicherheit von Anwendungen kontinuierlich optimieren.

Gewünschtes Ergebnis: Umfassende Erkenntnisse zum Anwendungsverhalten und zu Operationen, die aus einer gründlichen Protokollanalyse gewonnen wurden und für eine proaktive Problemerkennung und -behebung sorgen.

Typische Anti-Muster:

- Die Analyse von Protokollen vernachlässigen, bis ein kritisches Problem auftritt.
- Die Suite verfügbarer Tools für die Protokollanalyse nicht nutzen und wichtige Erkenntnisse verpassen.

- Alleiniges Vertrauen auf die manuelle Überprüfung von Protokollen, ohne Automatisierungs- und Abfragefunktionen zu nutzen.

Vorteile der Nutzung dieser bewährten Methode:

- Proaktive Identifizierung von operativen Engpässen, Sicherheitsbedrohungen und anderen potenziellen Problemen.
- Effiziente Nutzung von Protokolldaten für die kontinuierliche Anwendungsoptimierung.
- Verbessertes Verständnis des Anwendungsverhaltens, Unterstützung beim Debuggen und bei der Problembehandlung.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

[Amazon CloudWatch Logs](#) ist ein leistungsstarkes Tool für die Protokollanalyse. Integrierte Funktionen wie CloudWatch Logs Insights und Contributor Insights machen das Ableiten aussagekräftiger Informationen aus Protokollen intuitiv und effizient.

Implementierungsschritte

1. CloudWatch Protokolle einrichten: Konfigurieren Sie Anwendungen und Dienste so, dass sie Protokolle an CloudWatch Logs senden.
2. Verwenden Sie die Erkennung von Protokollanomalien: Verwenden Sie die [Anomalieerkennung von Amazon CloudWatch Logs](#), um ungewöhnliche Protokollmuster automatisch zu identifizieren und darauf hinzuweisen. Mit diesem Tool können Sie Anomalien in Ihren Protokollen proaktiv verwalten und potenzielle Probleme frühzeitig erkennen.
3. CloudWatch Logs Insights einrichten: Verwenden Sie [CloudWatch Logs Insights](#), um Ihre Protokolldaten interaktiv zu suchen und zu analysieren.
 - a. Erstellen Sie Abfragen, um Muster zu extrahieren, Protokolldaten zu visualisieren und umsetzbare Erkenntnisse abzuleiten.
 - b. Verwenden Sie die [CloudWatch Logs Insights-Musteranalyse](#), um häufige Protokollmuster zu analysieren und zu visualisieren. Dieses Feature hilft Ihnen, allgemeine Betriebstrends und potenzielle Ausreißer in Ihren Protokolldaten nachzuvollziehen.
 - c. Verwenden Sie [CloudWatch Logs compare \(diff\)](#), um eine Differenzanalyse zwischen verschiedenen Zeiträumen oder zwischen verschiedenen Protokollgruppen durchzuführen.

Verwenden Sie diese Funktion, um Änderungen zu lokalisieren und deren Auswirkungen auf die Leistung oder das Verhalten Ihres Systems zu bewerten.

4. Überwachen Sie Protokolle in Echtzeit mit Live Tail: Verwenden Sie [Amazon CloudWatch Logs Live Tail](#), um Protokolldaten in Echtzeit anzuzeigen. Sie können die Betriebsaktivitäten Ihrer Anwendung in Echtzeit aktiv überwachen, um sich einen unmittelbaren Einblick in die Systemleistung und potenzielle Probleme zu verschaffen.
5. Nutzen Sie Contributor Insights: Verwenden Sie [CloudWatchContributor Insights](#), um Top-Talker in Dimensionen mit hoher Kardinalität wie IP-Adressen oder Benutzeragenten zu identifizieren.
6. Implementieren Sie Metrikfilter für CloudWatch Logs: Konfigurieren Sie Metrikfilter für [CloudWatch Logs, um Protokolldaten in umsetzbare Metriken](#) umzuwandeln. Auf diese Weise können Sie Alarmer einstellen oder Muster näher analysieren.
7. Implementieren Sie [CloudWatchkontenübergreifende Beobachtbarkeit](#): Überwachen Sie Anwendungen, die sich über mehrere Konten innerhalb einer Region erstrecken, und beheben Sie Fehler.
8. Regelmäßige Überprüfung und Verfeinerung: Überprüfen Sie regelmäßig Ihre Protokollanalysestrategien, um alle relevanten Informationen zu erfassen und die Anwendungsleistung kontinuierlich zu optimieren.

Aufwand für den Implementierungsplan: Mittel

Ressourcen

Zugehörige bewährte Methoden:

- [OPS04-BP01 Identifizieren Sie die wichtigsten Leistungsindikatoren](#)
- [OPS04-BP02 Implementieren Sie Anwendungstelemetrie](#)
- [OPS08-BP01 Analysieren Sie Workload-Metriken](#)

Zugehörige Dokumente:

- [Analysieren von Protokolldaten mit CloudWatch Logs Insights](#)
- [CloudWatch Contributor Insights verwenden](#)
- [CloudWatch Log-Metrikfilter erstellen und verwalten](#)

Zugehörige Videos:

- [Analysieren Sie Protokolldaten mit CloudWatch Logs Insights](#)
- [Verwenden Sie CloudWatch Contributor Insights, um Daten mit hoher Kardinalität zu analysieren](#)

Zugehörige Beispiele:

- [CloudWatch Protokolliert Beispielabfragen](#)
- [Workshop zur Beobachtbarkeit](#)

OPS08-BP03 Analysieren Sie Workload-Traces

Die Analyse von Trace-Daten ist entscheidend, wenn es darum geht, einen umfassenden Überblick über den Betriebsverlauf einer Anwendung zu erhalten. Durch die Visualisierung und das Verständnis der Interaktionen zwischen verschiedenen Komponenten können die Leistung optimiert, Engpässe identifiziert und das Benutzererlebnis verbessert werden.

Gewünschtes Ergebnis: Sie verschaffen sich einen klaren Überblick über die verteilten Abläufe Ihrer Anwendung und erzielen dadurch eine schnellere Problemlösung und eine verbesserte Benutzererfahrung.

Typische Anti-Muster:

- Trace-Daten werden übersehen und man verlässt sich ausschließlich auf Protokolle und Metriken.
- Trace-Daten werden nicht mit zugehörigen Protokollen in Zusammenhang gebracht.
- Aus Traces abgeleitete Metriken wie Latenz und Fehlerraten werden ignoriert.

Vorteile der Nutzung dieser bewährten Methode:

- Verbessern Sie die Problembehandlung und reduzieren Sie die durchschnittliche Zeit bis zur Problemlösung (). MTTR
- Sie gewinnen Erkenntnisse über Abhängigkeiten und deren Auswirkungen.
- Sie können Leistungsprobleme rasch identifizieren und beheben.
- Sie nutzen von aus Trace abgeleitete Metriken für fundierte Entscheidungen.
- Sie erzielen ein besseres Benutzererlebnis durch optimierte Komponenteninteraktionen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

[AWS X-Ray](#) bietet eine umfassende Suite für die Analyse von Trace-Daten, die einen ganzheitlichen Überblick über Serviceinteraktionen, die Überwachung von Benutzeraktivitäten und die Erkennung von Leistungsproblemen bietet. Funktionen wie ServiceLens X-Ray Insights, X-Ray Analytics und Amazon DevOps Guru erweitern die Tiefe verwertbarer Erkenntnisse, die aus Trace-Daten gewonnen werden.

Implementierungsschritte

Die folgenden Schritte bieten einen strukturierten Ansatz zur effektiven Implementierung der Analyse von Spurendaten mithilfe von AWS Services:

1. Integrieren AWS X-Ray: Stellen Sie sicher, dass X-Ray in Ihre Anwendungen integriert ist, um Trace-Daten zu erfassen.
2. Analyse von X-Ray-Metriken: Untersuchen Sie anhand von X-Ray-Traces abgeleitete Metriken wie Latenz, Anfrageraten, Fehlerraten und Antwortzeitverteilungen mithilfe der [Service-Übersicht](#), um den Status der Anwendung zu überwachen.
3. Verwendung ServiceLens: Nutzen Sie die [ServiceLensKarte](#), um die Sichtbarkeit Ihrer Dienste und Anwendungen zu verbessern. Dies ermöglicht eine integrierte Anzeige von Traces, Metriken, Protokollen, Alarmen und anderen Statusinformationen.
4. Aktivieren von X-Ray-Insights:
 - a. Aktivieren Sie [X-Ray-Insights](#) zur automatisierten Erkennung von Anomalien in Traces.
 - b. Untersuchen Sie Erkenntnisse, um Muster zu identifizieren und die Ursachen zu ermitteln, z. B. erhöhte Fehlerraten oder Latenzen.
 - c. Eine chronologische Analyse der erkannten Probleme finden Sie in der Insights-Timeline.
5. Verwenden von X-Ray Analytics: [X-Ray Analytics](#) ermöglicht es Ihnen, Trace-Daten gründlich zu untersuchen, Muster zu lokalisieren und Erkenntnisse zu gewinnen.
6. Verwenden von Gruppen in X-Ray: Erstellen Sie Gruppen in X-Ray, um Traces nach Kriterien wie hoher Latenz zu filtern und so eine gezieltere Analyse zu ermöglichen.
7. Integrieren Sie Amazon DevOps Guru: Nutzen Sie [Amazon DevOps Guru](#), um von Modellen für maschinelles Lernen zu profitieren, mit denen betriebliche Anomalien in Spuren lokalisiert werden können.
8. Verwenden Sie CloudWatch Synthetics: Verwenden Sie [CloudWatchSynthetics](#), um Kanarien für die kontinuierliche Überwachung Ihrer Endpunkte und Workflows zu erstellen. Sie können

diese Canaries in X-Ray integrieren, um Trace-Daten für eine eingehende Analyse der getesteten Anwendungen bereitzustellen.

9. Verwenden Sie Real User Monitoring (RUM): Mit [AWS X-Ray und](#) können Sie den Anforderungspfad analysieren und debuggen CloudWatch RUM, angefangen bei den Endbenutzern Ihrer Anwendung bis hin zu nachgeschalteten Managed Services. AWS Auf diese Weise können Sie Latenzrends und Fehler identifizieren, die sich auf Ihre Endbenutzer auswirken.
- 10 Korrelieren von Daten mit Protokollen:: Bringen Sie [Trace-Daten mit zugehörigen Protokollen](#) innerhalb der X-Ray-Trace-Ansicht in Zusammenhang, um eine detaillierte Perspektive auf das Anwendungsverhalten zu erhalten. Auf diese Weise können Sie Protokollereignisse anzeigen, die direkt mit verfolgten Transaktionen verknüpft sind.
- 11 Implementieren Sie [CloudWatchkontenübergreifende Beobachtbarkeit](#): Überwachen Sie Anwendungen, die sich über mehrere Konten innerhalb einer Region erstrecken, und beheben Sie Fehler.

Aufwand für den Implementierungsplan: Mittel

Ressourcen

Zugehörige bewährte Methoden:

- [OPS08-BP01 Analysieren Sie Workload-Metriken](#)
- [OPS08-BP02 Analysieren Sie Workload-Protokolle](#)

Zugehörige Dokumente:

- [Verwendung ServiceLens zur Überwachung des Anwendungszustands](#)
- [Erkunden von Trace-Daten mit X-Ray Analytics](#)
- [Mit X-Ray-Insights Anomalien in Traces erkennen](#)
- [Kontinuierliche Überwachung mit CloudWatch Synthetics](#)

Zugehörige Videos:

- [Analysieren und Debuggen von Anwendungen mit Amazon CloudWatch Synthetics & AWS X-Ray](#)
- [Nutzung von AWS X-Ray -Insights](#)

Zugehörige Beispiele:

- [Workshop zur Beobachtbarkeit](#)
- [Implementierung von X-Ray mit AWS Lambda](#)
- [CloudWatchSynthetics Canary Schablonen](#)

OPS08-BP04 Erstellen umsetzbarer Warnmeldungen

Es ist entscheidend, Abweichungen im Verhalten Ihrer Anwendung umgehend zu erkennen und darauf zu reagieren. Besonders wichtig ist es, zu erkennen, wann die auf den wichtigsten Leistungsindikatoren (KPIs) basierenden Ergebnisse gefährdet sind oder unerwartete Anomalien auftreten. Wenn Sie Warnmeldungen auf KPIs basieren, stellen Sie dadurch sicher, dass die Signale, die Sie erhalten, direkt mit geschäftlichen oder betrieblichen Auswirkungen verknüpft sind. Der Ansatz mit umsetzbaren Warnmeldungen fördert proaktive Reaktionen und trägt zur Aufrechterhaltung der Systemleistung und Zuverlässigkeit bei.

Gewünschtes Ergebnis: Sie erhalten rechtzeitig relevante und umsetzbare Warnmeldungen, um potenzielle Probleme schnell zu erkennen und zu beheben, insbesondere wenn die KPI-Ergebnisse gefährdet sind.

Typische Anti-Muster:

- Es werden zu viele unkritische Warnmeldungen eingerichtet, was zu einer Alarmmüdigkeit führt.
- Warnmeldungen werden nicht anhand von KPIs priorisiert, was es schwierig macht, die geschäftlichen Auswirkungen von Problemen zu verstehen.
- Die eigentlichen Ursachen werden vernachlässigt, was zu wiederholten Warnmeldungen für dasselbe Problem führt.

Vorteile der Nutzung dieser bewährten Methode:

- Geringere Alarmermüdung durch Fokussierung auf umsetzbare und relevante Warnmeldungen.
- Verbesserte Systemverfügbarkeit und -zuverlässigkeit durch proaktive Problemerkennung und -behebung.
- Verbesserte Teamzusammenarbeit und schnellere Problemlösung durch die Integration in übliche Alarmierungs- und Kommunikationstools.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Um einen effektiven Warnmechanismus zu schaffen, ist es wichtig, Metriken, Protokolle und Trace-Daten zu verwenden, die darauf hinweisen, wenn auf KPIs basierende Ergebnisse gefährdet sind oder Anomalien erkannt werden.

Implementierungsschritte

1. Festlegen der wichtigsten Leistungskennzahlen (KPIs): Identifizieren Sie die KPIs Ihrer Anwendung. Warnmeldungen sollten mit diesen KPIs verknüpft werden, damit sie die Auswirkungen auf das Unternehmen genau widerspiegeln.
2. Implementierung der Erkennung von Anomalien:
 - Verwenden von Amazon CloudWatch-Anomalieerkennung: Richten Sie die [Amazon CloudWatch-Anomalieerkennung](#) so ein, dass ungewöhnliche Muster automatisch erkannt werden. So werden nur Warnmeldungen für echte Anomalien generiert.
 - Nutzung von AWS X-Ray-Insights:
 - a. Richten Sie [X-Ray-Insights](#) ein, um Anomalien in Trace-Daten zu erkennen.
 - b. Konfigurieren Sie [Benachrichtigungen für X-Ray-Insights](#), um bei erkannten Problemen gewarnt zu werden.
 - Integration mit Amazon DevOps Guru:
 - a. Nutzen Sie [Amazon DevOps Guru](#) für seine Machine-Learning-Funktionen für die Erkennung betrieblicher Anomalien anhand vorhandener Daten.
 - b. Navigieren Sie zu den [Benachrichtigungseinstellungen](#) in DevOps Guru, um Warnmeldungen über Anomalien einzurichten.
3. Implementieren umsetzbarer Warnmeldungen: Entwerfen Sie Warnmeldungen, die angemessene Informationen für sofortige Maßnahmen liefern.
 1. Überwachen Sie [AWS Health-Ereignisse mit Amazon-EventBridge-Regeln](#) oder integrieren Sie sie programmatisch mit der AWS Health API, um Aktionen zu automatisieren, wenn Sie AWS Health-Ereignisse empfangen. Dies können allgemeine Aktionen sein, z. B. das Senden aller geplanten Lebenszyklus-Ereignisnachrichten an eine Chat-Oberfläche, oder spezifische Aktionen, wie das Initiieren eines Workflows in einem IT-Servicemanagement-Tool.
4. Verringern der Alarmmüdigkeit: Minimieren Sie die Zahl der Warnmeldungen, die nicht kritisch sind. Wenn Teams mit zahllosen unbedeutenden Warnmeldungen überfordert werden, können sie den Überblick über kritische Probleme verlieren, was die Gesamteffektivität des Warnmechanismus beeinträchtigt.

5. Einrichten zusammengesetzter Alarme: Verwenden Sie [zusammengesetzte Alarme in Amazon CloudWatch](#), um mehrere Alarme zu konsolidieren.
6. Integration mit Warnmeldungs-Tools: Integrieren Sie Tools wie [Ops Genie](#) und [PagerDuty](#).
7. Einbinden von Amazon Q Developer in Chat-Anwendungen: Integration von [Amazon Q Developer in Chat-Anwendungen](#) zur Weiterleitung von Warnungen an Amazon Chime, Microsoft Teams und Slack.
8. Auf Protokollen basierende Warnungen: Verwenden Sie [metrische Protokollfilter](#) in CloudWatch, um Alarme auf der Grundlage bestimmter Protokollereignisse zu erstellen.
9. Überprüfen und wiederholen: Überprüfen und verfeinern Sie die Warnkonfigurationen regelmäßig.

Aufwand für den Implementierungsplan: Mittel

Ressourcen

Zugehörige bewährte Methoden:

- [OPS04-BP01 Identifizieren Sie die wichtigsten Leistungsindikatoren](#)
- [OPS04-BP02 Implementieren Sie Anwendungstelemetrie](#)
- [OPS04-BP03 Implementieren Sie Benutzererlebnis-Telemetrie](#)
- [OPS04-BP04 Implementieren einer Abhängigkeitstelemetrie](#)
- [OPS04-BP05 Implementieren Sie verteiltes Tracing](#)
- [OPS08-BP01 Analysieren Sie Workload-Metriken](#)
- [OPS08-BP02 Analysieren Sie Workload-Protokolle](#)
- [OPS08-BP03 Analysieren Sie Workload-Traces](#)

Zugehörige Dokumente:

- [Verwenden von Amazon-CloudWatch-Alarmen](#)
- [Erstellen eines zusammengesetzten Alarms](#)
- [Erstellen eines CloudWatch-Alarms basierend auf Anomalieerkennung](#)
- [DevOps Guru Notifications](#)
- [X-ray insights notifications](#)
- [Überwachung, Betrieb und Fehlerbehebung Ihrer AWS-Ressourcen mit interaktiven ChatOps](#)
- [Amazon CloudWatch Integration Guide | PagerDuty](#)

- [Integrate Opsgenie with Amazon CloudWatch](#)

Zugehörige Videos:

- [Create Composite Alarms in Amazon CloudWatch](#)
- [Überblick über Amazon Q Developer in Chat-Anwendungen](#)
- [AWS On Air ft. Mutative Befehle in Amazon Q Developer in Chat-Anwendungen](#)

Zugehörige Beispiele:

- [Alarme, Vorfallmanagement und Problembehebung in der Cloud mit Amazon CloudWatch](#)
- [Tutorial: Erstellen einer Amazon-EventBridge-Regel, die Benachrichtigungen an Amazon Q Developer in Chat-Anwendungen sendet](#)
- [Workshop zur Beobachtbarkeit](#)

OPS08-BP05 Erstellen von Dashboards

Dashboards sind die anwenderorientierte Sicht auf die Telemetriedaten Ihrer Workloads. Sie stellen zwar eine wichtige visuelle Schnittstelle dar, sollten aber nicht als Ersatz, sondern als Ergänzung für Warnmechanismen dienen. Wenn sie sorgfältig zusammengestellt werden, liefern sie nicht nur schnelle Erkenntnisse zum Status und zur Leistung des Systems, sondern bieten Stakeholdern auch Echtzeitinformationen über Geschäftsergebnisse und die Auswirkungen von Problemen.

Gewünschtes Ergebnis:

Klare, umsetzbare Erkenntnisse zur System- und Geschäftsstabilität mithilfe visueller Darstellungen.

Typische Anti-Muster:

- Überkomplizierte Dashboards mit zu vielen Metriken.
- Sich auf Dashboards verlassen, ohne Warnmeldungen zur Erkennung von Anomalien zu nutzen.
- Fehlende Aktualisierung der Dashboards im Laufe des Workload-Fortschritts.

Vorteile dieser bewährten Methode:

- Sofortiger Einblick in wichtige Systemmetriken und KPIs.
- Verbesserte Kommunikation und mehr Verständnis unter den Stakeholdern.

- Rasche Erkenntnisse zu den Auswirkungen operativer Probleme.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Geschäftsorientierte Dashboards

Dashboards, die auf Geschäfts-KPIs zugeschnitten sind, sprechen ein breiteres Spektrum von Stakeholdern an. Auch wenn diese Personen vielleicht nicht an Systemmetriken interessiert sind, haben sie dennoch großes Interesse daran, die geschäftlichen Auswirkungen dieser Zahlen zu verstehen. Ein geschäftsorientiertes Dashboard stellt sicher, dass alle technischen und betrieblichen Metriken, die überwacht und analysiert werden, auf die übergeordneten Geschäftsziele ausgerichtet sind. Diese Ausrichtung sorgt für Klarheit und stellt sicher, dass alle gleich darüber informiert sind, was wichtig ist und was nicht. Darüber hinaus sind Dashboards, die Geschäfts-KPIs hervorheben, in der Regel leichter umzusetzen. Sie bieten Stakeholdern die Möglichkeit, in kürzester Zeit den Status der Abläufe, die Bereiche, die Aufmerksamkeit erfordern, und die potenziellen Auswirkungen auf die Geschäftsergebnisse zu verstehen.

Vor diesem Hintergrund sollten Sie bei der Erstellung Ihrer Dashboards sicherstellen, dass ein Gleichgewicht zwischen technischen Metriken und Geschäfts-KPIs besteht. Beide sind wichtig, richten sich aber an unterschiedliche Zielgruppen. Idealerweise sollten Sie über Dashboards verfügen, die einen ganzheitlichen Überblick über den Status und die Leistung des Systems bieten und gleichzeitig wichtige Geschäftsergebnisse und deren Auswirkungen hervorheben.

Amazon-CloudWatch-Dashboards sind anpassbare Startseiten in der CloudWatch-Konsole, mit denen Sie Ihre Ressourcen in einer einzigen Ansicht überwachen können, auch solche, die über verschiedene AWS-Regionen und Konten verteilt sind.

Implementierungsschritte

1. Erstellen eines grundlegenden Dashboards: [Erstellen Sie ein neues Dashboard in CloudWatch](#) und geben Sie ihm einen aussagekräftigen Namen.
2. Verwenden von Markdown-Widgets: Bevor Sie sich mit den Metriken befassen, [verwenden Sie Markdown-Widgets](#), um Ihr Dashboard oben mit Kontext zu versehen. Dieser sollte den Inhalt des Dashboards beschreiben und angeben, welche Bedeutung den dargestellten Metriken zukommt. Er kann auch Links zu anderen Dashboards und Tools zur Fehlerbehebung enthalten.
3. Erstellen von Dashboard-Variablen: [Integrieren Sie gegebenenfalls Dashboard-Variablen](#), um dynamische und flexible Dashboard-Ansichten zu ermöglichen.

4. Erstellen von Metrik-Widgets: [Fügen Sie Metrik-Widgets hinzu](#), um verschiedene Metriken zu visualisieren, die Ihre Anwendung ausgibt, und passen Sie diese Widgets so an, dass sie den Systemstatus und die Geschäftsergebnisse effektiv darstellen.
5. Verwenden von Log-Insights-Abfragen: Verwenden Sie [CloudWatch Log Insights](#), um umsetzbare Metriken aus Ihren Protokollen abzurufen und diese Erkenntnisse auf Ihrem Dashboard anzuzeigen.
6. Einrichten von Alarmen: Integrieren Sie [CloudWatch-Alarme](#) in Ihr Dashboard, um einen raschen Überblick über alle Metriken zu erhalten, die ihre Schwellenwerte überschreiten.
7. Verwenden von Contributor Insights: Integrieren Sie [CloudWatch Contributor Insights](#), um Felder mit hoher Kardinalität zu analysieren und ein besseres Verständnis der wichtigsten Mitwirkenden Ihrer Ressource zu erhalten.
8. Entwerfen benutzerdefinierter Widgets: Für spezielle Anforderungen, die von Standard-Widgets nicht erfüllt werden, sollten Sie es in Betracht ziehen, [benutzerdefinierte Widgets](#) zu erstellen. Diese können Daten aus verschiedenen Datenquellen abrufen oder sie auf spezifische Weise darstellen.
9. Verwenden von AWS Health: AWS Health ist die autoritative Informationsquelle für den Zustand Ihrer AWS Cloud-Ressourcen. Verwenden Sie [AWS Health Dashboard](#) unverändert oder verwenden Sie AWS Health-Daten in Ihren eigenen Dashboards und Tools, damit Sie die richtigen Informationen zur Verfügung haben, um fundierte Entscheidungen zu treffen.
10. Wiederholen und optimieren: Im Laufe der Entwicklung Ihrer Anwendung sollten Sie Ihr Dashboard regelmäßig überprüfen, um sicherzustellen, dass es weiterhin relevant ist.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS04-BP01 Identifizieren Sie die wichtigsten Leistungsindikatoren](#)
- [OPS08-BP01 Analysieren Sie Workload-Metriken](#)
- [OPS08-BP02 Analysieren Sie Workload-Protokolle](#)
- [OPS08-BP03 Analysieren Sie Workload-Traces](#)
- [OPS08-BP04 Erstellen umsetzbarer Warnmeldungen](#)

Zugehörige Dokumente:

- [Erstellung von Dashboards für operative Sichtbarkeit](#)

- [Amazon CloudWatch Dashboards verwenden](#)

Zugehörige Videos:

- [Konto- und regionenübergreifende CloudWatch-Dashboards erstellen](#)
- [AWS re:Invent 2021 - Gain enterprise visibility with AWS Cloud operation dashboards](#)

Zugehörige Beispiele:

- [Workshop zur Beobachtbarkeit](#)
- [Anwendungsüberwachung mit Amazon CloudWatch](#)
- [Dashboards und Einblicke zu AWS Health Events Intelligence](#)
- [Visualisieren von AWS Health-Ereignissen mit Amazon Managed Grafana](#)

OPS 9. Wie können Sie den Zustand Ihrer Operationen beurteilen?

Definieren, erfassen und analysieren Sie Metriken für Operationen, um einen Einblick in Ereignisse rund um Ihre Betriebsabläufe zu erhalten. Dies ist wichtig, damit Sie bei Bedarf entsprechende Maßnahmen ergreifen können.

Best Practices

- [OPS09-BP01 Messen operativer Ziele und KPIs mit Metriken](#)
- [OPS09-BP02 Kommunizieren von Status und Trends zur Sicherung der operativen Transparenz](#)
- [OPS09-BP03 Überprüfen der Betriebsmetriken und Priorisieren von Verbesserungen](#)

OPS09-BP01 Messen operativer Ziele und KPIs mit Metriken

Ermitteln Sie Ziele und KPIs in Ihrem Unternehmen, die operativen Erfolg definieren, und legen Sie Metriken fest, die diese Werte widerspiegeln. Legen Sie Baselines als Bezugspunkt fest und bewerten Sie diese regelmäßig neu. Entwickeln Sie Mechanismen, um diese Metriken von Teams zur Bewertung zu erfassen. Die Metriken von [DevOps Research and Assessment \(DORA\)](#) stellen eine verbreitete Methode zur Messung der Fortschritte bei DevOps-Verfahren für die Softwarebereitstellung dar.

Gewünschtes Ergebnis:

- Die Organisation veröffentlicht und teilt die Ziele und KPIs für die Operations-Teams.
- Sie richten Metriken ein, die diese KPIs widerspiegeln. Mögliche Beispiele:
 - Tiefe der Ticket-Warteschlange oder Durchschnittsalter der Tickets
 - Anzahl der Tickets, gruppiert nach Art des Problems
 - Aufgewendete Zeit für die Bearbeitung von Problemen mit oder ohne standardisierte Betriebsverfahren (SOP)
 - Zeit, die zur Wiederherstellung nach einem fehlgeschlagenen Code-Push aufgewendet wurde
 - Anruflautstärke

Typische Anti-Muster:

- Bereitstellungsfristen werden nicht eingehalten, weil Entwickler mit der Lösung von Problemen beauftragt werden. Entwicklerteams fordern mehr Personal, können aber nicht einschätzen, wie viele Personen benötigt werden, da der Zeitaufwand nicht gemessen werden kann.
- Für die Abwicklung von Kundenanrufen wurde ein Problem-Desk Stufe 1 eingerichtet. Im Laufe der Zeit kamen weitere Workloads hinzu, aber dem Problem-Desk Stufe 1 wurde kein zusätzliches Personal zugewiesen. Die Kundenzufriedenheit leidet, da immer mehr Anrufe nötig sind und Probleme länger ungelöst bleiben. Das Management sieht diese Anzeichen jedoch nicht und ermöglicht keine Gegenmaßnahmen.
- Eine problematische Workload wurde zur Bearbeitung an ein separates Operations-Team übergeben. Im Gegensatz zu anderen Workloads wurde diese neue Workload nicht mit ordnungsgemäßer Dokumentation und Runbooks geliefert. Daher verbringen Teams mehr Zeit damit, Fehler zu suchen und zu beheben. Es gibt jedoch keine Metriken, die dies dokumentieren, was die Rechenschaftspflicht erschwert.

Vorteile der Nutzung dieser bewährten Methode: Während die Workload-Überwachung den Status unserer Anwendungen und Services anzeigt, liefert die Überwachung von Operations-Teams den Verantwortlichen Erkenntnisse hinsichtlich Veränderungen bei den Benutzern dieser Workloads, z. B. geänderte Geschäftsanforderungen. Messen Sie die Effektivität dieser Teams und bewerten Sie sie im Hinblick auf Ihre operativen Ziele, indem Sie Metriken erstellen, die den operativen Status widerspiegeln können. Anhand von Metriken können Supportprobleme aufgezeigt oder Abweichungen von einem angestrebten Servicelevel erkannt werden.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Planen Sie Besprechungen mit der Geschäftsleitung und Stakeholdern, um die allgemeinen Ziele des Services festzulegen. Ermitteln Sie, worin die Aufgaben der verschiedenen Operations-Teams bestehen sollten und mit welchen Herausforderungen sie beauftragt werden könnten. Führen Sie anhand dieser Daten ein Brainstorming der wichtigsten Leistungsindikatoren (KPIs) durch, die diese operativen Ziele widerspiegeln könnten. Dies können Faktoren wie Kundenzufriedenheit, die Zeitspanne zwischen Entwurf und Bereitstellung von Features, der durchschnittliche Zeitaufwand für die Problemlösung und andere sein.

Identifizieren Sie anhand der KPIs die Metriken und Datenquellen, die diese Ziele am besten widerspiegeln könnten. Kundenzufriedenheit kann eine Kombination aus verschiedenen Metriken wie Warte- oder Reaktionszeiten bei Anrufen, Zufriedenheitswerte und Art der dargelegten Probleme sein. Die Bereitstellungszeiten können die Summe des Zeitaufwands sein, der für Tests und Bereitstellungen benötigt wird, zuzüglich aller Korrekturen nach der Bereitstellung, die hinzugefügt werden mussten. Statistiken, aus denen hervorgeht, wie viel Zeit für verschiedene Arten von Problemen aufgewendet wurde (oder wie viele dieser Probleme auftraten), können Aufschluss darüber geben, wo gezielte Anstrengungen erforderlich sind.

Ressourcen

Zugehörige Dokumente:

- [Quick – KPIs verwenden](#)
- [Amazon CloudWatch – Verwenden von Metriken](#)
- [Erstellung von Dashboards](#)
- [Wie Sie mit dem KPI-Dashboard Ihre KPIs zur Kostenoptimierung nachverfolgen](#)
- [AWS-DevOps-Anleitung](#)

Zugehörige Beispiele:

- [Überwachen Sie die Leistung Ihrer Softwarebereitstellung mithilfe systemeigener AWS-Tools für Überwachung und Beobachtbarkeit](#)
- [Stellen Sie mit DORA-Metriken ein Gleichgewicht zwischen Geschwindigkeit und Stabilität bei der Bereitstellung her](#)
- [Beispiel für operative MLOps-Metriken in der Finanzdienstleistungsbranche](#)
- [Wie Sie mit dem KPI-Dashboard Ihre KPIs für die Kostenoptimierung nachverfolgen](#)

OPS09-BP02 Kommunizieren von Status und Trends zur Sicherung der operativen Transparenz

Wenn Sie in Erfahrung bringen wollen, wann Ergebnisse gefährdet sein könnten, ob zusätzliche Workloads unterstützt werden können oder nicht oder welche Auswirkungen Änderungen auf Ihre Teams hatten, müssen Sie unbedingt den Status Ihrer Betriebsabläufe und deren Trendrichtung kennen. Bei Betriebsereignissen können Statusseiten, auf denen Benutzer und Operations-Teams Informationen abrufen können, den Druck auf die Kommunikationskanäle verringern und Informationen proaktiv verbreiten.

Gewünschtes Ergebnis:

- Betriebsleiter erhalten auf einen Blick Erkenntnisse darüber, welches Anrufvolumen ihre Teams bewältigen müssen und welche Maßnahmen möglicherweise im Gange sind, z. B. Bereitstellungen.
- Wenn Auswirkungen auf den normalen Betrieb auftreten, werden Warnmeldungen an Stakeholder und Benutzergemeinschaften versendet.
- Unternehmensleitung und Stakeholder können als Reaktion auf eine Warnung oder Auswirkung eine Statusseite aufrufen und Informationen zu einem betrieblichen Ereignis abrufen, z. B. Kontaktstellen, Ticketinformationen und erwartete Wiederherstellungszeiten.
- Führungskräften und anderen Stakeholdern werden Berichte zur Verfügung gestellt, damit sie über Betriebsstatistiken wie das Anrufvolumen über einen bestimmten Zeitraum, Benutzerzufriedenheitswerte, Anzahl ausstehender Tickets und deren Alter informiert sind.

Typische Anti-Muster:

- Eine Workload fällt aus und ein Dienst wird nicht verfügbar. Das Anrufvolumen steigt, da Benutzer wissen möchten, was vor sich geht. Manager erhöhen dieses Volumen, da sie nachfragen, wer an dem Problem arbeitet. Verschiedene Operations-Teams bemühen sich doppelt, Untersuchungen durchzuführen.
- Der Wunsch nach neuen Funktionen führt dazu, dass mehrere Mitarbeiter umpositioniert werden, um an einem speziellen technischen Vorhaben zu arbeiten. Dadurch entstehende Lücken werden nicht aufgefüllt und die Problemlösungszeiten steigen. Diese Informationen werden nicht erfasst, und erst nach mehreren Wochen und viel negativem Feedback unzufriedener Benutzer wird die Unternehmensleitung auf das Problem aufmerksam.

Vorteile der Nutzung dieser bewährten Methode: Bei betrieblichen Ereignissen, die das Geschäft beeinträchtigen, wird manchmal viel Zeit und Energie damit verschwendet, Informationen von verschiedenen Teams abzufragen, die versuchen, die Situation zu verstehen. Durch die Einrichtung

und Verbreitung von Statusseiten und Dashboards können Stakeholder rasch Informationen darüber abrufen, ob ein Problem festgestellt wurde oder nicht, wer mit der Lösung des Problems beschäftigt ist oder wann mit einer Rückkehr zum normalen Betrieb zu rechnen ist. Dadurch müssen die Teammitglieder nicht zu viel Zeit damit verbringen, anderen den Status mitzuteilen und haben mehr Zeit, Probleme zu lösen.

Darüber hinaus können Dashboards und Berichte Entscheidungsträgern und Stakeholdern Einblicke bieten, um zu sehen, wie Operations-Teams auf Geschäftsanforderungen reagieren können und wie ihre Ressourcen zugewiesen werden. Dies ist entscheidend, um festzustellen, ob angemessene Ressourcen zur Unterstützung des Unternehmens vorhanden sind.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Erstellen Sie Dashboards, die die aktuellen Schlüsselmetriken für Ihre Operations-Teams anzeigen, und machen Sie sie sowohl für die Betriebsleitung als auch für das Management leicht zugänglich.

Erstellen Sie Statusseiten, die schnell aktualisiert werden können, um zu zeigen, wann sich ein Vorfall oder ein Ereignis abspielt, wer dafür verantwortlich ist und wer die Reaktion darauf koordiniert. Kommunizieren Sie auf dieser Seite alle Schritte oder Problemumgehungen, die Benutzer in Betracht ziehen sollten, und machen Sie sie für alle Beteiligten verfügbar. Bitten Sie Benutzer, zuerst diese Seite zu überprüfen, wenn sie mit einem unbekannten Problem konfrontiert werden.

Erfassen Sie Daten und stellen Sie Berichte bereit, die den Zustand der Betriebsabläufe im Zeitverlauf aufzeigen, und verteilen Sie diese an Führungskräfte und Entscheidungsträger, um die Arbeit des Betriebs sowie die Herausforderungen und Bedürfnisse zu veranschaulichen.

Teilen Sie die Metriken und Berichte, die die Ziele und KPIs am besten widerspiegeln, mit den Teams, und zeigen Sie ihnen, wo sie besonders deutlich einen Wandel vorangetrieben haben. Nehmen Sie sich Zeit für diese Aktivitäten, um den Abläufen innerhalb und zwischen Teams mehr Bedeutung beizumessen.

Verwenden Sie [AWS Health](#) zusammen mit Ihren eigenen Dashboards oder integrieren Sie AWS Health-Ereignisse, sodass Ihre Teams Anwendungsprobleme mit dem Status des AWS-Service korrelieren können.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS09-BP01 Messen operativer Ziele und KPIs mit Metriken](#)

Zugehörige Dokumente:

- [Fortschritt messen](#)
- [Erstellung von Dashboards für operative Sichtbarkeit](#)

Zugehörige Beispiele:

- [Datenoperationen](#)
- [Wie Sie mit dem KPI-Dashboard Ihre KPIs zur Kostenoptimierung nachverfolgen](#)
- [The Importance of Key Performance Indicators \(KPIs\) for Large-Scale Cloud Migrations](#)

OPS09-BP03 Überprüfen der Betriebsmetriken und Priorisieren von Verbesserungen

Durch die Bereitstellung von Zeit und Ressourcen für die Überprüfung des Betriebsstatus wird sichergestellt, dass die Betreuung der täglichen Geschäftstätigkeit weiterhin Priorität hat. Bringen Sie Betriebsleiter und Stakeholder an einen Tisch, um regelmäßig Metriken zu überprüfen, Ziele und Vorgaben zu bestätigen oder zu ändern und Verbesserungen zu priorisieren.

Gewünschtes Ergebnis:

- Betriebsleiter und Mitarbeiter treffen sich regelmäßig, um die Metriken für einen bestimmten Berichtszeitraum zu überprüfen. Herausforderungen werden kommuniziert, Erfolge gefeiert und gewonnene Erkenntnisse geteilt.
- Stakeholder und Unternehmensleiter werden regelmäßig über den Stand der laufenden Operationen informiert und um ihre Meinung gebeten, was Ziele, KPIs und zukünftige Initiativen angeht. Kompromisse zwischen Servicebereitstellung, Betrieb und Wartung werden erörtert und in Zusammenhang gebracht.

Typische Anti-Muster:

- Ein neues Produkt wird auf den Markt gebracht, aber die Operations-Teams der Stufe 1 und 2 sind nicht ausreichend geschult, um Support zu leisten, oder bräuchten zusätzliches Personal. Metriken, die den Anstieg der Bearbeitungsdauer von Tickets und der Anzahl der Vorfälle belegen, werden von Führungskräften nicht berücksichtigt. Erst Wochen später werden Maßnahmen ergriffen, weil die Zahl der Abonnements zu sinken beginnt, da unzufriedene Benutzer die Plattform verlassen.

- Ein manuelles Verfahren zur Durchführung von Wartungsarbeiten an einer Workload gibt es schon lange. Der Wunsch nach Automatisierung war zwar vorhanden, hatte aber angesichts der geringen Bedeutung des Systems nur geringe Priorität. Im Laufe der Zeit hat das System jedoch an Bedeutung gewonnen und heute nehmen diese manuellen Prozesse einen Großteil der Betriebszeit in Anspruch. Es sind keine Ressourcen für die Bereitstellung von mehr Tools für den Betrieb vorgesehen, was zu einer Überlastung der Mitarbeiter führt, wenn die Workload zunimmt. Die Unternehmensleitung wird sich der Probleme bewusst, als sie erfährt, dass Mitarbeiter zu anderen Wettbewerbern wechseln.

Vorteile der Nutzung dieser bewährten Methode: In einigen Unternehmen kann es zu einer Herausforderung werden, für die Servicebereitstellung die gleiche Zeit und Aufmerksamkeit aufzuwenden, die neuen Produkten oder Angeboten entgegengebracht wird. Wenn dies zutrifft, kann der Geschäftsbereich darunter leiden und das erwartete Serviceniveau verschlechtert sich nach und nach. Dies liegt daran, dass sich der Betrieb nicht mit dem wachsenden Geschäft ändert und weiterentwickelt, wodurch er bald ins Hintertreffen gerät. Ohne eine regelmäßige Überprüfung der Erkenntnisse, die Operations erfasst, wird das Risiko für das Unternehmen möglicherweise erst sichtbar, wenn es zu spät ist. Wenn jedoch sowohl dem Betriebspersonal als auch den Führungskräften Zeit für die Überprüfung von Metriken und Verfahren eingeräumt wird, bleibt die entscheidende Rolle, die der Betrieb spielt, sichtbar und Risiken können erkannt werden, lange bevor sie ein kritisches Niveau erreichen. Operations-Teams erhalten einen besseren Überblick über bevorstehende Geschäftsänderungen und Initiativen, sodass proaktive Maßnahmen ergriffen werden können. Wenn Führungskräfte die Gelegenheit haben, die Betriebsmetriken zu prüfen, erkennen sie, welche Rolle diese Teams für die Kundenzufriedenheit spielen –sowohl intern als auch extern. So können sie Operations die Möglichkeit geben, Entscheidungen im Hinblick auf Prioritäten besser abzuwägen oder sicherzustellen, dass die Teams über die Zeit und die Ressourcen verfügen, um mit neuen Geschäfts- und Workload-Initiativen zu wachsen und sich weiterzuentwickeln.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Nehmen Sie sich Zeit, um die Betriebsmetriken gemeinsam mit Stakeholdern und Operations-Teams zu überprüfen und die Berichtsdaten zu lesen. Stellen Sie diese Berichte in den Kontext der Ziele und Vorgaben der Organisation, um festzustellen, ob sie erreicht werden. Identifizieren Sie Unklarheiten, bei denen die Ziele nicht eindeutig sind oder wo Konflikte bestehen zwischen dem, was verlangt wird, und dem, was gegeben wird.

Identifizieren Sie, wo Zeit, Mitarbeiter und Tools zu Betriebsergebnissen beitragen können. Ermitteln Sie, auf welche KPIs sich dies auswirken würde und welche Erfolgsziele verfolgt werden sollten. Greifen Sie Ihre Überlegungen regelmäßig wieder auf, um sicherzustellen, dass der Betrieb über ausreichende Ressourcen verfügt, um den Geschäftsbereich zu unterstützen.

Ressourcen

Zugehörige Dokumente:

- [Amazon Athena](#)
- [Referenzinformationen zu Metriken und Dimensionen von Amazon CloudWatch](#)
- [Amazon Quick](#)
- [AWS Glue](#)
- [AWS Glue Data Catalog](#)
- [Sammeln von Metriken und Protokollen von Amazon-EC2-Instances und On-Premises-Servern mit dem Amazon-CloudWatch-Agenten](#)
- [Verwenden von Amazon-CloudWatch-Metriken](#)

OPS 10. Wie bewältigen Sie Workload- und operationsspezifische Ereignisse?

Erarbeiten und prüfen Sie Verfahren für die Reaktion auf Ereignisse, um Beeinträchtigungen für Ihre Workload zu minimieren.

Best Practices

- [OPS10-BP01 Verwenden eines Prozesses für die Bewältigung von Ereignissen, Vorfällen und Problemen](#)
- [OPS10-BP02 Implementieren eines Prozesses für jede Warnmeldung](#)
- [OPS10-BP03 Priorisieren von betrieblichen Ereignissen auf Basis der Auswirkung auf das Unternehmen](#)
- [OPS10-BP04 Definieren von Eskalationspfaden](#)
- [OPS10-BP05 Definieren eines Kundenkommunikationsplans für Ereignisse, die sich auf den Service auswirken](#)
- [OPS10-BP06 Bekanntgeben des Status über Dashboards](#)
- [OPS10-BP07 Automatisieren Sie Reaktionen auf Ereignisse](#)

OPS10-BP01 Verwenden eines Prozesses für die Bewältigung von Ereignissen, Vorfällen und Problemen

Die Fähigkeit, Ereignisse, Vorfälle und Probleme effizient zu verwalten, ist der Schlüssel zur Aufrechterhaltung der Workload und der Leistung. Es ist wichtig, die Unterschiede zwischen diesen Elementen zu erkennen und zu verstehen, um eine effektive Reaktions- und Lösungsstrategie zu entwickeln. Die Einrichtung und Einhaltung eines klar definierten Prozesses für jeden Aspekt hilft Ihrem Team, alle auftretenden betrieblichen Herausforderungen schnell und effektiv zu bewältigen.

Gewünschtes Ergebnis: Ihr Unternehmen verwaltet betriebliche Ereignisse, Vorfälle und Probleme effektiv durch gut dokumentierte und zentral gespeicherte Prozesse. Diese Prozesse werden ständig aktualisiert, um Änderungen zu berücksichtigen, die Handhabung zu optimieren und eine hohe Servicezuverlässigkeit und Workload-Leistung aufrechtzuerhalten.

Typische Anti-Muster:

- Sie reagieren eher reaktiv als proaktiv auf Ereignisse.
- Bei verschiedenen Arten von Ereignissen oder Vorfällen werden inkonsistente Ansätze verfolgt.
- Ihr Unternehmen analysiert keine Vorfälle und lernt nicht aus ihnen, um zukünftige Vorfälle zu verhindern.

Vorteile der Nutzung dieser bewährten Methode:

- optimierte und standardisierte Reaktionsprozesse
- geringere Auswirkungen von Vorfällen auf Services und Kunden
- beschleunigte Problemlösung
- kontinuierliche Verbesserung der betrieblichen Abläufe

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Wenn Sie diese bewährte Methode implementieren, bedeutet dies, dass Sie Workload-Ereignisse nachverfolgen. Sie haben Prozesse für den Umgang mit Vorfällen und Problemen. Die Prozesse werden dokumentiert, geteilt und oft aktualisiert. Die Probleme werden identifiziert, priorisiert und behoben.

Verstehen von Ereignissen, Vorfällen und Problemen

- **Ereignisse:** Bei einem Ereignis handelt es sich um eine Beobachtung einer Aktion, eines Vorkommens oder einer Statusänderung. Ereignisse können geplant oder ungeplant sein und sie können intern oder extern zur Workload entstehen.
- **Vorfälle:** Vorfälle sind Ereignisse, die eine Reaktion erfordern, wie ungeplante Unterbrechungen oder Beeinträchtigungen der Servicequalität. Sie stellen Störungen dar, die sofortige Aufmerksamkeit erfordern, um den normalen Workload-Betrieb wiederherzustellen.
- **Probleme:** Probleme sind die zugrundeliegenden Ursachen für einen oder mehrere Vorfälle. Bei der Identifizierung und Lösung von Problemen geht es darum, den Vorfällen auf den Grund zu gehen, um zukünftige Vorfälle zu verhindern.

Implementierungsschritte

Ereignisse

1. Überwachen von Ereignissen:

- [Implementieren Sie Beobachtbarkeit](#) und [nutzen Sie Workload-Beobachtbarkeit](#).
- Monitor-Aktionen, die von einem Benutzer, einer Rolle oder einem AWS-Service ausgeführt werden, werden als Ereignisse in [AWS CloudTrail](#) aufgezeichnet.
- Reagieren Sie auf betriebliche Änderungen in Ihren Anwendungen in Echtzeit mit [Amazon EventBridge](#).
- Bewerten, überwachen und zeichnen Sie Änderungen der Ressourcenkonfiguration mit [AWS Config](#) kontinuierlich auf.

2. Erstellen von Prozessen:

- Entwickeln Sie ein Verfahren zur Bewertung, welche Ereignisse signifikant sind und überwacht werden müssen. Dies beinhaltet die Festlegung von Schwellenwerten und Parametern für normale und abnormale Aktivitäten.
- Legen Sie Kriterien für die Eskalation eines Ereignisses in Bezug auf einen Vorfall fest. Dies kann auf Grundlage des Schweregrads, der Auswirkungen auf die Benutzer oder der Abweichung vom erwarteten Verhalten erfolgen.
- Überprüfen Sie regelmäßig die Prozesse zur Überwachung und Reaktion auf Ereignisse. Dazu gehören die Analyse früherer Vorfälle, die Anpassung von Schwellenwerten und die Verfeinerung von Warnmechanismen.

Vorfälle

1. Reaktion auf Vorfälle:

- Nutzen Sie die Erkenntnisse aus den Tools zur Beobachtbarkeit, um Vorfälle schnell zu erkennen und darauf zu reagieren.
- Implementieren Sie [AWS Systems Manager Ops Center](#), um betriebliche Aufgaben und Vorfälle zu sammeln, zu organisieren und zu priorisieren.
- Nutzen Sie Services wie [Amazon CloudWatch](#) und [AWS X-Ray](#) für eingehendere Analysen und Problembehebungen.
- Ziehen Sie [AWS Managed Services \(AMS\)](#) für ein verbessertes Vorfallmanagement in Betracht, indem Sie die proaktiven, präventiven und detektivischen Fähigkeiten nutzen. AMS erweitert den betrieblichen Support um Services wie Überwachung, Vorfallerkennung und -reaktion sowie Sicherheitsmanagement.
- Kunden von Enterprise Support können [AWS-Vorfallerkennung und -reaktion](#) verwenden, wodurch eine kontinuierliche proaktive Überwachung und ein Vorfallmanagement für Produktions-Workloads ermöglicht wird.

2. Erstellen eines Vorfallmanagementprozesses:

- Richten Sie einen strukturierten Vorfallmanagementprozess ein, der klare Rollen, Kommunikationsprotokolle und Lösungsschritte umfasst.
- Integrieren Sie das Vorfallmanagement mit Tools wie [Amazon Q Developer in Chat-Anwendungen](#) für eine effiziente Reaktion und Koordination.
- Kategorisieren Sie Vorfälle nach Schweregrad mit vordefinierten [Vorfallreaktionsplänen](#) für jede Kategorie.

3. Lernen und Verbessern:

- Führen Sie [Analysen nach Vorfällen](#) aus, um die Grundursachen und die Effektivität der Lösung zu verstehen.
- Aktualisieren und verbessern Sie die Reaktionspläne kontinuierlich auf Grundlage von Überprüfungen und sich entwickelnden Praktiken.
- Dokumentieren Sie die gewonnenen Erkenntnisse und geben Sie sie an andere Teams weiter, um die betriebliche Widerstandsfähigkeit zu verbessern.
- Kunden mit Enterprise Support können den [Workshop zum Vorfallmanagement](#) bei ihrem Technical Account Manager anfordern. Dieser angeleitete Workshop testet Ihren vorhandenen Reaktionsplan für Vorfälle und hilft Ihnen, Verbesserungsmöglichkeiten zu identifizieren.

1. Identifizieren von Problemen:

- Verwenden Sie Daten aus früheren Vorfällen, um wiederkehrende Muster zu erkennen, die auf tiefere systemische Probleme hinweisen könnten.
- Nutzen Sie Tools wie [AWS CloudTrail](#) und [Amazon CloudWatch](#), um Trends zu analysieren und grundlegende Probleme aufzudecken.
- Binden Sie funktionsübergreifende Teams ein, einschließlich Betriebs-, Entwicklungs- und Geschäftsbereiche, um unterschiedliche Sichtweisen auf die Grundursachen zu gewinnen.

2. Erstellen eines Problemmanagementprozesses:

- Entwickeln Sie einen strukturierten Prozess für das Problemmanagement, der sich auf langfristige Lösungen statt auf schnelle Lösungen konzentriert.
- Integrieren Sie Techniken zur Ursachenanalyse, um die zugrunde liegenden Ursachen von Vorfällen zu untersuchen und zu verstehen.
- Aktualisieren Sie Betriebsrichtlinien, Verfahren und Infrastruktur auf Grundlage der Erkenntnisse, um Wiederholungen zu verhindern.

3. Kontinuierliche Verbesserungen:

- Fördern Sie eine Kultur des ständigen Lernens und der Verbesserung und ermutigen Sie Ihre Teams, potenzielle Probleme proaktiv zu erkennen und anzugehen.
- Überprüfen und überarbeiten Sie regelmäßig die Problemmanagementprozesse und -tools, um sie an die sich entwickelnde Geschäfts- und Technologielandschaft anzupassen.
- Tauschen Sie Erkenntnisse und bewährte Methoden innerhalb des Unternehmens aus, um eine widerstandsfähigere und effizientere Betriebsumgebung zu schaffen.

4. Einsatz von AWS Support:

- Nutzen Sie AWS-Support-Ressourcen, z. B. [AWS Trusted Advisor](#), für proaktive Anleitungen und Optimierungsempfehlungen.
- Kunden von Enterprise Support können auf spezielle Programme wie [AWS Countdown](#) zugreifen, um bei kritischen Ereignissen Unterstützung zu erhalten.

Aufwand für den Implementierungsplan: Mittel

Ressourcen

Zugehörige bewährte Methoden:

- [OPS04-BP01 Identifizieren Sie die wichtigsten Leistungsindikatoren](#)

- [OPS04-BP02 Implementieren Sie Anwendungstelemetrie](#)
- [OPS07-BP03 Verwenden von Runbooks zur Durchführung von Verfahren](#)
- [OPS07-BP04 Verwenden von Playbooks zum Untersuchen von Problemen](#)
- [OPS08-BP01 Analysieren Sie Workload-Metriken](#)
- [OPS11-BP02 Führen Sie eine Analyse nach dem Vorfall durch](#)

Zugehörige Dokumente:

- [AWS Security Incident Response Guide](#)
- [AWS-Vorfallerkennung und -reaktion](#)
- [AWS Cloud Adoption Framework: Betriebsperspektive – Vorfall- und Problemmanagement](#)
- [Vorfallmanagement im Zeitalter von DevOps und SRE](#)
- [PagerDuty – What is Incident Management?](#)

Zugehörige Videos:

- [Die besten Tipps zur Reaktion auf Vorfälle in AWS](#)
- [AWS re:Invent 2022 – Die Amazon Builders' Library: 25 Jahre operative Exzellenz von Amazon](#)
- [AWS re:Invent 2022 – AWS-Vorfallerkennung und -reaktion \(SUP201\)](#)
- [Introducing Incident Manager from AWS Systems Manager](#)

Zugehörige Beispiele:

- [AWS Proactive Services – Workshop zum Vorfallmanagement](#)
- [Automatisierung der Vorfallbehandlung mit PagerDuty und AWS Systems Manager Incident Manager](#)
- [Einbeziehung des Notfallteams in die Bereitschaftsdienstpläne in AWS Systems Manager Incident Manager](#)
- [Verbesserung der Sichtbarkeit und Zusammenarbeit bei der Bearbeitung von Vorfällen in AWS Systems Manager Incident Manager](#)
- [Vorfallberichte und Serviceanfragen in AMS](#)

Zugehörige Services:

- [Amazon EventBridge](#)

OPS10-BP02 Implementieren eines Prozesses für jede Warnmeldung

Die Einrichtung eines klaren und definierten Prozesses für jede Warnmeldung in Ihrem System ist für ein effektives und effizientes Vorfallmanagement unerlässlich. Diese Vorgehensweise stellt sicher, dass jede Warnmeldung zu einer spezifischen, umsetzbaren Reaktion führt, wodurch die Zuverlässigkeit und Reaktionsfähigkeit Ihrer Abläufe verbessert wird.

Gewünschtes Ergebnis: Jede Warnmeldung leitet einen bestimmten, genau definierten Reaktionsplan ein. Wenn möglich, werden die Antworten automatisiert, mit klaren Zuständigkeiten und einem definierten Eskalationspfad. Warnmeldungen sind mit einer aktuellen Wissensdatenbank verknüpft, sodass jeder Bediener konsistent und effektiv reagieren kann. Die Antworten sind schnell und einheitlich, was die betriebliche Effizienz und Zuverlässigkeit erhöht.

Typische Anti-Muster:

- Für Warnmeldungen gibt es keinen vordefinierten Reaktionsprozess, was zu provisorischen und verzögerten Lösungen führt.
- Eine Überlastung mit Warnmeldungen führt dazu, dass wichtige Warnmeldungen übersehen werden.
- Warnmeldungen werden uneinheitlich gehandhabt, da es an klaren Zuständigkeiten und Verantwortlichkeiten mangelt.

Vorteile der Nutzung dieser bewährten Methode:

- Weniger Ermüdungserscheinungen, da nur umsetzbare Warnmeldungen ausgelöst werden.
- Geringere durchschnittliche Zeit bis zur Behebung (MTTR) von Betriebsproblemen.
- Geringere durchschnittliche Zeit bis zur Untersuchung, was zur Verringerung der MTTR beiträgt.
- Verbesserte Fähigkeit, operative Reaktionen zu skalieren.
- Verbesserte Konsistenz und Zuverlässigkeit beim Umgang mit Betriebsereignissen.

Sie haben beispielsweise einen definierten Prozess für AWS Health-Ereignisse für kritische Konten eingerichtet, einschließlich Anwendungsalarmen, operationaler Probleme und geplanter Lebenszyklusereignisse (z. B. die Aktualisierung von Amazon-EKS-Versionen vor der automatischen Aktualisierung von Clustern), und stellen Ihren Teams die Möglichkeit bereit, diese Ereignisse aktiv

zu überwachen, zu kommunizieren und auf sie zu reagieren. Diese Maßnahmen helfen Ihnen, Serviceunterbrechungen aufgrund AWS-seitiger Änderungen zu verhindern oder diese schneller zu beheben, wenn unerwartete Probleme auftreten.

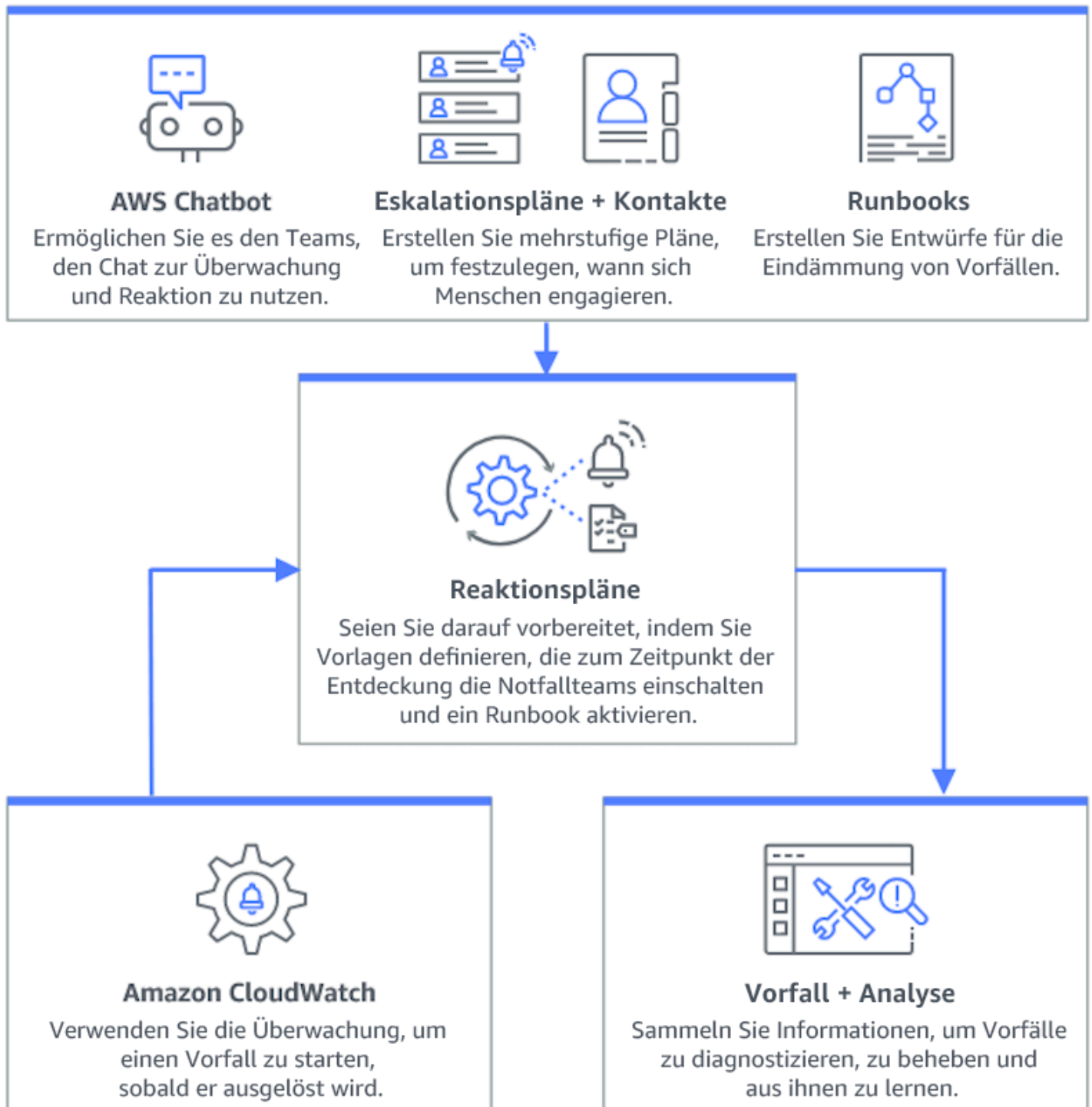
Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Ein Prozess pro Warnmeldung beinhaltet die Erstellung eines klaren Reaktionsplans für jede Warnmeldung, die Automatisierung von Reaktionen (soweit dies möglich ist) und die kontinuierliche Optimierung dieser Prozesse auf Grundlage des betrieblichen Feedbacks und der sich entwickelnden Anforderungen.

Implementierungsschritte

Das folgende Diagramm veranschaulicht den Arbeitsablauf für das Vorfallmanagement in [AWS Systems Manager Incident Manager](#). Es ist so konzipiert, dass es schnell auf betriebliche Probleme reagiert, indem es automatisch Vorfälle als Reaktion auf bestimmte Ereignisse von [Amazon CloudWatch](#) oder [Amazon EventBridge](#) erstellt. Wenn ein Vorfall entweder automatisch oder manuell erstellt wird, zentralisiert Incident Manager die Verwaltung des Vorfalls, organisiert relevante Informationen über AWS-Ressourcen und initiiert vordefinierte Reaktionspläne. Dazu gehört das Ausführen von Systems-Manager-Automation-Runbooks für sofortige Maßnahmen sowie das Erstellen eines übergeordneten betrieblichen Arbeitselements in OpsCenter, um verwandte Aufgaben und Analysen zu verfolgen. Dieser optimierte Prozess beschleunigt und koordiniert die Reaktion auf Vorfälle in Ihrer gesamten AWS-Umgebung.



1. Verwendung zusammengesetzter Alarme: Erstellen Sie [zusammengesetzte Alarme](#) in CloudWatch, um zusammenhängende Alarme zu gruppieren, das Rauschen zu reduzieren und sinnvollere Reaktionen zu ermöglichen.

2. Bleiben Sie mit [AWS Health](#) auf dem Laufenden: AWS Health ist die maßgebliche Informationsquelle für den Zustand Ihrer AWS Cloud-Ressourcen. Ermöglicht AWS Health die Visualisierung und den Erhalt von Benachrichtigungen über aktuelle Serviceereignisse und bevorstehende Änderungen, z. B. geplante Lebenszyklusereignisse. So können Sie Maßnahmen ergreifen, um die Auswirkungen zu minimieren.
 - a. [Erstellen Sie angepasste AWS Health-Ereignisbenachrichtigungen](#) für E-Mail- und Chat-Kanäle über [AWS-Benutzerbenachrichtigungen](#) und integrieren Sie diese programmgesteuert mit [Ihren Überwachungs- und Warnungstools über Amazon EventBridge](#) oder die [AWS Health-API](#).
 - b. Planen und verfolgen Sie den Fortschritt bei Gesundheitsereignissen, die Maßnahmen erfordern, indem Sie diese in Änderungsmanagement- oder ITSM-Tools (wie [Jira](#) oder [ServiceNow](#)) integrieren, die Sie möglicherweise bereits über Amazon EventBridge oder die AWS Health-API verwenden.
 - c. Wenn Sie AWS Organizations verwenden, aktivieren Sie die [Organisationsansicht für AWS Health](#), um AWS Health-Ereignisse über Konten hinweg zu aggregieren.
3. Integration von Amazon-CloudWatch-Alarmen in Incident Manager: Konfigurieren Sie CloudWatch-Alarme für die automatische Erstellung von Vorfällen in [AWS Systems Manager Incident Manager](#).
4. Integration von Amazon EventBridge in Incident Manager: Erstellen Sie [EventBridge-Regeln](#), um auf Ereignisse zu reagieren und Vorfälle mithilfe definierter Reaktionspläne zu erstellen.
5. Vorbereitung auf Vorfälle in Incident Manager:
 - Richten Sie in Incident Manager detaillierte [Reaktionspläne](#) für jede Art von Warnmeldung ein.
 - Richten Sie über [Amazon Q Developer in Chat-Anwendungen](#) Chat-Kanäle ein, die mit Reaktionsplänen in Incident Manager verknüpft sind und die Echtzeitkommunikation bei Vorfällen über Plattformen wie Slack, Microsoft Teams und Amazon Chime ermöglichen.
 - Integrieren Sie [Systems-Manager-Automation-Runbooks](#) in Incident Manager, um automatisierte Reaktionen auf Vorfälle zu ermöglichen.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS04-BP01 Identifizieren Sie die wichtigsten Leistungsindikatoren](#)
- [OPS08-BP04 Erstellen umsetzbarer Warnmeldungen](#)

Zugehörige Dokumente:

- [AWS Cloud Adoption Framework: Betriebsperspektive – Vorfall- und Problemmanagement](#)
- [Verwenden von Amazon-CloudWatch-Alarmen](#)
- [Einrichten von AWS Systems Manager Incident Manager](#)
- [Preparing for incidents in Incident Manager](#)

Zugehörige Videos:

- [Die besten Tipps zur Reaktion auf Vorfälle von AWS](#)
- [re:Invent 2023 | Manage resource lifecycle events at scale with AWS Health](#)

Zugehörige Beispiele:

- [AWS-Workshops – AWS Systems Manager Incident Manager – Automatisierung der Reaktion auf Sicherheitsvorfälle](#)

OPS10-BP03 Priorisieren von betrieblichen Ereignissen auf Basis der Auswirkung auf das Unternehmen

Eine schnelle Reaktion auf Betriebsereignisse ist von entscheidender Bedeutung, aber nicht alle Ereignisse sind gleich. Wenn Sie Ihre Prioritäten auf Grundlage der geschäftlichen Auswirkungen festlegen, müssen Sie sich auch vorrangig mit Ereignissen befassen, die erhebliche Folgen haben könnten, wie z. B. Sicherheit, finanzielle Verluste, Verstöße gegen Vorschriften oder Rufschädigung.

Gewünschtes Ergebnis: Die Reaktionen auf betriebliche Ereignisse werden auf Grundlage der potenziellen Auswirkungen auf die Geschäftsabläufe und -ziele priorisiert. Dadurch werden die Reaktionen effizient und effektiv.

Typische Anti-Muster:

- Jedes Ereignis wird mit der gleichen Dringlichkeit behandelt, was zu Verwirrung und Verzögerungen bei der Behandlung kritischer Probleme führt.
- Sie unterscheiden nicht zwischen Ereignissen mit hoher und geringer Auswirkung, was zu einer Fehlallokation von Ressourcen führt.
- Ihrem Unternehmen fehlt ein klarer Rahmen für die Priorisierung, was zu inkonsistenten Reaktionen auf Betriebsereignisse führt.

- Ereignisse werden in der Reihenfolge ihrer Meldung priorisiert und nicht nach ihrer Auswirkung auf die Geschäftsergebnisse.

Vorteile der Nutzung dieser bewährten Methode:

- Stellt sicher, dass wichtige Geschäftsfunktionen zuerst berücksichtigt werden, um mögliche Schäden zu minimieren.
- Verbessert die Ressourcenzuweisung bei mehreren gleichzeitigen Ereignissen.
- Verbessert die Fähigkeit der Organisation, das Vertrauen zu erhalten und die gesetzlichen Anforderungen zu erfüllen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Wenn Sie mit mehreren betrieblichen Ereignissen konfrontiert sind, ist ein strukturierter Ansatz zur Priorisierung auf Grundlage von Auswirkungen und Dringlichkeit unerlässlich. Dieser Ansatz hilft Ihnen, fundierte Entscheidungen zu treffen, Ihre Maßnahmen auf die Bereiche zu lenken, wo sie am dringendsten benötigt werden, und das Risiko für die Geschäftskontinuität zu mindern.

Implementierungsschritte

1. Bewertung von Auswirkungen: Entwickeln Sie ein Klassifizierungssystem, um den Schweregrad von Ereignissen im Hinblick auf ihre potenziellen Auswirkungen auf den Geschäftsbetrieb und die Ziele zu bewerten. Das folgende Beispiel zeigt die Wirkungskategorien:

Auswirkungsgrad	Beschreibung
Hoch	Betrifft viele Mitarbeiter oder Kunden, hohe finanzielle Auswirkungen, hoher Reputationsschaden oder Verletzungen
Mittel	Betrifft eine Gruppe von Mitarbeitern oder Kunden, mäßige finanzielle Auswirkungen oder mäßiger Reputationsschaden

Auswirkungsgrad	Beschreibung
Niedrig	Betrifft einzelne Mitarbeiter oder Kunden, geringe finanzielle Auswirkungen oder geringer Reputationsschaden

2. Bewertung der Dringlichkeit: Definieren Sie Dringlichkeitsstufen danach, wie schnell auf ein Ereignis reagiert werden muss, und berücksichtigen Sie dabei Faktoren wie Sicherheit, finanzielle Auswirkungen und Service Level Agreements (SLAs). Das folgende Beispiel zeigt die Dringlichkeitskategorien:

Dringlichkeitsstufe	Beschreibung
Hoch	Exponentiell steigender Schaden, Beeinträchtigung zeitkritischer Aufgaben, drohende Eskalation oder betroffene VIP-Benutzer oder Gruppen
Mittel	Der Schaden nimmt im Laufe der Zeit zu oder es ist ein einzelner VIP-Benutzer oder eine Gruppe betroffen
Niedrig	Geringfügige Schadenszunahme im Laufe der Zeit oder nicht zeitkritische Arbeit beeinträchtigt

3. Erstellen einer Priorisierungsmatrix:

- Verwenden Sie eine Matrix, um Auswirkungen und Dringlichkeit miteinander zu vergleichen, und weisen Sie verschiedenen Kombinationen Prioritätsstufen zu.
- Machen Sie die Matrix allen Teammitgliedern, die für die Reaktion auf betriebliche Ereignisse verantwortlich sind, zugänglich und verständlich.
- Die folgende Beispielmatrix zeigt den Schweregrad eines Vorfalls nach Dringlichkeit und Auswirkung an:

Dringlichkeit und Auswirkungen	Hoch	Mittel	Niedrig
Hoch	Kritisch	Dringend	Hoch
Mittel	Dringend	Hoch	Normal
Niedrig	Hoch	Normal	Niedrig

4. Trainieren und Kommunizieren: Schulen Sie die Response-Teams im Umgang mit der Prioritätenmatrix und der Wichtigkeit, diese während eines Ereignisses zu befolgen. Kommunizieren Sie den Priorisierungsprozess an alle Stakeholder, um klare Erwartungen zu schaffen.
5. Integration der Vorfallreaktion:
 - Integrieren Sie die Priorisierungsmatrix in Ihre Pläne und Tools zur Reaktion auf Vorfälle.
 - Automatisieren Sie nach Möglichkeit die Klassifizierung und Priorisierung von Ereignissen, um die Reaktionszeiten zu verkürzen.
 - Kunden von Enterprise Support können [AWS-Vorfallerkennung und -reaktion](#) nutzen, wodurch eine proaktive Überwachung rund um die Uhr und ein Vorfalldmanagement für Produktions-Workloads ermöglicht wird.
6. Überprüfen und Anpassen: Überprüfen Sie regelmäßig die Effektivität des Priorisierungsprozesses und nehmen Sie Anpassungen auf der Grundlage von Rückmeldungen und Änderungen im Geschäftsumfeld vor.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS03-BP03 Eskalation wird gefördert](#)
- [OPS08-BP04 Erstellen umsetzbarer Warnmeldungen](#)
- [OPS09-BP01 Messen operativer Ziele und KPIs mit Metriken](#)

Zugehörige Dokumente:

- [Atlassian – Verständnis der Schweregrade von Vorfällen](#)

- [IT-Prozessplan – Checkliste der Vorfallpriorität](#)

OPS10-BP04 Definieren von Eskalationspfaden

Legen Sie in Ihren Protokollen zur Vorfallreaktion klare Eskalationspfade fest, um rechtzeitige und effektive Maßnahmen zu ermöglichen. Dazu gehören die Festlegung von Aufforderungen zur Eskalation, die detaillierte Beschreibung des Eskalationsprozesses und die vorherige Genehmigung von Maßnahmen, um die Entscheidungsfindung zu beschleunigen und die durchschnittliche Zeit für die Behebung zu verkürzen.

Gewünschtes Ergebnis: Ein strukturierter und effizienter Prozess, der Vorfälle an das entsprechende Personal weiterleitet und so die Reaktionszeiten und Auswirkungen minimiert.

Typische Anti-Muster:

- Mangelnde Klarheit über die Wiederherstellungsverfahren führt zu provisorischen Maßnahmen bei kritischen Vorfällen.
- Das Fehlen von definierten Berechtigungen und Zuständigkeiten führt zu Verzögerungen, wenn dringende Maßnahmen erforderlich sind.
- Stakeholder und Kunden werden nicht erwartungsgemäß informiert.
- Wichtige Entscheidungen verzögern sich.

Vorteile der Nutzung dieser bewährten Methode:

- Optimierte Reaktion auf Vorfälle durch vordefinierte Eskalationsverfahren.
- Reduzierte Ausfallzeiten durch vorab genehmigte Maßnahmen und klare Zuständigkeiten.
- Verbesserte Ressourcenzuweisung und Anpassung der Support-Ebene an den Schweregrad des Vorfalles.
- Verbesserte Kommunikation mit Stakeholdern und Kunden.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Richtig definierte Eskalationspfade sind entscheidend für eine schnelle Reaktion auf Vorfälle. AWS Systems Manager Incident Manager unterstützt die Einrichtung strukturierter Eskalations- und

Bereitschaftspläne, die die richtigen Mitarbeiter alarmieren, damit sie bei Vorfällen handlungsbereit sind.

Implementierungsschritte

1. Einrichtung von Eskalationsaufforderungen: Richten Sie [CloudWatch-Alarme](#) ein, um einen Vorfall in [AWS Systems Manager Incident Manager](#) zu erstellen.
2. Erstellen von Bereitschaftsplänen: Erstellen Sie [Bereitschaftspläne](#) in Incident Manager, die auf Ihre Eskalationspfade abgestimmt sind. Statten Sie das Bereitschaftspersonal mit den erforderlichen Berechtigungen und Tools aus, um schnell handeln zu können.
3. Detaillierte Eskalationsverfahren:
 - Legen Sie bestimmte Bedingungen fest, unter denen ein Vorfall eskaliert werden sollte.
 - Erstellen Sie [Eskalationspläne](#) in Incident Manager.
 - Eskalationskanäle sollten aus einem Ansprechpartner oder einem Bereitschaftsplan bestehen.
 - Definieren Sie die Rollen und Verantwortlichkeiten des Teams auf jeder Eskalationsstufe.
4. Genehmigung von Schadensbegrenzungsmaßnahmen im Voraus: Arbeiten Sie mit Entscheidungsträgern zusammen, um Maßnahmen für erwartete Szenarien vorab zu genehmigen. Verwenden Sie die in Incident Manager integrierten [Systems-Manager-Automation-Runbooks](#), um die Behebung von Vorfällen zu beschleunigen.
5. Angabe der Zuständigkeit: Identifizieren Sie eindeutig die internen Besitzer für jeden Schritt des Eskalationspfads.
6. Details zu Eskalationen mit Drittanbietern:
 - Dokumentieren Sie Service Level Agreements (SLAs) von Drittanbietern und richten Sie sie an internen Zielen aus.
 - Legen Sie klare Protokolle für die Lieferantenkommunikation bei Vorfällen fest.
 - Integrieren Sie Lieferantenkontakte in die Tools zum Vorfallmanagement, um direkten Zugriff zu erhalten.
 - Führen Sie regelmäßige Übungen durch, die Reaktionsszenarien von Drittanbietern beinhalten.
 - Sorgen Sie dafür, dass die Informationen zur Lieferanteneskalation gut dokumentiert und leicht zugänglich sind.
7. Trainieren und Testen von Eskalationsplänen: Schulen Sie Ihr Team im Eskalationsprozess und führen Sie regelmäßig Übungen zur Reaktion auf Vorfälle oder den Ernstfall durch. Kunden mit Enterprise Support können einen [Workshop zum Vorfallmanagement](#) anfordern.

8. Weitere Verbesserung: Überprüfen Sie regelmäßig die Wirksamkeit Ihrer Eskalationspfade. Aktualisieren Sie Ihre Prozesse auf Grundlage der Erkenntnisse aus den Nachuntersuchungen von Vorfällen und dem kontinuierlichen Feedback.

Aufwand für den Implementierungsplan: Mittel

Ressourcen

Zugehörige bewährte Methoden:

- [OPS08-BP04 Erstellen umsetzbarer Warnmeldungen](#)
- [OPS10-BP02 Implementieren eines Prozesses für jede Warnmeldung](#)
- [OPS11-BP02 Führen Sie eine Analyse nach dem Vorfall durch](#)

Zugehörige Dokumente:

- [AWS Systems Manager Incident Manager-Eskalationspläne](#)
- [Working with on-call schedules in Incident Manager](#)
- [Erstellen und Verwalten von Runbooks](#)
- [Temporäre erweiterte Zugriffsverwaltung mit AWS IAM Identity Center](#)
- [Atlassian – Eskalationsrichtlinien für effektives Vorfallmanagement](#)

OPS10-BP05 Definieren eines Kundenkommunikationsplans für Ereignisse, die sich auf den Service auswirken

Eine effektive Kommunikation bei Ereignissen, die sich auf den Service auswirken, ist entscheidend, um das Vertrauen und die Transparenz gegenüber den Kunden aufrechtzuerhalten. Ein klar definierter Kommunikationsplan hilft Ihrem Unternehmen, bei Vorfällen schnell und klar Informationen sowohl intern als auch extern auszutauschen.

Gewünschtes Ergebnis:

- Ein robuster Kommunikationsplan, der Kunden und Stakeholder bei Ereignissen, die sich auf den Service auswirken, effektiv informiert.
- Transparenz in der Kommunikation, um Vertrauen aufzubauen und Ängste der Kunden abzubauen.
- Minimierung der Auswirkungen von Ereignissen, die sich auf den Service in Bezug auf das Kundenerlebnis und den Geschäftsbetrieb auswirken.

Typische Anti-Muster:

- Eine unzureichende oder verzögerte Kommunikation führt zu Verwirrung und Unzufriedenheit der Kunden.
- Allzu technische oder vage Nachrichten vermitteln nicht die tatsächlichen Auswirkungen auf die Benutzer.
- Es gibt keine vordefinierte Kommunikationsstrategie, was zu inkonsistenten und reaktiven Nachrichten führt.

Vorteile der Nutzung dieser bewährten Methode:

- Mehr Vertrauen und Zufriedenheit bei den Kunden durch proaktive und klare Kommunikation.
- Entlastung der Support-Teams durch präventive Behandlung von Kundenanliegen.
- Verbesserte Fähigkeit, Vorfälle effektiv zu verwalten und zu bewältigen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Die Erstellung eines umfassenden Kommunikationsplans für Ereignisse, die sich auf den Service auswirken, umfasst mehrere Facetten, von der Auswahl der richtigen Kanäle bis hin zur Formulierung der Botschaft und des Tonfalls. Der Plan sollte anpassungsfähig und skalierbar sein und verschiedene Ausfallszenarien berücksichtigen.

Implementierungsschritte

1. Definieren von Rollen und Zuständigkeiten:

- Beauftragen Sie einen Hauptzuständigen für die Vorfallreaktion mit der Überwachung der Maßnahmen.
- Benennen Sie einen Kommunikationsmanager, der für die Koordination der gesamten externen und internen Kommunikation verantwortlich ist.
- Beziehen Sie den Support-Manager ein, um eine konsistente Kommunikation über Support-Tickets zu gewährleisten.

2. Identifizieren von Kommunikationskanälen: Wählen Sie Kanäle wie Arbeitsplatz-Chat, E-Mail, SMS, soziale Medien, In-App-Benachrichtigungen und Statusseiten aus. Diese Kanäle sollten robust und in der Lage sein, bei Ereignissen, die den Service beeinträchtigen, unabhängig zu arbeiten.

3. Schnelle, klare und regelmäßige Kommunikation mit Kunden:

- Entwickeln Sie Vorlagen für verschiedene Szenarien, bei denen Beeinträchtigungen des Serviceangebots vorliegen, und achten Sie dabei auf Einfachheit und wichtige Details. Fügen Sie Informationen über die Beeinträchtigung des Services, die erwartete Lösungszeit und die Auswirkungen hinzu.
- Verwenden Sie Amazon Pinpoint, um Kunden mithilfe von Push-Benachrichtigungen, In-App-Benachrichtigungen, E-Mails, Textnachrichten, Sprachnachrichten und Nachrichten über benutzerdefinierte Kanäle zu informieren.
- Verwenden Sie Amazon Simple Notification Service (Amazon SNS), um Subscriber programmgesteuert oder per E-Mail, mobilen Push-Benachrichtigungen und Textnachrichten zu benachrichtigen.
- Kommunizieren Sie den Status über Dashboards, indem Sie ein Amazon-CloudWatch-Dashboard öffentlich teilen.
- Förderung des Engagements in den sozialen Medien:
 - Verfolgen Sie aktiv die sozialen Medien, um die Stimmung der Kunden zu verstehen.
 - Posten Sie auf Social-Media-Plattformen, um die Öffentlichkeit auf dem Laufenden zu halten und die Community einzubeziehen.
 - Bereiten Sie Vorlagen für eine konsistente und klare Kommunikation in sozialen Medien vor.

4. Koordination der internen Kommunikation: Implementieren Sie interne Protokolle mithilfe von Tools wie Amazon Q Developer in Chat-Anwendungen für die Teamkoordination und Kommunikation. Verwenden Sie CloudWatch-Dashboards, um den Status zu kommunizieren.

5. Orchestrierung der Kommunikation mit speziellen Tools und Services:

- Verwenden Sie AWS Systems Manager Incident Manager mit Amazon Q Developer in Chat-Anwendungen, um spezielle Chat-Kanäle für die interne Kommunikation und Koordination in Echtzeit bei Vorfällen einzurichten.
- Verwenden Sie AWS Systems Manager Incident Manager-Runbooks, um Kundenbenachrichtigungen über Amazon Pinpoint, Amazon SNS oder Tools von Drittanbietern wie Social-Media-Plattformen bei Vorfällen zu automatisieren.
- Integrieren Sie Genehmigungs-Workflows in Runbooks, um optional die gesamte externe Kommunikation vor dem Versand zu überprüfen und zu autorisieren.

6. Praktizieren und verbessern:

- Führen Sie Trainingkurse zum Einsatz von Kommunikationsmitteln und -strategien durch. Ermöglichen Sie es Teams, bei Vorfällen rechtzeitig Entscheidungen zu treffen.

- Testen Sie den Kommunikationsplan durch regelmäßige Übungen oder Ernstfallübungen. Mithilfe dieser Tests können Sie Ihre Botschaften präzisieren und die Effektivität der Kanäle bewerten.
- Implementieren Sie Feedback-Mechanismen, um die Effektivität der Kommunikation bei Vorfällen zu bewerten. Entwickeln Sie den Kommunikationsplan auf Grundlage des Feedbacks und der sich ändernden Bedürfnisse kontinuierlich weiter.

Aufwand für den Implementierungsplan: Hoch

Ressourcen

Zugehörige bewährte Methoden:

- [OPS07-BP03 Verwenden von Runbooks zur Durchführung von Verfahren](#)
- [OPS10-BP06 Bekanntgeben des Status über Dashboards](#)
- [OPS11-BP02 Führen Sie eine Analyse nach dem Vorfall durch](#)

Zugehörige Dokumente:

- [Atlassian – Bewährte Methoden der Kommunikation bei Vorfällen](#)
- [Atlassian – Verfassen eines guten Status-Updates](#)
- [PagerDuty – Leitfaden für die Kommunikation bei Vorfällen](#)

Zugehörige Videos:

- [Atlassian – Erstellung eines eigenen Kommunikationsplans für Vorfälle: Vorlagen für Zwischenfälle](#)

Zugehörige Beispiele:

- [AWS Health-Dashboard](#)

OPS10-BP06 Bekanntgeben des Status über Dashboards

Verwenden Sie Dashboards als strategisches Werkzeug, um den Betriebsstatus und wichtige Metriken in Echtzeit an verschiedene Zielgruppen zu vermitteln, darunter interne technische Teams, Führungskräfte und Kunden. Diese Dashboards bieten eine zentrale, visuelle Darstellung des

Systemzustands und der Geschäftsleistung und erhöhen so die Transparenz und die Effizienz der Entscheidungsfindung.

Gewünschtes Ergebnis:

- Ihre Dashboards bieten einen umfassenden Überblick über das System und die Geschäftskennzahlen, die für verschiedene Stakeholder relevant sind.
- Stakeholder können proaktiv auf Betriebsinformationen zugreifen, sodass keine häufigen Statusanfragen mehr erforderlich sind.
- Die Entscheidungsfindung in Echtzeit wird während des normalen Betriebs und bei Vorfällen verbessert.

Typische Anti-Muster:

- Techniker, die an einem Vorfallmanagementgespräch teilnehmen, benötigen Statusaktualisierungen, um sich auf dem Laufenden zu halten.
- Sie verlassen sich auf die manuelle Berichterstattung für das Management, was zu Verzögerungen und möglichen Ungenauigkeiten führt.
- Die Arbeit der Operations-Teams wird bei Vorfällen häufig für Statusaktualisierungen unterbrochen.

Vorteile der Nutzung dieser bewährten Methode:

- Ermöglicht Stakeholdern den sofortigen Zugriff auf wichtige Informationen und fördert so fundierte Entscheidungen.
- Reduziert betriebliche Ineffizienzen, indem manuelle Berichte und häufige Statusabfragen minimiert werden.
- Erhöht die Transparenz und das Vertrauen durch Echtzeiteinblicke in die Systemleistung und Geschäftskennzahlen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Dashboards vermitteln effektiv den Status Ihres Systems und Ihrer Geschäftskennzahlen und können an die Anforderungen verschiedener Zielgruppen angepasst werden. Mit Tools wie Amazon-CloudWatch-Dashboards und Amazon Quick können Sie interaktive Echtzeit-Dashboards für die Systemüberwachung und Business Intelligence erstellen.

Implementierungsschritte

1. Ermittlung der Bedürfnisse der Stakeholder: Ermitteln Sie den spezifischen Informationsbedarf verschiedener Zielgruppen, z. B. technische Teams, Führungskräfte und Kunden.
2. Auswahl der richtigen Tools: Wählen Sie geeignete Tools wie [Amazon CloudWatch-Dashboards](#) für die Systemüberwachung und [Amazon Quick](#) für interaktive Business Intelligence aus. [AWS Health](#) stellt eine sofort einsatzbereite Erfahrung in der [AWS Health Dashboard](#) bereit. Sie können auch Integritätsereignisse in Amazon EventBridge oder über die AWS Health-API verwenden, um Ihre eigenen Dashboards zu erweitern.
3. Entwicklung effektiver Dashboards:
 - Entwickeln Sie Dashboards, um relevante Metriken und KPIs übersichtlich darzustellen und sicherzustellen, dass sie verständlich und umsetzbar sind.
 - Integrieren Sie bei Bedarf Ansichten auf System- und Unternehmensebene.
 - Inkludieren Sie sowohl Dashboards auf hoher Ebene (für umfassende Übersichten) als auch auf niedriger Ebene (für detaillierte Analysen).
 - Integrieren Sie automatische Alarmer in Dashboards, um kritische Probleme hervorzuheben.
 - Kommentieren Sie Dashboards mit wichtigen Schwellenwerten und Zielen für Metriken für sofortige Sichtbarkeit.
4. Integration von Datenquellen:
 - Verwenden Sie [Amazon CloudWatch](#), um Metriken aus verschiedenen AWS-Services zu aggregieren und anzuzeigen und [Metriken aus anderen Datenquellen abzufragen](#). So erhalten Sie eine einheitliche Ansicht der Integritäts- und Geschäftskennzahlen Ihres Systems.
 - Verwenden Sie Features wie [CloudWatch Logs Insights](#), um Protokolldaten aus verschiedenen Anwendungen und Services abzufragen und zu visualisieren.
 - Verwenden Sie AWS Health-Ereignisse, um über den operativen Status und bestätigte operative Probleme in AWS-Services über die [AWS Health-API](#) oder über [AWS Health-Ereignisse in Amazon EventBridge](#) informiert zu bleiben.
5. Bereitstellung von Selfservice-Zugriff:
 - Teilen Sie CloudWatch-Dashboards mit relevanten Stakeholdern für den Selfservicezugriff auf Informationen mithilfe von [Dashboard-Freigabe-Features](#).
 - Stellen Sie sicher, dass Dashboards leicht zugänglich sind und aktuelle Informationen in Echtzeit bereitstellen.
6. Regelmäßige Aktualisierungen und Verbesserungen:

- Aktualisieren und verbessern Sie die Dashboards kontinuierlich, um sie an die sich entwickelnden Geschäftsanforderungen und das Feedback der Stakeholder anzupassen.
- Überprüfen Sie die Dashboards regelmäßig, um sicherzustellen, dass sie relevant und effektiv sind, um die erforderlichen Informationen zu vermitteln.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS08-BP05 Erstellen von Dashboards](#)

Zugehörige Dokumente:

- [Erstellung von Dashboards für operative Sichtbarkeit](#)
- [Amazon-CloudWatch-Dashboards verwenden](#)
- [Flexible Dashboards mit Dashboard-Variablen erstellen](#)
- [CloudWatch-Dashboards freigeben](#)
- [Metriken aus anderen Datenquellen abfragen](#)
- [Hinzufügen eines benutzerdefinierten Widgets zu einem CloudWatch-Dashboard](#)

Zugehörige Beispiele:

- [Workshop zur Beobachtbarkeit – Dashboards](#)

OPS10-BP07 Automatisieren Sie Reaktionen auf Ereignisse

Die Automatisierung von Reaktionen auf Ereignisse ist der Schlüssel für eine schnelle, konsistente und fehlerfreie operative Abwicklung. Erstellen Sie optimierte Prozesse und verwenden Sie Tools, um Ereignisse automatisch zu verwalten und darauf zu reagieren, um manuelle Eingriffe zu minimieren und die betriebliche Effizienz zu steigern.

Gewünschtes Ergebnis:

- weniger menschliche Fehler und schnellere Lösungszeiten durch Automatisierung
- konsistente und zuverlässige Handhabung betrieblicher Ereignisse
- verbesserte betriebliche Effizienz und Systemzuverlässigkeit

Typische Anti-Muster:

- Die manuelle Behandlung von Ereignissen führt zu Verzögerungen und Fehlern.
- Bei sich wiederholenden, kritischen Aufgaben wird die Automatisierung übersehen.
- Sich wiederholende, manuelle Aufgaben führen zu Ermüdungserscheinungen und zum Übersehen kritischer Probleme.

Vorteile der Nutzung dieser bewährten Methode:

- beschleunigte Reaktionen auf Ereignisse, wodurch sich die Ausfallzeiten des Systems reduzieren
- zuverlässiger Betrieb mit automatisierter und konsistenter Ereignisbehandlung

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Integrieren Sie Automatisierung, um effiziente Arbeitsabläufe zu schaffen und manuelle Eingriffe zu minimieren.

Implementierungsschritte

1. Identifizieren von Möglichkeiten zur Automatisierung: Bestimmen Sie sich wiederholende Aufgaben für die Automatisierung, wie beispielsweise Problembehebung, Ticketverbesserung, Kapazitätsmanagement, Skalierung, Bereitstellung und Tests.
2. Identifizieren von Automatisierungsaufforderungen:
 - Beurteilen und definieren Sie spezifische Bedingungen oder Kennzahlen, die automatische Reaktionen mithilfe von [CloudWatch Amazon-Alarmaktionen auslösen](#).
 - Verwenden Sie [Amazon EventBridge](#), um auf Ereignisse in AWS Services, benutzerdefinierten Workloads und SaaS-Anwendungen zu reagieren.
 - Berücksichtigen Sie Initiierungsereignisse wie [bestimmte Protokolleinträge](#), [Schwellenwerte für Leistungskennzahlen](#) oder [Statusänderungen](#) AWS von Ressourcen.
3. Implementieren der ereignisgesteuerten Automatisierung:
 - Verwenden Sie AWS Systems Manager Automation-Runbooks, um Wartungs-, Bereitstellungs- und Problembehebungsaufgaben zu vereinfachen.
 - [Beim Erstellen von Vorfällen in Incident Manager](#) werden automatisch Details zu den beteiligten AWS Ressourcen gesammelt und dem Vorfall hinzugefügt.

- Überwachen Sie Kontingente proaktiv mit [Quota Monitor for AWS](#).
 - Passen Sie die Kapazität mit [AWS Auto Scaling](#) automatisch an, um Verfügbarkeit und Leistung aufrechtzuerhalten.
 - Automatisieren Sie Entwicklungspipelines mit [Amazon CodeCatalyst](#).
 - Testen Sie die Endgeräte oder überwachen Sie sie kontinuierlich und APIs [verwenden Sie synthetische](#) Überwachung.
4. Schadensbegrenzung durch Automatisierung:
- Implementieren Sie [automatisierte Sicherheitsmaßnahmen](#), um schnell auf Risiken zu reagieren.
 - Verwenden Sie [State Manager von AWS Systems Manager](#), um Konfigurationsabweichungen zu reduzieren.
 - [Korrigieren Sie Ressourcen, die nicht den Vorschriften entsprechen](#), mit AWS-Config-Regeln

Aufwand für den Implementierungsplan: Hoch

Ressourcen

Zugehörige bewährte Methoden:

- [OPS08-BP04 Erstellen umsetzbarer Warnmeldungen](#)
- [OPS10-BP02 Implementieren eines Prozesses für jede Warnmeldung](#)

Zugehörige Dokumente:

- [Verwendung von Systems-Manager-Automation-Runbooks mit Incident Manager](#)
- [Erstellen von Vorfällen in Incident Manager](#)
- [AWS Servicekontingenten](#)
- [Überwachen der Ressourcennutzung und Senden von Benachrichtigungen, wenn das Kontingent fast erreicht ist](#)
- [AWS Auto Scaling](#)
- [Was ist Amazon CodeCatalyst?](#)
- [CloudWatch Amazon-Alarme verwenden](#)
- [CloudWatch Amazon-Alarmaktionen verwenden](#)
- [Behebung nicht konformer Ressourcen mit AWS-Config-Regeln](#)

- [Erstellen von Metriken aus Protokollereignissen mithilfe von Filtern](#)
- [AWS Systems Manager State Manager](#)

Zugehörige Videos:

- [Erstellen Sie Automatisierungs-Runbooks mit AWS Systems Manager](#)
- [So automatisieren Sie den IT-Betrieb auf AWS](#)
- [AWS Security Hub CSPM Automatisierungsregeln](#)
- [Starten Sie Ihr Softwareprojekt schnell mit Amazon CodeCatalyst Blueprints](#)

Zugehörige Beispiele:

- [CodeCatalyst Amazon-Tutorial: Erstellen eines Projekts mit dem Blueprint für moderne dreistufige Webanwendungen](#)
- [Workshop zur Beobachtbarkeit](#)
- [Reaktion auf Vorfälle mit Incident Manager](#)

Weiterentwicklung

Frage

- [OPS 11. Wie können Sie Arbeitsvorgänge weiterentwickeln?](#)

OPS 11. Wie können Sie Arbeitsvorgänge weiterentwickeln?

Widmen Sie nahezu kontinuierlichen inkrementellen Verbesserungen Zeit und Ressourcen, um die Effektivität und Effizienz Ihrer Betriebsabläufe weiterzuentwickeln.

Best Practices

- [OPS11-BP01 Implementieren eines Prozesses für die kontinuierliche Verbesserung](#)
- [OPS11-BP02 Führen Sie eine Analyse nach dem Vorfall durch](#)
- [OPS11-BP03 Implementieren Sie Feedback-Schleifen](#)
- [OPS11-BP04 Wissensmanagement durchführen](#)
- [OPS11-BP05 Definieren von Verbesserungsfaktoren](#)

- [OPS11-BP06 Erkenntnisse validieren](#)
- [OPS11-BP07 Führen Sie Prüfungen der Betriebsmetriken durch](#)
- [OPS11-BP08 Dokumentieren und Weitergeben von Erkenntnissen](#)
- [OPS11-BP09 Nehmen Sie sich Zeit, um Verbesserungen vorzunehmen](#)

OPS11-BP01 Implementieren eines Prozesses für die kontinuierliche Verbesserung

Bewerten Sie Ihre Workload mithilfe bewährter Methoden für interne und externe Architekturen. Führen Sie häufige, bewusste Workload-Überprüfungen durch. Räumen Sie Verbesserungsmöglichkeiten in Ihrem Softwareentwicklungsplan Priorität ein.

Gewünschtes Ergebnis:

- Sie analysieren Ihre Workload regelmäßig anhand bewährter Methoden für die Architektur.
- Sie räumen den Features in Ihrem Softwareentwicklungsprozess die gleiche Priorität wie Verbesserungsmöglichkeiten ein.

Typische Anti-Muster:

- Sie haben seit der Bereitstellung Ihrer Workload vor einigen Jahren keine Architekturüberprüfung durchgeführt.
- Verbesserungsmöglichkeiten haben geringere Priorität. Im Vergleich zu neuen Features bleiben diese Möglichkeiten im Backlog.
- In der Organisation gibt es keinen Standard für die Umsetzung von Änderungen an bewährten Methoden.

Vorteile der Nutzung dieser bewährten Methode:

- Ihre Workload wird durch bewährte Methoden für die Architektur auf dem aktuellen Stand gehalten.
- Sie entwickeln Ihre Workload gezielt weiter.
- Sie können die bewährten Methoden der Organisation nutzen, um alle Workloads zu verbessern.
- Sie erzielen marginale Gewinne, deren kumulative Wirkung jedoch zu einer höheren Effizienz führen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Führen Sie regelmäßig eine Überprüfung der Architektur Ihrer Workload durch. Bewerten Sie anhand interner und externer bewährter Methoden Ihre Workload und ermitteln Sie Verbesserungsmöglichkeiten. Räumen Sie Verbesserungsmöglichkeiten in Ihrem Softwareentwicklungsplan Priorität ein.

Implementierungsschritte

1. Führen Sie in vereinbarten Intervallen Überprüfungen der Architektur Ihrer Produktions-Workloads durch. Verwenden Sie einen dokumentierten Architekturstandard mit AWS-spezifischen bewährten Methoden.
 - a. Verwenden Sie Ihre intern definierten Standards für diese Bewertungen. Wenn Sie nicht über einen internen Standard verfügen, verwenden Sie das AWS Well-Architected Framework.
 - b. Verwenden Sie AWS Well-Architected Tool, um einen Fokusbereich Ihrer internen bewährten Methoden zu erstellen und Ihre Architekturprüfung durchzuführen.
 - c. Wenden Sie sich an Ihren AWS Solution Architect oder Technical Account Manager, um einen geführten Well-Architected Framework Review Ihrer Workload durchzuführen.
2. Räumen Sie den während der Überprüfung ermittelten Verbesserungsmöglichkeiten in Ihrem Softwareentwicklungsprozess Priorität ein.

Aufwand für den Implementierungsplan: Niedrig. Sie können das AWS Well-Architected Framework zur Durchführung Ihrer jährlichen Architekturprüfung verwenden.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS11-BP02 Durchführen von Analysen nach Vorfällen](#)
- [OPS11-BP08 Dokumentieren und Weitergeben von Erkenntnissen](#)
- [OPS04 – Implementieren von Beobachtbarkeit](#)

Zugehörige Dokumente:

- [AWS Well-Architected Tool – Fokusbereiche](#)
- [AWS Well-Architected Whitepaper – The review process](#)
- [Customize Well-Architected Reviews using Custom Lenses and the AWS Well-Architected Tool](#)

- [Implementing the AWS Well-Architected Custom Lens lifecycle in your organization](#)

Zugehörige Videos:

- [AWS re:Invent 2023 – Scaling AWS Well-Architected best practices across your organization](#)

Zugehörige Beispiele:

- [AWS Well-Architected Tool](#)

OPS11-BP02 Führen Sie eine Analyse nach dem Vorfall durch

Überprüfen Sie die Ereignisse mit Auswirkungen auf Kunden und bestimmen Sie die beitragenden Faktoren und Präventivmaßnahmen. Entwickeln Sie anhand dieser Informationen Abhilfemaßnahmen, um Wiederholungen einzuschränken oder zu verhindern. Entwickeln Sie Verfahren für schnelle und effektive Reaktionen. Informieren Sie nach Bedarf auf zielgruppengerechte Weise über beitragende Faktoren und Korrekturmaßnahmen.

Gewünschtes Ergebnis:

- Sie haben Prozesse für das Vorfallmanagement eingerichtet, die auch Analysen nach dem Vorfall beinhalten.
- Sie verfügen über Pläne zur Beobachtbarkeit, um Daten über Ereignisse zu sammeln.
- Anhand dieser Daten können Sie Metriken verstehen und erfassen, die Sie bei der Analyse nach einem Vorfall unterstützen.
- Sie lernen aus Vorfällen, um zukünftige Ergebnisse zu verbessern.

Typische Anti-Muster:

- Sie verwalten einen Anwendungsserver. Ungefähr alle 23 Stunden und 55 Minuten werden alle Ihre aktiven Sitzungen beendet. Sie haben versucht, festzustellen, wo der Fehler auf Ihrem Anwendungsserver liegt. Sie vermuten, dass es sich um ein Netzwerkproblem handeln könnte, das Netzwerkteam zeigt sich jedoch unkooperativ, da es für Ihr Anliegen zu beschäftigt ist. Sie haben keinen vordefinierten Prozess, den Sie befolgen könnten, um Support zu erhalten und die nötigen Informationen zu sammeln, um dem Problem auf den Grund zu gehen.
- Bei Ihrer Workload kam es zu Datenverlust. Dies ist das erste Mal, dass dieses Problem aufgetreten ist, und die Ursache ist nicht klar. Sie entscheiden, dass es nicht wichtig ist, da Sie

die Daten wiederherstellen können. Datenverluste beginnen mit größerer Häufigkeit aufzutreten und wirken sich auf Ihre Kunden aus. Dadurch steigt auch der betriebliche Aufwand, wenn Sie die fehlenden Daten wiederherstellen.

Vorteile der Nutzung dieser bewährten Methode:

- Durch vordefinierte Prozesse zur Bestimmung der Komponenten, Bedingungen, Maßnahmen und Ereignisse, die zu einem Vorfall beigetragen haben, können Sie Verbesserungsmöglichkeiten ermitteln.
- Sie können Daten aus der Analyse nach einem Vorfall nutzen, um Verbesserungen vorzunehmen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Verwenden Sie einen Prozess zur Ermittlung der Faktoren, die dazu beitragen. Überprüfen Sie alle Vorfälle, die sich auf Kunden auswirken. Erarbeiten Sie ein Verfahren, um die beitragenden Faktoren eines Vorfalls zu ermitteln und zu dokumentieren. Damit können Sie Abhilfemaßnahmen entwickeln, um ein erneutes Auftreten einzudämmen oder gänzlich zu verhindern, und Verfahren für eine rasche und wirksame Reaktion erstellen. Informieren Sie gegebenenfalls über die Ursachen von Vorfällen und passen Sie die Kommunikation an Ihre Zielgruppe an. Teilen Sie Ihre Erkenntnisse offen innerhalb Ihrer Organisation mit.

Implementierungsschritte

1. Erfassen Sie Metriken wie Bereitstellungsänderungen, Konfigurationsänderungen, Startzeit des Vorfalls, Zeitpunkt des Alarms, Zeitpunkt des Einsatzes, Startzeit der Schadensbegrenzung und Zeitpunkt der Behebung des Vorfalls.
2. Beschreiben Sie wichtige Zeitpunkte auf der Zeitleiste, um die Ereignisse des Vorfalls zu verstehen.
3. Stellen Sie die folgenden Fragen:
 - a. Könnten Sie die Zeit bis zur Erkennung verkürzen?
 - b. Gibt es Aktualisierungen von Metriken und Alarmen, durch die der Vorfall früher erkannt würde?
 - c. Können Sie die Zeit bis zur Diagnose verkürzen?
 - d. Gibt es Aktualisierungen Ihrer Reaktions- oder Eskalationspläne, mit denen die richtigen Notfallteams früher eingeschaltet werden könnten?

- e. Können Sie die Zeit bis zur Schadensbegrenzung verkürzen?
 - f. Gibt es Runbook- oder Playbook-Schritte, die Sie hinzufügen oder verbessern könnten?
 - g. Können Sie zukünftige Vorfälle verhindern?
4. Erstellen Sie Checklisten und Aktionen. Verfolgen und führen Sie alle Aktionen durch.

Aufwand für den Implementierungsplan: Mittel

Ressourcen

Zugehörige bewährte Methoden:

- [OPS11-BP01 Implementieren eines Prozesses für die kontinuierliche Verbesserung](#)
- [OPS4 — Implementieren Sie Beobachtbarkeit](#)

Zugehörige Dokumente:

- [Durchführen einer Analyse nach einem Vorfall im Incident Manager](#)
- [Überprüfung der Betriebsbereitschaft](#)

OPS11-BP03 Implementieren Sie Feedback-Schleifen

Feedbackschleifen bieten umsetzbare Erkenntnisse zur Unterstützung der Entscheidungsfindung. Integrieren Sie Feedbackschleifen in Ihre Verfahren und Workloads. Damit können Sie Probleme und Bereiche identifizieren, für die Verbesserungen erforderlich sind. Diese validieren auch Investitionen für Verbesserungen. Diese Feedbackschleifen sind die Grundlage für die kontinuierliche Verbesserung Ihrer Workload.

Feedbackschleifen können in zwei Kategorien unterteilt werden: sofortiges Feedback und nachträgliche Analysen. Sofortiges Feedback wird durch Prüfung der Leistung und der Ergebnisse betrieblicher Aktivitäten eingeholt. Dieses Feedback kommt von Teammitgliedern, Kunden oder der automatisierten Ausgabe der Aktivität. Sofortiges Feedback kommt von Dingen wie A/B-Tests und der Auslieferung neuer Features und ist für das „Schnell scheitern“-Konzept von entscheidender Bedeutung.

Nachträgliche Analysen werden regelmäßig durchgeführt, um Feedback aus der Überprüfung betrieblicher Ergebnisse und Metriken in der Vergangenheit zu erhalten. Dies geschieht am Ende einer Phase, in regelmäßigem Rhythmus oder nach größeren Releases oder Veranstaltungen. Diese

Art von Feedbackschleife validiert Investitionen in Betriebsabläufe oder Ihre Workload. Dies hilft Ihnen beim Messen des Erfolgs und bei der Validierung Ihrer Strategie.

Gewünschtes Ergebnis: Sie nutzen sofortiges Feedback und nachträgliche Analysen für weitere Verbesserungen. Es gibt einen Mechanismus zur Erfassung des Feedbacks von Benutzern und Teammitgliedern. Nachträgliche Analysen identifizieren Trends, die Verbesserungen unterstützen können.

Typische Anti-Muster:

- Sie starten ein neues Feature, haben aber keine Möglichkeit, Feedback von den Kunden dazu zu erhalten.
- Nach einer Investition in verbesserte Betriebsabläufe führen Sie keine nachträgliche Analyse für deren Validierung durch.
- Sie holen das Feedback von Kunden ein, überprüfen dies jedoch nicht regelmäßig.
- Feedbackschleifen führen zu vorgeschlagenen Maßnahmen, werden jedoch nicht in den Softwareentwicklungsprozess einbezogen.
- Kunden erhalten kein Feedback zu Verbesserungen, die sie vorgeschlagen haben.

Vorteile der Nutzung dieser bewährten Methode:

- Sie können vom Kunden aus rückwärts arbeiten, um neue Features zu unterstützen.
- Ihre Organisationskultur kann schneller auf Änderungen reagieren.
- Trends dienen zur Identifizierung von Verbesserungsmöglichkeiten.
- Nachträgliche Analysen validieren in Ihre Workloads und Betriebsabläufe getätigte Investitionen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Die Implementierung dieser bewährten Methode bedeutet, dass Sie sofortiges Feedback und nachträgliche Analysen verwenden. Diese Feedbackschleifen erleichtern Verbesserungen. Es gibt zahlreiche Mechanismen für sofortiges Feedback, z. B. Umfragen, Kundenbefragungen oder Feedbackformulare. Ihre Organisation nutzt nachträgliche Analysen auch, um Möglichkeiten für Verbesserungen zu identifizieren und Initiativen zu validieren.

Kundenbeispiel

AnyCompany Retail hat ein Webformular erstellt, über das Kunden Feedback geben oder Probleme melden können. Bei der wöchentlichen Scrum-Sitzung evaluiert das Softwareentwicklungsteam das Benutzerfeedback. Das Feedback wird regelmäßig genutzt, um die Weiterentwicklung der Plattform zu steuern. Am Ende jeder Etappe wird eine nachträgliche Analyse durchgeführt, um Punkte zu identifizieren, bei denen Verbesserungsbedarf besteht.

Implementierungsschritte

1. Sofortiges Feedback

- Sie benötigen einen Mechanismus für den Erhalt von Feedback von Kunden und Teammitgliedern. Ihre betrieblichen Aktivitäten können auch so konfiguriert werden, dass Sie automatisiertes Feedback erhalten.
- Ihre Organisation benötigt einen Prozess zur Prüfung dieses Feedbacks, zum Feststellen der Verbesserungsbereiche und zur Planung der Verbesserungen.
- Das Feedback muss in Ihren Softwareentwicklungsprozess integriert werden.
- Wenn Sie Verbesserungen durchführen, informieren Sie die Personen, die dazu Feedback gegeben haben.
 - Sie können [AWS Systems Manager OpsCenter](#) verwenden, um diese Verbesserungen zu erstellen und nachzuverfolgen als [OpsItems](#).

2. Nachträgliche Analyse

- Führen Sie nachträgliche Analysen am Ende eines Entwicklungszyklus, in regelmäßigen Abständen oder nach einem größeren Release durch.
- Laden Sie an der Workload beteiligte Stakeholder zu einer Nachbesprechung ein.
- Erstellen Sie auf einem Whiteboard oder in einem Spreadsheet drei Spalten: Beenden, Starten und Beibehalten.
 - Beenden gilt für alles, mit dem Ihr Team aufhören soll.
 - Starten gilt für Ideen, die ab sofort umgesetzt werden sollen.
 - Beibehalten gilt für Elemente, die weiterhin durchgeführt werden sollen.
- Holen Sie das Feedback aller anwesenden Stakeholder ein.
- Priorisieren Sie das Feedback. Weisen Sie allen „Starten“- oder „Beibehalten“-Elementen Aktionen und Stakeholder zu.
- Fügen Sie die Aktionen Ihrem Softwareentwicklungsprozess hinzu und halten Sie die Stakeholder bei Ihren Verbesserungen über den Status auf dem Laufenden.

Aufwand für den Implementierungsplan: Mittel. Zur Implementierung dieser bewährten Methode benötigen Sie ein Verfahren zum Einholen und zur Analyse sofortigen Feedbacks. Dazu müssen Sie auch einen Prozess für die nachträgliche Analyse einrichten.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS01-BP01 Bewerten der Bedürfnisse externer Kunden](#): Feedbackschleifen sind ein Mechanismus zum Ermitteln der Anforderungen externer Kunden.
- [OPS01-BP02 Evaluieren Sie die internen Kundenbedürfnisse](#): Interne Stakeholder können Feedbackschleifen nutzen, um Bedürfnisse und Anforderungen zu kommunizieren.
- [OPS11-BP02 Führen Sie eine Analyse nach dem Vorfall durch](#): Analysen nach einem Vorfall sind eine wichtige Form nachträglicher Analyse nach Vorfällen.
- [OPS11-BP07 Führen Sie Prüfungen der Betriebsmetriken durch](#): Durch die Prüfung betrieblicher Metriken können Sie Trends und Bereiche für Verbesserungen identifizieren.

Zugehörige Dokumente:

- [7 Fallstricke, die Sie beim Bau eines vermeiden sollten CCOE](#)
- [Atlassian Team Playbook – Retrospectives](#)
- [E-Mail-Definitionen: Feedbackschleifen](#)
- [Etablierung von Feedback-Schleifen auf der Grundlage des AWS Well-Architected Framework Review](#)
- [IBMGarage Methodology — Halten Sie einen Rückblick](#)
- [Investopedia — Der Zyklus PDCS](#)
- [Maximizing Developer Effectiveness von Tim Cochran](#)
- [Bewertungen der Betriebsbereitschaft \(ORR\) Whitepaper — Iteration](#)
- [ITILCSI- Kontinuierliche Serviceverbesserung](#)
- [Toyota und E-Commerce: Lean bei Amazon](#)

Zugehörige Videos:

- [Aufbau effektiver Kundenfeedbackschleifen](#)

Zugehörige Beispiele:

- [Astuto – Open-Source-Tool für Kundenfeedback](#)
- [AWS Lösungen — Q nABot auf AWS](#)
- [Fider – Eine Plattform zur Organisation von Kundenfeedback](#)

Zugehörige Services:

- [AWS Systems Manager OpsCenter](#)

OPS11-BP04 Wissensmanagement durchführen

Durch ein Wissensmanagement erhalten Teammitglieder die Informationen, die sie für ihre Arbeit benötigen. In lernenden Organisationen werden Informationen frei geteilt, was jedem Einzelnen die nötigen Kompetenzen eröffnet. Die Informationen können entdeckt oder gesucht werden. Die Informationen sind korrekt und aktuell. Es gibt Mechanismen, um neue Informationen zu erstellen, bestehende Informationen zu aktualisieren und veraltete Informationen zu archivieren. Das gängigste Beispiel für eine Wissensmanagement-Plattform ist ein Content-Management-System wie ein Wiki.

Gewünschtes Ergebnis:

- Teammitglieder haben Zugriff auf zeitnahe, präzise Informationen.
- Die Informationen sind durchsuchbar.
- Es gibt Mechanismen zum Hinzufügen, Aktualisieren und Archivieren von Informationen.

Typische Anti-Muster:

- Es gibt keinen zentralen Wissensspeicher. Die Teammitglieder verwalten ihre eigenen Notizen auf ihren lokalen Rechnern.
- Sie haben ein selbst gehostetes Wiki, aber keine Mechanismen zum Verwalten von Informationen, was dazu führt, dass die Informationen veraltet sind.
- Jemand stellt fest, dass Informationen fehlen, aber es gibt keinen Prozess, um das Hinzufügen dieser Informationen zum Team-Wiki anzustoßen. Er fügt sie selbst hinzu, aber versäumt einen wichtigen Schritt, was zu einem Ausfall führt.

Vorteile der Nutzung dieser bewährten Methode:

- Die Teammitglieder werden gestärkt, weil Informationen frei geteilt werden.
- Neue Teammitglieder werden schneller eingearbeitet, weil die Dokumentation aktuell und durchsuchbar ist.
- Die Informationen sind zeitnah, präzise und umsetzbar.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Das Wissensmanagement ist eine wichtige Facette von lernenden Organisationen. Zunächst benötigen Sie ein zentrales Repository, in dem Sie Ihr Wissen speichern (z. B. ein selbst gehostetes Wiki). Sie müssen Prozesse entwickeln, um Wissen hinzuzufügen, zu aktualisieren und zu archivieren. Entwickeln Sie Standards für das, was dokumentiert werden soll, und lassen Sie alle Beteiligten dazu beitragen.

Kundenbeispiel

AnyCompany Der Einzelhandel hostet ein internes Wiki, in dem das gesamte Wissen gespeichert ist. Die Teammitglieder werden ermutigt, die Wissensdatenbank im Rahmen ihrer täglichen Arbeit zu ergänzen. Ein funktionsübergreifendes Team bewertet vierteljährlich, welche Seiten am wenigsten aktualisiert werden, und entscheidet, ob sie archiviert oder aktualisiert werden sollen.

Implementierungsschritte

1. Beginnen Sie damit, das Content-Management-System zu bestimmen, in dem das Wissen gespeichert werden soll. Holen Sie die Zustimmung der Stakeholder in Ihrer Organisation ein.
 - a. Wenn Sie kein vorhandenes Content-Management-System haben, können Sie ein selbst gehostetes Wiki oder ein Versionsverwaltungssystem als Ausgangspunkt verwenden.
2. Entwickeln Sie Runbooks für das Hinzufügen, Aktualisieren und Archivieren von Informationen. Informieren Sie Ihr Team über diese Prozesse.
3. Bestimmen Sie, welches Wissen im Content-Management-System gespeichert werden soll. Beginnen Sie mit den täglichen Aktivitäten (Runbooks und Playbooks), die die Teammitglieder ausführen. Arbeiten Sie mit Stakeholdern zusammen, um Prioritäten für das hinzuzufügende Wissen festzulegen.
4. Arbeiten Sie regelmäßig mit Interessengruppen zusammen, um out-of-date Informationen zu identifizieren und zu archivieren oder auf den neuesten Stand zu bringen.

Aufwand für den Implementierungsplan: Mittel. Wenn Sie kein vorhandenes Content-Management-System haben, können Sie ein selbst gehostetes Wiki oder ein Dokumenten-Repository mit Versionsverwaltung einrichten.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS11-BP08 Dokumentieren und Weitergeben von Erkenntnissen](#) – Das Wissensmanagement erleichtert den Austausch von Informationen über gewonnene Erkenntnisse.

Zugehörige Dokumente:

- [Atlassian - Knowledge Management](#)

Zugehörige Beispiele:

- [DokuWiki](#)
- [Gollum](#)
- [MediaWiki](#)
- [Wiki.js](#)

OPS11-BP05 Definieren von Verbesserungsfaktoren

Identifizieren Sie Verbesserungsmöglichkeiten, damit Sie Chancen basierend auf Daten und Feedbackschleifen bewerten und priorisieren können. Erkunden Sie Verbesserungsmöglichkeiten in Ihren Systemen und Prozessen und automatisieren Sie bei Bedarf.

Gewünschtes Ergebnis:

- Sie verfolgen Daten aus Ihrer gesamten Umgebung.
- Sie korrelieren Ereignisse und Aktivitäten mit Geschäftsergebnissen.
- Sie können Umgebungen und Systeme vergleichen und gegenüberstellen.
- Sie führen einen detaillierten Aktivitätsverlauf Ihrer Bereitstellungen und Ergebnisse.
- Sie sammeln Daten, um Ihren Sicherheitsstatus zu stärken.

Typische Anti-Muster:

- Sie sammeln Daten aus Ihrer gesamten Umgebung, korrelieren jedoch keine Ereignisse und Aktivitäten.
- Sie sammeln detaillierte Daten aus Ihrem gesamten Bestand, was die Aktivität und Kosten von Amazon CloudWatch und AWS CloudTrail in die Höhe treibt. Sie ziehen jedoch keinen sinnvollen Nutzen aus diesen Daten.
- Bei der Definition von Verbesserungsfaktoren berücksichtigen Sie nicht die Geschäftsergebnisse.
- Sie messen nicht die Auswirkungen neuer Features.

Vorteile der Nutzung dieser bewährten Methode:

- Sie minimieren die Auswirkungen ereignisbasierter Motivationen oder emotionaler Investitionen, indem Sie Verbesserungskriterien festlegen.
- Sie reagieren auf alle, nicht nur technische Geschäftsereignisse.
- Sie messen Ihre Umgebung, um Verbesserungsbereiche zu identifizieren.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

- Kenntnis der Verbesserungsfaktoren: Sie sollten ein System nur dann ändern, wenn das gewünschte Ergebnis auch unterstützt wird.
 - Gewünschte Fähigkeiten: Prüfen Sie bei der Bewertung von Verbesserungsmöglichkeiten die gewünschten Features und Fähigkeiten.
 - [Neuerungen bei AWS](#)
 - Nicht akzeptable Probleme: Prüfen Sie bei der Bewertung von Verbesserungsmöglichkeiten nicht akzeptable Probleme, Fehler und Schwachstellen. Informieren Sie sich über Dimensionierungsoptionen und suchen Sie nach Optimierungsmöglichkeiten.
 - [AWS Latest Security Bulletins](#)
 - [AWS Trusted Advisor](#)
 - [Cloud Intelligence Dashboards](#)
 - Complianceanforderungen: Prüfen Sie bei der Bewertung von Verbesserungsmöglichkeiten, welche Updates und Änderungen erforderlich sind, um Vorschriften bzw. Richtlinien einzuhalten oder weiterhin den Support eines Drittanbieters nutzen zu können.
 - [AWS Compliance](#)

- [AWS Compliance Programs](#)
- [AWS Compliance Latest News](#)

Ressourcen

Zugehörige bewährte Methoden:

- [OPS01 Organisationsprioritäten](#)
- [OPS02 Beziehungen und Eigentümerschaft](#)
- [OPS04-BP01 Ermitteln wichtiger Leistungskennzahlen](#)
- [OPS08 Nutzung der Workload-Beobachtbarkeit](#)
- [OPS09 Grundlegendes zum betrieblichen Status](#)
- [OPS11-BP03 Implementieren von Feedbackschleifen](#)

Zugehörige Dokumente:

- [Amazon Athena](#)
- [Schnell](#)
- [AWS Compliance](#)
- [AWS Compliance Latest News](#)
- [AWS Compliance Programs](#)
- [AWS Glue](#)
- [AWS Latest Security Bulletins](#)
- [AWS Trusted Advisor](#)
- [Exportieren von Protokolldaten nach Amazon S3](#)
- [Neuerungen bei AWS](#)
- [The Imperatives of Customer-Centric Innovation](#)
- [Digital Transformation: Hype or a Strategic Necessity?](#)

Zugehörige Videos

- [AWS re:Invent 2023 – Improve operational efficiency and resilience with Support \(SUP310\)](#)

OPS11-BP06 Erkenntnisse validieren

Überprüfen Sie Ihre Analyseergebnisse und Reaktionen mit fachbereichsübergreifenden Teams und Geschäftsverantwortlichen. Schaffen Sie mithilfe dieser Prüfungen ein allgemeines Verständnis, ermitteln Sie weitere Auswirkungen und legen Sie einen Maßnahmenkatalog fest. Passen Sie die Reaktionen bei Bedarf an.

Gewünschte Ergebnisse:

- Sie überprüfen regelmäßig Erkenntnisse mit Geschäftsbereichsleitern. Geschäftsinhaber bieten zusätzlichen Kontext für neu gewonnene Erkenntnisse.
- Sie überprüfen Erkenntnisse und bitten um Feedback von Fachkollegen, und Sie teilen Ihre Erkenntnisse mit allen Teams.
- Sie veröffentlichen Daten und Erkenntnisse, die andere technische und Geschäftsteams überprüfen können. Sie entwickeln aus Ihren Erkenntnisse neue Methoden für andere Abteilungen.
- Sie fassen neue Erkenntnisse zusammen und besprechen sie mit Führungskräften. Führungskräfte nutzen neue Erkenntnisse, um die Strategie zu definieren.

Typische Anti-Muster:

- Sie veröffentlichen ein neues Feature. Dieses Feature verändert das Verhalten einiger Ihrer Kunden. Ihre Beobachtbarkeit berücksichtigt diese Änderungen nicht. Sie quantifizieren die Vorteile dieser Änderungen nicht.
- Sie veröffentlichen ein neues Update und vernachlässigen es, Ihr Update zu aktualisieren. CDN Der CDN Cache ist nicht mehr mit der neuesten Version kompatibel. Sie messen den Prozentsatz der Anforderungen mit Fehlern. Alle Ihre Benutzer melden HTTP 400 Fehler bei der Kommunikation mit Backend-Servern. Sie untersuchen die Kundenfehler und stellen fest, dass Sie die Zeit verschwendet haben, weil Sie die falsche Dimension gemessen haben.
- Ihr Service Level Agreement sieht eine Verfügbarkeit von 99,9 % vor und Ihr Wiederherstellungszeitpunkt liegt bei vier Stunden. Der Servicebesitzer behauptet, dass das System keine Ausfallzeiten hat. Sie implementieren eine teure und komplexe Replikationslösung, die Zeit und Geld verschwendet.

Vorteile der Nutzung dieser bewährten Methode:

- Durch die Prüfung von Erkenntnissen zusammen mit Geschäftsinhabern und Fachexperten bauen Sie ein gemeinsames Verständnis auf und sorgen effektiver für Verbesserungen.

- Sie entdecken verborgene Probleme und berücksichtigen sie bei zukünftigen Entscheidungen.
- Ihr Fokus verlagert sich von technischen Ergebnissen hin zu Geschäftsergebnissen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

- Prüfen von Erkenntnissen: Wenden Sie sich an die Geschäftsinhaber und Fachexperten, um sicherzustellen, dass die Bedeutung der von Ihnen gesammelten Daten allgemein verstanden und vereinbart ist. Ermitteln Sie zusätzliche Bedenken, potenzielle Auswirkungen und bestimmen Sie eine Vorgehensweise.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS01-BP06 Bewerten Sie Kompromisse und managen Sie gleichzeitig die Vorteile und Risiken](#)
- [OPS02-BP06 Die Verantwortlichkeiten zwischen den Teams sind vordefiniert oder werden ausgehandelt](#)
- [OPS11-BP03 Implementieren Sie Feedback-Schleifen](#)

Zugehörige Dokumente:

- [Gestaltung eines Cloud-Exzellenzzentrums \(\) CCOE](#)

Zugehörige Videos:

- [Building observability to increase resiliency](#)

OPS11-BP07 Führen Sie Prüfungen der Betriebsmetriken durch

Führen Sie regelmäßig teamübergreifend mit Teilnehmern aus verschiedenen Unternehmensbereichen nachträgliche Analysen der operationsspezifischen Metriken durch. Ermitteln Sie mithilfe dieser Prüfungen Verbesserungspotenziale sowie mögliche Maßnahmen und teilen Sie diese Erkenntnisse auch anderen mit. Berücksichtigen Sie bei Ihrer Suche nach Verbesserungsmöglichkeiten all Ihre Umgebungen (z. B. Entwicklungs-, Test- und Produktionsumgebung).

Gewünschtes Ergebnis:

- Sie überprüfen häufig Metriken, die sich auf das Geschäft auswirken.
- Sie erkennen und überprüfen Anomalien mithilfe Ihrer Beobachtbarkeitsfunktionen.
- Sie verwenden Daten, um die Erreichung von Geschäftsergebnissen und Zielen zu unterstützen.

Typische Anti-Muster:

- Ihr Wartungsfenster unterbricht eine wichtige Verkaufsaktion. Das Unternehmen weiß weiterhin nicht, dass es ein Standard-Wartungsfenster gibt, das verzögert werden könnte, wenn sich andere wichtige Ereignisse auf das Geschäft auswirken.
- Sie hatten einen längeren Ausfall, weil in Ihrer Organisation häufig eine veraltete Bibliothek verwendet wird. Inzwischen sind Sie zu einer unterstützten Bibliothek migriert. Die anderen Teams in Ihrer Organisation wissen nicht, dass diese Gefahr besteht.
- Sie überprüfen den Kundenerfolg nicht regelmäßig. SLAs Sie tendieren dazu, Ihren Kunden nicht zu treffen. SLAs Wenn Sie Ihren Kunden SLAs nicht treffen, drohen finanzielle Sanktionen.

Vorteile der Nutzung dieser bewährten Methode:

- Indem Sie sich regelmäßig treffen, um Betriebsmetriken, Ereignisse und Vorfälle zu überprüfen, sorgen Sie für ein gemeinsames Verständnis aller Teams.
- Ihr Team trifft sich regelmäßig, um Kennzahlen und Vorfälle zu überprüfen. Auf diese Weise können Sie Maßnahmen gegen Risiken ergreifen und Kunden SLAs erkennen.
- Sie teilen Ihre gewonnenen Erkenntnisse, die Daten zur Priorisierung und zur gezielten Verbesserung der Geschäftsergebnisse liefern.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

- Führen Sie regelmäßig teamübergreifend mit Teilnehmern aus verschiedenen Unternehmensbereichen nachträgliche Analysen der operationsspezifischen Metriken durch.
- Binden Sie alle Stakeholder, einschließlich der Teams aus den Bereichen Betriebswirtschaft, Entwicklung und Operationen, ein, indem Sie Ihre Erkenntnisse aus dem sofortigen Feedback und der nachträglichen Analyse und gewonnene Erkenntnisse austauschen.

- Machen Sie sich deren Erkenntnisse zunutze, um Verbesserungspotenziale und mögliche Maßnahmen ausfindig zu machen.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS08-BP05 Dashboards erstellen](#)
- [OPS09-BP03 Überprüfen Sie die Betriebskennzahlen und priorisieren Sie Verbesserungen](#)
- [OPS10-BP01 Verwenden Sie einen Prozess für das Ereignis-, Vorfall- und Problemmanagement](#)

Zugehörige Dokumente:

- [Amazon CloudWatch](#)
- [Referenz zu CloudWatch Amazon-Kennzahlen und -Dimensionen](#)
- [Veröffentlichen von benutzerdefinierten Metriken](#)
- [Verwenden von CloudWatch Amazon-Metriken](#)
- [Dashboards und Visualisierungen mit CloudWatch](#)

OPS11-BP08 Dokumentieren und Weitergeben von Erkenntnissen

Dokumentieren Sie die Erkenntnisse aus den betrieblichen Aktivitäten und geben Sie diese weiter, damit Sie sie sowohl intern als auch teamübergreifend nutzen können. Die Erkenntnisse Ihres Teams sollten Sie an andere in Ihrer Organisation weitergeben, damit alle davon profitieren. Teilen Sie Informationen und Ressourcen, um vermeidbare Fehler zu verhindern und Entwicklungsbemühungen zu unterstützen, und konzentrieren Sie sich auf die Bereitstellung der angestrebten Features.

Definieren Sie mithilfe von AWS Identity and Access Management (IAM) Berechtigungen, die den gesteuerten Zugriff auf die Ressourcen ermöglichen, die Sie innerhalb von Konten und kontenübergreifend freigeben möchten.

Gewünschtes Ergebnis:

- Anschließend sollten Sie versionsgesteuerte Repositories verwenden, um Anwendungsbibliotheken, skriptbasierte Verfahren, Verfahrens- und andere Systemdokumentationen freizugeben.
- Sie teilen Ihre Infrastrukturstandards als versionskontrollierte AWS CloudFormation-Vorlagen.

- Sie überprüfen die Erkenntnisse, die Sie teamübergreifend gelernt haben.

Typische Anti-Muster:

- Sie erlitten einen längeren Ausfall, weil Ihre Organisation häufig eine fehlerhafte Bibliothek verwendet. Seitdem sind Sie zu einer zuverlässigen Bibliothek migriert. Die anderen Teams in Ihrer Organisation wissen nicht, dass diese Gefahr besteht. Niemand dokumentiert und teilt die Erfahrung mit dieser Bibliothek und sie sind sich des Risikos nicht bewusst.
- Sie haben einen Grenzfall in einem intern gemeinsam genutzten Microservice ermittelt, der dazu führt, dass Sitzungen unterbrochen werden. Sie rufen den Service jetzt anders auf, um diesen Grenzfall zu vermeiden. Die anderen Teams in Ihrer Organisation wissen nicht, dass diese Gefahr besteht.
- Sie haben eine Möglichkeit gefunden, die Anforderungen an die CPU-Auslastung eines Ihrer Microservices deutlich zu reduzieren. Sie wissen nicht, ob andere Teams auch von diesem Verfahren profitieren könnten.

Vorteile der Einführung dieser bewährten Methode: Teilen Sie die Erkenntnisse, um Verbesserungen zu unterstützen und erfahrungsbasierte Vorteile zu maximieren.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Niedrig

Implementierungsleitfaden

- Dokumentieren und Weitergeben von Erkenntnissen: Implementieren Sie Verfahren zur Dokumentation der aus der Durchführung von betrieblichen Aktivitäten und nachträglichen Analysen gewonnenen Erkenntnisse, damit auch andere Teams davon profitieren.
- Weitergeben von Erkenntnissen: Nutzen Sie Verfahren für den teamübergreifenden Austausch gewonnener Erkenntnisse und zugehöriger Artefakte. Veröffentlichen Sie beispielsweise aktualisierte Verfahren, Richtlinien, Governance und bewährte Methoden in einem allgemein zugänglichen Wiki. Teilen Sie Skripte, Code und Bibliotheken über ein gemeinsames Repository.
 - Nutzen Sie [AWS re:PostPrivate](#) als Wissensservice, um die Zusammenarbeit und den Wissensaustausch innerhalb Ihres Unternehmens zu optimieren.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS02-BP06 Zuständigkeiten zwischen Teams werden vordefiniert oder ausgehandelt](#)
- [OPS05-BP01 Verwendung einer Versionskontrolle](#)
- [OPS05-BP06 Gemeinsame Design-Standards](#)
- [OPS11-BP03 Implementieren von Feedbackschleifen](#)
- [OPS11-BP07 Prüfung von Betriebsmetriken](#)

Zugehörige Dokumente:

- [Verbessern Sie die Zusammenarbeit, und teilen Sie Cloud-Wissen auf sichere Weise mit AWS re:Post Private](#)
- [Reduzieren Sie Projektverzögerungen mit einer Docs-as-a-Code-Lösung](#)

Zugehörige Videos:

- [AWS re:Invent 2023 – Collaborate within your company and with AWS using AWS re:Post Private](#)
- [Supports You | Exploring the Incident Management Tabletop Exercise](#)

OPS11-BP09 Nehmen Sie sich Zeit, um Verbesserungen vorzunehmen

Reservieren Sie Zeit und Ressourcen innerhalb Ihrer Prozesse, um kontinuierliche, schrittweise Verbesserungen zu ermöglichen.

Gewünschtes Ergebnis:

- Sie können temporäre Duplikate von Umgebungen erstellen. Das senkt die Risiken, den Aufwand und Kosten, die mit dem Experimentieren und Testen verbunden sind.
- Diese duplizierten Umgebungen können Sie nutzen, um die aus Ihren Analysen gezogenen Rückschlüsse zu testen, Verbesserungen zu entwickeln und geplante Verbesserungen zu testen.
- Sie veranstalten Spieltage und verwenden den Fault Injection Service (FIS), um die Kontrollen und Leitplanken bereitzustellen, die Teams für die Durchführung von Experimenten in einer Produktionsumgebung benötigen.

Typische Anti-Muster:

- Es besteht ein bekanntes Leistungsproblem auf Ihrem Anwendungsserver. Es wird im Backlog hinter jeder geplanten Feature-Implementierung priorisiert. Bleibt die Rate der hinzugefügten geplanten Features konstant, wird das Leistungsproblem niemals behoben.
- Genehmigen Sie den Administratoren und Entwicklern, dass sie ihre Überstunden zur Auswahl und Implementierung von Verbesserungen nutzen können, um kontinuierliche Verbesserungen zu unterstützen. Es werden niemals Verbesserungen vorgenommen.
- Die Betriebsabnahme ist abgeschlossen und Sie testen die betrieblichen Praktiken nicht erneut.

Vorteile der Einführung dieser bewährten Methode: Indem Sie Zeit und Ressourcen innerhalb Ihrer Prozesse reservieren, können Sie kontinuierliche, schrittweise Verbesserungen ermöglichen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Niedrig

Implementierungsleitfaden

- Einplanen von Zeit für Verbesserungen: Reservieren Sie Zeit und Ressourcen innerhalb Ihrer Prozesse, um kontinuierliche, schrittweise Verbesserungen zu ermöglichen.
- Implementieren Sie Änderungen, die zu Verbesserungen führen sollen, und beurteilen Sie deren Ergebnisse.
- Versuchen Sie alternative Vorgehensweisen, wenn die Ergebnisse die Ziele nicht erfüllen und die Verbesserung immer noch Priorität hat.
- Simulieren Sie Produktionsworkloads durch GameDays, und nutzen Sie die Erkenntnisse aus diesen Simulationen, um sich zu verbessern.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS05-BP08 Verwenden Sie mehrere Umgebungen](#)

Zugehörige Videos:

- [AWS re:Invent 2023 — Verbessern Sie die Ausfallsicherheit von Anwendungen mit dem Fault Injection Service AWS](#)

Sicherheit

In der Säule der Sicherheit wird beschrieben, wie Sie Daten, Systeme und Komponenten so schützen, dass Sie Cloud-Technologien nutzen können, um Ihre Sicherheitslage zu verbessern. Verbindliche Anleitungen zur Implementierung finden Sie im [Whitepaper „Säule der Sicherheit“](#).

Bereiche für bewährte Methoden

- [Sicherheitsgrundlagen](#)
- [Identity and Access Management](#)
- [Erkennung](#)
- [Schutz der Infrastruktur](#)
- [Datenschutz](#)
- [Vorfallreaktion](#)
- [Anwendungssicherheit](#)

Sicherheitsgrundlagen

Frage

- [SEC 1. Wie wird Ihre Workload sicher verwaltet?](#)

SEC 1. Wie wird Ihre Workload sicher verwaltet?

Um Ihre Workload sicher zu betreiben, müssen Sie in allen Sicherheitsbereichen übergreifende bewährte Methoden anwenden. Wenden Sie die Anforderungen und Prozesse, die Sie im Bereich Operational Excellence auf Organisations- und Workload-Ebene definiert haben, auf alle Bereiche an. Wenn Sie über AWS- und Branchenempfehlungen sowie Bedrohungsinformationen auf dem Laufenden bleiben, können Sie Ihr Bedrohungsmodell und Ihre Kontrollziele weiterentwickeln. Die Automatisierung von Sicherheitsprozessen, Tests und Validierung ermöglicht es Ihnen, Ihre operativen Abläufe zu skalieren.

Best Practices

- [SEC01-BP01 Trennen von Workloads mithilfe von Konten](#)
- [SEC01-BP02 Schutz des Konto-Root-Benutzers und seiner Eigenschaften](#)
- [SEC01-BP03 Identifizieren und Validieren von Kontrollzielen](#)

- [SEC01-BP04 Bleiben Sie über Sicherheitsbedrohungen und Empfehlungen auf dem Laufenden](#)
- [SEC01-BP05 Reduzieren Sie den Umfang des Sicherheitsmanagements](#)
- [SEC01-BP06 Automatisieren Sie die Implementierung von Standardsicherheitskontrollen](#)
- [SEC01-BP07 Identifizieren von Bedrohungen und Priorisieren von Abhilfemaßnahmen unter Verwendung eines Bedrohungsmodells](#)
- [SEC01-BP08 Regelmäßiges Bewerten und Implementieren neuer Sicherheitservices und -features](#)

SEC01-BP01 Trennen von Workloads mithilfe von Konten

Sorgen Sie mit einer Mehrkonten-Strategie für wirksamen Integritätsschutz und Isolierungen zwischen Umgebungen (etwa Produktion, Entwicklung und Test) sowie Workloads. Die Trennung auf Kontoebene wird nachdrücklich angeraten, da diese für die wirksame Isolierung für Sicherheits-, Fakturierungs- und Zugriffszwecke sorgt.

Gewünschtes Ergebnis: eine Kontostruktur, die Cloud-Vorgänge, nicht zusammengehörige Workloads und Umgebungen in separaten Konten voneinander isoliert, sodass die Sicherheit in der gesamten Cloud-Infrastruktur verbessert wird.

Typische Anti-Muster:

- Platzierung mehrerer nicht zusammengehöriger Workloads mit unterschiedlicher Datensensitivität in einem einzigen Konto
- Schlecht definierte Organizational Unit (OU, Organisationseinheit)-Struktur

Vorteile der Nutzung dieser bewährten Methode:

- Geringere Auswirkungen bei versehentlichen Zugriffen auf eine Workload
- Zentrale Verwaltung des Zugriffs auf AWS-Services, Ressourcen und Regionen
- Wahrung der Sicherheit der Cloud-Infrastruktur durch Richtlinien und die zentralisierte Verwaltung von Sicherheitservices
- Automatisierte Kontoerstellung und Wartungsprozesse
- Zentralisierte Prüfung Ihrer Infrastruktur auf Compliance- und regulatorische Anforderungen

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

AWS-Konten bieten eine Sicherheitsisolierungsgrenze zwischen Workloads oder Ressourcen, die auf unterschiedlichen Sensitivitätsstufen operieren. AWS bietet Tools, mit denen Sie Ihre umfangreichen Cloud-Workloads über eine Mehrkonten-Strategie verwalten und so die Isolierungsgrenze nutzen können. Erläuterungen der Konzepte, Muster und der Implementierung einer Mehrkonten-Strategie in AWS finden Sie unter [Organizing Your AWS Environment Using Multiple Accounts](#).

Wenn Sie mehrere AWS-Konten zentral verwalten, sollten Ihre Konten in einer gemäß den Ebenen der Organisationseinheiten (OEs) definierten Hierarchie organisiert sein. Dadurch können Sicherheitskontrollen anhand der OEs und der Mitgliedskonten organisiert und auf diese angewendet werden, was eine konsistente präventive Kontrolle der Mitgliedskonten in der Organisation ermöglicht. Die Sicherheitskontrollen werden weitergegeben, sodass Sie nach verfügbaren Berechtigungen für Mitgliedskonten auf unteren Ebenen der OE-Hierarchie filtern können. Ein gutes Design macht sich diese Weitergabe zunutze, um die Anzahl und die Komplexität der Sicherheitsrichtlinien, die für die erwünschten Sicherheitskontrollen für jedes Mitgliedskonto erforderlich sind, zu reduzieren.

[AWS Organizations](#) und [AWS Control Tower](#) sind zwei Services, mit denen Sie diese Mehrkontenstruktur in Ihrer AWS-Umgebung implementieren und verwalten können. AWS Organizations ermöglicht die Organisation von Konten in einer von einer oder mehreren OE-Ebenen definierten Hierarchie, wobei jede OE eine Reihe von Mitgliedskonten enthält. [Service-Kontrollrichtlinien \(SCPs\)](#) ermöglichen einem Organisationsadministrator die Einrichtung detaillierter präventiver Kontrollen für Mitgliedskonten und [AWS Config](#) kann verwendet werden, um proaktive und erkennende Kontrollen für Mitgliedskonten zu aktivieren. Viele AWS-Services lassen sich [in AWS Organizations integrieren](#) und bieten so delegierte administrative Kontrollen und führen servicespezifische Aufgaben für alle Mitgliedskonten in der Organisation durch.

Auf der Ebene über AWS Organizations ermöglicht [AWS Control Tower](#) die Einrichtung bewährter Methoden mit einem Klick für eine AWS-Mehrkontenumgebung mit einer [Landing Zone](#). Die Landing Zone ist der Einstiegspunkt für die Mehrkonten-Umgebung, eingerichtet von Control Tower. Control Tower bietet mehrere [Vorteile](#) gegenüber AWS Organizations. Hier sind drei Vorteile, die die Kontoverwaltung verbessern:

- Integrierter verpflichtender Integritätsschutz, der automatisch auf für die Organisation zugelassene Konten angewendet wird
- Optionaler Integritätsschutz, der für einen bestimmten Satz von OEs aktiviert und deaktiviert werden kann

- [AWS Control Tower Account Factory](#) bietet eine automatisierte Bereitstellung von Konten mit vorab genehmigten Baselines und Konfigurationsoptionen innerhalb Ihrer Organisation.

Implementierungsschritte

1. Entwurf einer Struktur für Organisationseinheiten: Eine ordnungsgemäß gestaltete Struktur für Organisationseinheiten reduziert den Verwaltungsaufwand für die Erstellung und Wahrung von Service-Kontrollrichtlinien und anderen Sicherheitskontrollen. Ihre Struktur für Organisationseinheiten sollte [an Ihre geschäftlichen Anforderungen, die Sensitivität der Daten und die Workload-Struktur angepasst sein](#).
2. Erstellen einer Landing Zone für Ihre Mehrkontenumgebung: Eine Landing Zone bietet eine konsistente Sicherheits- und Infrastrukturbasis, über die Ihre Organisation Workloads schnell entwickeln, starten und bereitstellen kann. Sie können eine [individuell erstellte Landing Zone AWS Control Tower oder](#) für die Orchestrierung Ihrer Umgebung verwenden.
3. Einrichtung von Integritätsschutz: Implementieren Sie konsistenten Integritätsschutz für Ihre Umgebung über Ihre Landing Zone. AWS Control Tower bietet eine Liste [verpflichtender](#) und [optionaler](#) Kontrollen, die bereitgestellt werden können. Verpflichtende Kontrollen werden automatisch bereitgestellt, wenn Control Tower implementiert wird. Überprüfen Sie die Liste nachdrücklich empfohlener sowie optionaler Kontrollen und implementieren Sie diejenigen, die Ihren Anforderungen entsprechen.
4. Einschränken des Zugriffs auf neu hinzugefügte Regionen: Für neue AWS-Regionen werden IAM-Ressourcen, z. B. Benutzer und Rollen, nur an die von Ihnen angegebenen Regionen weitergegeben. Dieser Vorgang kann über die [Konsole durchgeführt werden, wenn Sie Control Tower verwenden](#), oder durch die Anpassung von [IAM-Berechtigungsrichtlinien in AWS Organizations](#).
5. Erwägen der Verwendung von AWS [CloudFormation StackSets](#): StackSets helfen dabei, Ressourcen wie IAM-Richtlinien, -Rollen und -Gruppen aus einer genehmigten Vorlage in verschiedenen AWS-Konten-Konten und Regionen bereitzustellen.

Ressourcen

Zugehörige bewährte Methoden:

- [SEC02-BP04 Verlassen auf einen zentralen Identitätsanbieter](#)

Zugehörige Dokumente:

- [AWS Control Tower](#)
- [Richtlinien zur AWS-Sicherheitsprüfung](#)
- [IAM Best Practices](#)
- [Use CloudFormation StackSets to provision resources across multiple AWS-Konten and regions](#)
- [Organizations – Häufig gestellte Fragen](#)
- [Terminologie und Konzepte von AWS Organizations](#)
- [Best Practices for Service Control Policies in an AWS Organizations Multi-Account Environment](#)
- [AWS-Referenzhandbuch zur Kontoverwaltung](#)
- [Organisieren Sie Ihre AWS-Umgebung mit mehreren Konten](#)

Zugehörige Videos:

- [Enable AWS adoption at scale with automation and governance](#)
- [Security Best Practices the Well-Architected Way](#)
- [Building and Governing Multiple Accounts using AWS Control Tower](#)
- [Enable Control Tower for Existing Organizations](#)

SEC01-BP02 Schutz des Konto-Root-Benutzers und seiner Eigenschaften

Der Root-Benutzer ist in einem AWS-Konto der Benutzer mit den meisten Berechtigungen und vollständigem administrativem Zugriff auf alle Ressourcen in dem Konto und kann in manchen Fällen nicht von Sicherheitsrichtlinien eingeschränkt werden. Die Deaktivierung des programmatischen Zugriffs auf den Root-Benutzer, die Einrichtung geeigneter Kontrollen für den Root-Benutzer und das Vermeiden der routinemäßigen Verwendung des Root-Benutzers senken die Risiken einer unbeabsichtigten Offenlegung der Anmeldeinformationen des Root-Benutzers und daraus resultierender ernsthafter Probleme für die Cloud-Umgebung.

Gewünschtes Ergebnis: Das Sichern des Root-Benutzers hilft dabei, die Gefahr zu verringern, dass versehentliche oder beabsichtigte Schäden durch den Missbrauch der Anmeldeinformationen des Root-Benutzers entstehen. Die Einrichtung erkennender Kontrollen kann auch für die Benachrichtigung der richtigen Personen sorgen, wenn Aktionen unter Verwendung des Root-Benutzers durchgeführt werden.

Typische Anti-Muster:

- Verwendung des Root-Benutzers für andere Aufgaben als die wenigen, für die Root-Benutzer-Anmeldeinformationen erforderlich sind
- Versäumnis, Notfallpläne regelmäßig zu testen, um das Funktionieren kritischer Infrastrukturen, Prozesse und des Personals während eines Notfalls zu überprüfen.
- ausschließliche Berücksichtigung des typischen Kontoanmeldungsprozesses und keine Berücksichtigung alternativer Kontowiederherstellungsverfahren
- keine Behandlung von DNS, E-Mail-Servern und Telefonanbietern als Teil des kritischen Sicherheitsperimeters, da diese in den Kontowiederherstellungsabläufen verwendet werden

Vorteile der Nutzung dieser bewährten Methode: Der Schutz des Zugriffs auf den Root-Benutzer stärkt das Vertrauen dazu, dass Aktionen in Ihrem Konto kontrolliert und überwacht werden.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

AWS bietet zahlreiche Tools für den Schutz Ihres Kontos. Da einige dieser Maßnahmen aber nicht standardmäßig aktiviert sind, müssen Sie sie selbst implementieren. Betrachten Sie diese Empfehlungen als grundlegende Schritte für den Schutz Ihres AWS-Konto. Bei der Implementierung dieser Schritte ist es wichtig, dass Sie einen Prozess für die kontinuierliche Bewertung und Überwachung der Sicherheitskontrollen einrichten.

Wenn Sie ein AWS-Konto anlegen, beginnen Sie mit einer Identität, mit der Sie auf alle mit dem Konto verbundenen AWS-Services und -Ressourcen zugreifen können. Diese Identität wird als AWS-Konto-Root-Benutzer bezeichnet. Sie können sich als Stammbenutzer mit der E-Mail-Adresse und dem Passwort anmelden, die Sie bei der Erstellung des Kontos verwendet haben. Da der AWS-Root-Benutzer erweiterte Zugriffsrechte hat, müssen Sie die Verwendung des AWS-Root-Benutzers auf die Aufgaben beschränken, für die er [ausdrücklich erforderlich](#) ist. Die Anmeldeinformationen des Root-Benutzers müssen sehr gut geschützt werden, und für den Root-Benutzer des AWS-Konto sollte immer die Multi-Faktor-Authentifizierung (MFA) genutzt werden.

Zusätzlich zum normalen Authentifizierungsablauf bei der Anmeldung als Root-Benutzer mit einem Benutzernamen, Passwort und einem Gerät zur Multi-Faktor-Authentifizierung (MFA) gibt es Kontowiederherstellungsabläufe für die Anmeldung Ihres AWS-Konto als Root-Benutzer mit Zugriff auf die mit Ihrem Konto verbundene E-Mail-Adresse und die Telefonnummer. Daher ist es ebenso wichtig, das E-Mail-Konto des Root-Benutzers, an das die Wiederherstellungs-E-Mail gesendet wird, und die mit dem Konto verknüpfte Telefonnummer zu sichern. Denken Sie auch an mögliche zirkuläre

Abhängigkeiten, bei denen die zum Root-Benutzer gehörende E-Mail-Adresse auf E-Mail-Servern oder Domain Name Service (DNS)-Ressourcen von demselben AWS-Konto gehostet wird.

Bei Verwendung von AWS Organizations gibt es mehrere AWS-Konten, die jeweils einen Root-Benutzer haben. Ein Konto fungiert als Verwaltungskonto und mehrere Ebenen von Mitgliedskonten können dann darunter hinzugefügt werden. Priorisieren Sie den Schutz des Root-Benutzers Ihres Verwaltungskontos und kümmern Sie sich dann um diejenigen der Mitgliedskonten. Die Strategie zum Schutz des Root-Benutzers Ihres Verwaltungskontos kann sich von der für die Root-Benutzer der Mitgliedskonten unterscheiden und Sie können präventive Sicherheitskontrollen für die Root-Benutzer Ihrer Mitgliedskonten einrichten.

Implementierungsschritte

Die folgenden Implementierungsschritte werden für die Einrichtung der Kontrollen für den Root-Benutzer empfohlen. Gegebenenfalls verweisen die Empfehlungen auf [CIS AWS Foundations Benchmark, Version 1.4.0](#). Konsultieren Sie zusätzlich zu diesen Schritten die [Richtlinien zu bewährten Methoden für AWS](#) für den Schutz Ihres AWS-Konto und Ihrer Ressourcen.

Präventive Kontrollen

1. Richten Sie genaue [Kontaktinformationen](#) für das Konto ein.
 - a. Diese Informationen werden für die Abläufe zur Wiederherstellung verlorener Passwörter, verlorener MFA-Gerätekonten und für die kritische sicherheitsrelevante Kommunikation mit Ihrem Team verwendet.
 - b. Verwenden Sie eine von ihrer Unternehmensdomain gehostete E-Mail-Adresse, vorzugsweise eine Verteilerliste, als E-Mail-Adresse des Root-Benutzers. Die Verwendung einer Verteilerliste anstelle einer einzelnen E-Mail-Adresse sorgt für zusätzliche Redundanz und Kontinuität beim Zugriff auf das Root-Konto über längere Zeiträume hinweg.
 - c. Die in den Kontaktinformationen angegebene Telefonnummer sollte eine für diesen Zweck speziell eingerichtete und sichere Telefonnummer sein. Diese Telefonnummer sollte nicht eingetragen sein oder an andere weitergegeben werden.
2. Erstellen Sie keine Zugriffsschlüssel für den Root-Benutzer. Wenn Zugriffsschlüssel vorhanden sind, entfernen Sie diese (CIS 1.4).
 - a. Entfernen Sie alle langfristigen programmatischen Anmeldeinformationen (Zugriffs- und geheime Schlüssel) für den Root-Benutzer.
 - b. Wenn bereits Zugriffsschlüssel für den Root-Benutzer vorhanden sind, sollten Prozesse, die diese Schlüssel verwenden, so umgestaltet werden, dass sie temporäre Zugriffsschlüssel von

einer AWS Identity and Access Management (IAM)-Rolle verwenden; löschen Sie dann die [Zugriffsschlüssel des Root-Benutzers](#).

3. Ermitteln Sie, ob Sie Anmeldeinformationen für den Root-Benutzer speichern müssen.
 - a. Wenn Sie AWS Organizations zum Erstellen neuer Mitgliedskonten verwenden, wird das ursprüngliche Passwort für den Root-Benutzer in neuen Mitgliedskonten auf einen zufälligen Wert festgelegt, der Ihnen nicht angezeigt wird. Erwägen Sie die Nutzung der Passwortrücksetzung von Ihrem AWS-Organization-Verwaltungskonto, um bei Bedarf [Zugriff auf das Mitgliedskonto zu erhalten](#).
 - b. Für Standalone-AWS-Konten oder das AWS-Organization-Verwaltungskonto sollten Sie Anmeldeinformationen für den Root-Benutzer erstellen und sicher speichern. Verwenden Sie MFA für den Root-Benutzer.
4. Aktivieren Sie präventive Kontrollen für Root-Benutzer von Mitgliedskonten in AWS-Mehrkonten-Umgebungen.
 - a. Erwägen Sie die präventive Sicherheitsvorkehrung [Erstellung von Zugriffsschlüsseln für den Root-Benutzer nicht zulassen](#) für Mitgliedskonten.
 - b. Erwägen Sie die Aktivierung der präventiven Sicherheitsmaßnahme [Aktionen als Root-Benutzer nicht zulassen](#) für Mitgliedskonten.
5. Wenn Sie Anmeldeinformationen für den Root-Benutzer benötigen:
 - a. Verwenden Sie ein komplexes Passwort.
 - b. Aktivieren Sie Multi-Faktor-Authentifizierung (MFA) für den Root-Benutzer, besonders für AWS Organizations-Verwaltungskonten (Bezahlerkonten) (CIS 1.5).
 - c. Erwägen Sie die Nutzung von Hardware-MFA-Geräten für Resilienz und Sicherheit, da Einweggeräte auf MFA-Funktionen begrenzt sind und so die Wahrscheinlichkeit verringern, dass die Geräte mit Ihren MFA-Codes für andere Zwecke verwendet werden. Stellen Sie sicher, dass batteriebetriebene MFA-Geräte regelmäßig ausgetauscht werden. (CIS 1.6)
 - Befolgen Sie zur Konfiguration der MFA für den Root-Benutzer die Anleitungen für die Aktivierung einer [virtuellen MFA](#) oder eines [Hardware-MFA-Geräts](#).
 - d. Erwägen Sie die Nutzung mehrerer MFA-Geräte als Sicherung. [Pro Konto sind bis zu 8 MFA-Geräte zulässig](#).
 - Beachten Sie, dass die Verwendung von mehr als einem MFA-Gerät für den Root-Benutzer automatisch den [Ablauf für die Wiederherstellung Ihres Kontos bei Verlust des MFA-Geräts deaktiviert](#).

- e. Speichern Sie das Passwort in sicherer Weise, und beachten Sie zirkuläre Abhängigkeiten bei der elektronischen Speicherung des Passworts. Speichern Sie das Passwort nicht so, dass Zugriff auf dasselbe AWS-Konto erforderlich wäre, um es abzurufen.
6. Optional: Erwägen Sie die Einrichtung einer periodischen Passwortrotation für den Root-Benutzer.
- Bewährte Methoden für die Verwaltung von Anmeldeinformationen hängen von Ihren jeweiligen regulatorischen und Richtlinienanforderungen ab. Durch MFA geschützte Root-Benutzer sind nicht auf das Passwort als einzigen Authentifizierungsfaktor angewiesen.
 - [Die regelmäßige Änderung des Root-Benutzer-Passworts](#) senkt das Risiko, dass ein unbeabsichtigt offengelegtes Passwort missbraucht werden kann.

Detektivische Kontrollen

- Erstellen Sie Alarme, um die Verwendung der Root-Anmeldeinformationen zu erkennen (CIS 1.7). [Amazon GuardDuty](#) kann die Nutzung der API-Anmeldeinformationen des Root-Benutzers überwachen und Sie über das Ergebnis von [RootCredentialUsage](#) benachrichtigen.
- Evaluieren und implementieren Sie die im [Konformitätspaket für AWS Config der AWS-Well-Architected-Säule „Sicherheit“](#) enthaltenen aufdeckenden Kontrollen oder, falls Sie AWS Control Tower verwenden, die [nachdrücklich empfohlenen Kontrollen](#), die in Control Tower verfügbar sind.

Operative Anleitung

- Legen Sie fest, wer in der Organisation Zugriff auf die Root-Benutzer-Anmeldeinformationen haben sollte.
- Verwenden Sie eine Zwei-Personen-Regel, damit keine einzelne Person Zugang zu allen erforderlichen Anmeldeinformationen und zur MFA hat, um sich Root-Benutzer-Zugriff zu verschaffen.
- Stellen Sie sicher, dass die Organisation – und nicht nur eine einzelne Person – die Kontrolle über die mit dem Konto verbundene Telefonnummer und das entsprechende E-Mail-Alias hat (diese werden für die Passwort- und die MFA-Rücksetzung verwendet).
- Verwenden Sie nur im Ausnahmefall den Root-Benutzer (CIS 1.7).
 - Der AWS-Root-Benutzer darf nicht für alltägliche Aktivitäten verwendet werden, auch nicht für administrative. Melden Sie sich nur dann als Root-Benutzer an, wenn Sie [AWS-Aufgaben durchführen müssen, für die der Root-Benutzer erforderlich ist](#). Alle anderen Aktionen sollten von anderen Benutzern mit den entsprechenden Rollen durchgeführt werden.

- Prüfen Sie regelmäßig, ob der Zugriff auf den Root-Benutzer funktioniert, um Prozeduren vor dem Eintreten von Notsituationen zu testen, die die Verwendung der Root-Benutzer-Anmeldeinformationen erfordern.
- Prüfen Sie regelmäßig, ob die mit dem Konto verbundene E-Mail-Adresse und die unter [Alternative Kontakte](#) aufgeführten E-Mail-Adressen funktionieren. Überwachen Sie diese E-Mail-Posteingänge auf etwaige Sicherheitsmitteilungen von <abuse@amazon.com>. Stellen Sie auch sicher, dass alle mit dem Konto verbundenen Telefonnummern funktionieren.
- Bereiten Sie Notfallreaktionsprozeduren vor, um auf den Missbrauch des Root-Kontos reagieren zu können. Weitere Informationen zum Aufbau einer Sicherheitsstrategie für Ihr AWS-Konto finden Sie im [AWS-Reaktionsleitfaden für Sicherheitsvorfälle](#) und in den bewährten Methoden im [Abschnitt zu Vorfälleaktionen im Whitepaper zur Säule „Sicherheit“](#).

Ressourcen

Zugehörige bewährte Methoden:

- [SEC01-BP01 Trennen von Workloads mithilfe von Konten](#)
- [SEC02-BP01 Verwenden von starken Anmeldemechanismen](#)
- [SEC03-BP02 Gewähren des Zugriffs mit den geringsten Berechtigungen](#)
- [SEC03-BP03 Einrichtung eines Notfallzugriffprozesses](#)
- [SEC10-BP05 Vorab bereitgestellter Zugriff](#)

Zugehörige Dokumente:

- [AWS Control Tower](#)
- [Richtlinien zur AWS-Sicherheitsprüfung](#)
- [IAM Best Practices](#)
- [Amazon GuardDuty – Warnung bei Verwendung der Root-Anmeldeinformationen](#)
- [Schritt-für-Schritt-Anleitung zur Überwachung der Verwendung von Root-Anmeldeinformationen mit CloudTrail](#)
- [Zur Verwendung mit AWS genehmigte MFA-Token](#)
- [Implementieren von „Break Glass“-Zugriff in AWS](#)
- [Top 10 security items to improve in your AWS-Konto](#)

- [What do I do if I notice unauthorized activity in my AWS-Konto?](#)

Zugehörige Videos:

- [Enable AWS adoption at scale with automation and governance](#)
- [Security Best Practices the Well-Architected Way](#)
- [Limiting use of AWS root credentials](#) from AWS re:inforce 2.022 – Security best practices with AWS IAM

SEC01-BP03 Identifizieren und Validieren von Kontrollzielen

Entsprechend Ihren Compliance-Anforderungen und Risiken, die aus Ihrem Bedrohungsmodell identifiziert werden, können Sie die Kontrollziele und Kontrollen ableiten und validieren, die Sie für Ihre Workload benötigen. Die laufende Validierung von Kontrollzielen und Kontrollen hilft Ihnen, die Effektivität der Risikominderung zu messen.

Gewünschtes Ergebnis: Die Kontrollziele Ihres Unternehmens sind klar definiert und auf Ihre Compliance-Anforderungen abgestimmt. Kontrollen werden durch Automatisierung und Richtlinien implementiert und durchgesetzt und kontinuierlich auf ihre Wirksamkeit bei der Erreichung Ihrer Ziele überprüft. Die Belege für die Wirksamkeit sowohl zu einem bestimmten Zeitpunkt als auch über einen bestimmten Zeitraum hinweg sind jederzeit für Prüfer abrufbar.

Typische Anti-Muster:

- Regulatorische Anforderungen, Markterwartungen und Branchenstandards für verlässliche Sicherheit sind in Ihrem Unternehmen nicht hinreichend vertraut.
- Ihr Framework für die Cybersicherheit und Ihre Kontrollziele sind nicht an den Anforderungen Ihres Unternehmens ausgerichtet.
- Die Implementierung der Kontrollen ist nicht messbar auf Ihre Kontrollziele ausgerichtet.
- Sie verwenden keine Automatisierung zur Berichterstattung über die Wirksamkeit Ihrer Kontrollen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Es gibt zahlreiche gängige Frameworks für die Cybersicherheit, die die Grundlage für Ihre Sicherheitskontrollziele bilden können. Berücksichtigen Sie die regulatorischen Anforderungen,

die Markterwartungen und die Branchenstandards für Ihr Unternehmen, um festzustellen, welches Framework Ihre Anforderungen am besten erfüllt. Beispiele hierfür sind u. a. [AICPA SOC 2](#), [HITRUST](#), [PCI-DSS](#), [ISO 27001](#) und [NIST SP 800-53](#).

Für die von Ihnen festgelegten Kontrollziele sollten Sie verstehen, wie die von Ihnen in Anspruch genommenen AWS-Services Ihnen helfen, diese Ziele zu erreichen. Unter [AWS Artifact](#) finden Sie Dokumentationen und Berichte, die auf Ihre Zielframeworks abgestimmt sind. Darin wird der Verantwortungsbereich von AWS beschrieben. Ferner können Sie dort Anleitungen erhalten, in denen der verbleibende Umfang, für den Sie verantwortlich sind, beschrieben wird. Weitere servicespezifische Anleitungen, die sich an verschiedenen Regelwerken orientieren, finden Sie unter [AWS Customer Compliance Guides](#).

Während Sie die Kontrollen zur Erreichung Ihrer Ziele definieren, kodifizieren Sie die Durchsetzung mithilfe von präventiven Kontrollen und automatisieren die Abschwächung mithilfe von detektivischen Kontrollen. Verhindern Sie nicht konforme Ressourcenkonfigurationen und Aktionen in AWS Organizations mithilfe von [Service-Kontrollrichtlinien \(SCPs\)](#). Implementieren Sie Regeln in [AWS Config](#) zur Überwachung und Berichterstattung über nicht konforme Ressourcen und wechseln Sie dann zu einem Durchsetzungsmodell, sobald Sie von deren Verhalten überzeugt sind. Wenn Sie vordefinierte und verwaltete Regeln bereitstellen möchten, die sich an Ihren Cybersicherheits-Rahmenbedingungen orientieren, sollten Sie die Verwendung von [AWS Security Hub CSPM-Standards](#) als erste Wahl in Betracht ziehen. Der Standard „Foundational Service Best Practices (FSBP)“ von AWS und der CIS-AWS-Foundations-Benchmark sind gute Ausgangspunkte mit Kontrollen, die auf zahlreiche Ziele ausgerichtet sind, die in mehreren Standardframeworks gemeinsam genutzt werden. In Fällen, in denen Security Hub CSPM nicht intrinsisch die gewünschten Kontrollmeldungen verfügt, kann es durch [AWS Config-Konformitätspakete](#) ergänzt werden.

Verwenden Sie [APN-Partnerpakete](#), die vom Global Security and Compliance Acceleration (GSCA)-Team von AWS empfohlen werden, um bei Bedarf Unterstützung von Sicherheitsberatern, Beratungsagenturen, Beweissammlungs- und Berichtssystemen, Prüfern und anderen ergänzenden Services zu erhalten.

Implementierungsschritte

1. Bewerten Sie gängige Frameworks für Cybersicherheit und richten Sie Ihre Kontrollziele an den ausgewählten Frameworks aus.
2. Beschaffen Sie sich mithilfe von AWS Artifact einschlägige Unterlagen über Leitlinien und Verantwortlichkeiten für Ihr Framework. Machen Sie sich klar, welche Teile der Compliance in

den AWS-Bereich des Modells der gemeinsamen Verantwortung fallen und für welche Teile Sie verantwortlich sind.

3. Verwenden Sie SCPs, Ressourcenrichtlinien, Rollenvertrauensrichtlinien und andere Maßnahmen für den Integritätsschutz, um nicht konforme Ressourcenkonfigurationen und Aktionen zu verhindern.
4. Evaluieren Sie die Implementierung von Security-Hub-CSPM-Standards und AWS Config-Konformitätspaketen, die mit Ihren Kontrollzielen übereinstimmen.

Ressourcen

Zugehörige bewährte Methoden:

- [SEC03-BP01 Definieren von Zugriffsanforderungen](#)
- [SEC04-BP01 Konfigurieren der Service- und Anwendungsprotokollierung](#)
- [SEC07-BP01 Verstehen Ihres Schemas zur Datenklassifizierung](#)
- [OPS01-BP03 Bewerten der Governance-Anforderungen](#)
- [OPS01-BP04 Bewerten der Compliance-Anforderungen](#)
- [PERF01-BP05 Verwenden von Richtlinien und Referenzarchitekturen](#)
- [COST02-BP01 Entwickeln von Richtlinien auf Basis Ihrer Organisationsanforderungen](#)

Zugehörige Dokumente:

- [AWS Customer Compliance Guides](#)

Zugehörige Tools:

- [AWS Artifact](#)

SEC01-BP04 Bleiben Sie über Sicherheitsbedrohungen und Empfehlungen auf dem Laufenden

Bleiben Sie auf dem Laufenden über die neuesten Bedrohungen und Abhilfemaßnahmen, indem Sie Veröffentlichungen zu Bedrohungsdaten und Datenfeeds der Branche auf Aktualisierungen verfolgen. Prüfen Sie Angebote für verwaltete Services, die automatisch auf der Grundlage der neuesten Bedrohungsdaten aktualisiert werden.

Gewünschtes Ergebnis: Sie bleiben auf dem Laufenden, da die Branchenpublikationen mit den neuesten Bedrohungen und Empfehlungen aktualisiert werden. Sie nutzen die Automatisierung, um potenzielle Schwachstellen und Gefährdungen zu erkennen, während Sie neue Bedrohungen identifizieren. Sie ergreifen Maßnahmen zur Eindämmung dieser Bedrohungen. Sie AWS nutzen Dienste, die automatisch mit den neuesten Bedrohungsinformationen aktualisiert werden.

Typische Anti-Muster:

- Kein zuverlässiger und wiederholbarer Mechanismus, um über die neuesten Bedrohungsdaten informiert zu sein
- Manuelle Bestandsführung Ihres Technologieportfolios, Ihrer Workloads und Abhängigkeiten, was menschliches Eingreifen im Hinblick auf potenzielle Schwachstellen und Gefährdungen erfordert
- Fehlende Mechanismen zur Aktualisierung Ihrer Workloads und Abhängigkeiten auf die neuesten verfügbaren Versionen, die bekannte Bedrohungsabwehrmaßnahmen bieten

Vorteile der Nutzung dieser bewährten Methode: Die Verwendung von Bedrohungsdatenquellen, um auf dem Laufenden zu bleiben, verringert das Risiko, wichtige Änderungen in der Bedrohungslandschaft zu verpassen, die sich auf Ihr Unternehmen auswirken können. Wenn Sie Ihre Workloads und deren Abhängigkeiten automatisiert auf potenzielle Schwachstellen oder Gefährdungen prüfen, diese erkennen und beheben, können Sie Risiken im Vergleich zu manuellen Alternativen schnell und vorhersehbar eindämmen. Dies trägt dazu bei, Zeit und Kosten im Zusammenhang mit der Behebung von Schwachstellen zu kontrollieren.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Verfolgen Sie vertrauenswürdige Veröffentlichungen zu Bedrohungsdaten, um über die Bedrohungslandschaft auf dem Laufenden zu bleiben. Dokumentation zu bekannten gegnerischen Taktiken, Techniken und Verfahren finden Sie in der [MITRE ATT&CK](#) Knowledge Base (). TTPs Lesen Sie MITRE die Liste der [häufigsten Sicherheitslücken und Risiken](#) (CVE), um über bekannte Sicherheitslücken in Produkten, auf die Sie sich verlassen, auf dem Laufenden zu bleiben. Machen Sie sich mit dem beliebten [OWASPTop-10-Projekt](#) des Open Worldwide Application Security Project (OWASP) ein Bild von kritischen Risiken für Webanwendungen.

Bleiben Sie mit AWS den [Sicherheitsbulletins AWS](#) für über Sicherheitsereignisse und empfohlene Maßnahmen zur Problembehebung auf dem Laufenden. CVEs

Um Ihren Gesamtaufwand und die Kosten für die Aktualisierung zu reduzieren, sollten Sie die Nutzung von AWS Diensten in Betracht ziehen, die im Laufe der Zeit automatisch neue Bedrohungsinformationen einbeziehen. [Amazon GuardDuty](#) hält sich beispielsweise über branchenspezifische Bedrohungsinformationen auf dem Laufenden, um ungewöhnliche Verhaltensweisen und Bedrohungssignaturen in Ihren Konten zu erkennen. [Amazon Inspector](#) hält automatisch eine Datenbank mit den CVEs für seine kontinuierlichen Scanfunktionen verwendeten Daten auf dem neuesten Stand. Sowohl [AWS WAF](#) als auch [AWS Shield Advanced](#) bieten verwaltete Regelgruppen, die automatisch aktualisiert werden, wenn neue Bedrohungen auftauchen.

Informationen zum automatisierten Flottenmanagement und Patching finden Sie unter [Säule „Operative Exzellenz“ – Well-Architected-Framework](#)

Implementierungsschritte

- Abonnieren Sie Updates für Bedrohungsinformationen, die für Ihr Unternehmen und Ihre Branche relevant sind. Abonnieren Sie die AWS -Sicherheitsberichte.
- Erwägen Sie die Einführung von Diensten wie Amazon GuardDuty und Amazon Inspector, die neue Bedrohungsinformationen automatisch integrieren.
- Erstellen Sie eine Flottenmanagement- und Patching-Strategie, die sich an den bewährten Methoden der der Säule „Operative Exzellenz“ des Well-Architected-Framework“ orientiert.

Ressourcen

Zugehörige bewährte Methoden:

- [SEC01-BP07 Identifizieren Sie Bedrohungen und priorisieren Sie Abhilfemaßnahmen mithilfe eines Bedrohungsmodells](#)
- [OPS01-BP05 Bewerten Sie die Bedrohungslandschaft](#)
- [OPS11-BP01 Haben Sie einen Prozess zur kontinuierlichen Verbesserung](#)

SEC01-BP05 Reduzieren Sie den Umfang des Sicherheitsmanagements

Stellen Sie fest, ob Sie Ihren Sicherheitsbereich reduzieren können, indem Sie AWS Dienste verwenden, bei denen die Verwaltung bestimmter Kontrollen auf AWS (verwaltete Dienste) verlagert wird. Mit diesen Services können Sie Ihre Wartungsaufgaben im Bereich Sicherheit reduzieren, z. B. die Bereitstellung der Infrastruktur, die Einrichtung von Software, Patches oder Sicherungen.

Gewünschtes Ergebnis: Sie berücksichtigen den Umfang Ihres Sicherheitsmanagements, wenn Sie AWS Dienste für Ihren Workload auswählen. Die Kosten für Verwaltungsaufwand und Wartungsaufgaben (die Gesamtbetriebskosten oder TCO) werden zusätzlich zu anderen Well-Architected-Überlegungen gegen die Kosten der von Ihnen ausgewählten Services abgewogen. Sie beziehen die AWS Kontroll- und Compliance-Dokumentation in Ihre Kontrollbewertungs- und Überprüfungsverfahren ein.

Typische Anti-Muster:

- Bereitstellung von Workloads ohne gründliches Verständnis des Modells der geteilten Verantwortung für die von Ihnen ausgewählten Services
- Hosten von Datenbanken und anderen Technologien auf virtuellen Maschinen, ohne einen entsprechenden verwalteten Service evaluiert zu haben
- Nichtberücksichtigung von Sicherheitsverwaltungsaufgaben bei den Gesamtbetriebskosten des Hostings von Technologien auf virtuellen Maschinen im Vergleich zu verwalteten Serviceoptionen

Vorteile der Nutzung dieser bewährten Methode: Der Einsatz von verwalteten Services kann Ihren Gesamtaufwand für die Verwaltung der betrieblichen Sicherheitskontrollen verringern, was Ihre Sicherheitsrisiken und Gesamtbetriebskosten reduzieren kann. Die Zeit, die Sie sonst für bestimmte Sicherheitsaufgaben aufwenden müssten, können Sie in Aufgaben investieren, die Ihrem Unternehmen einen größeren Nutzen bringen. Verwaltete Services können auch den Umfang Ihrer Compliance-Anforderungen reduzieren, indem sie einige Kontrollanforderungen in AWS verlagern.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Es gibt mehrere Möglichkeiten, wie Sie die Komponenten Ihrer Workload in AWS integrieren können. Die Installation und Ausführung von Technologien auf EC2 Amazon-Instances erfordert häufig, dass Sie den größten Teil der gesamten Sicherheitsverantwortung übernehmen. Um den Aufwand für die Durchführung bestimmter Kontrollen zu verringern, sollten Sie AWS Managed Services identifizieren, die Ihren Teil des Modells der gemeinsamen Verantwortung einschränken, und sich darüber informieren, wie Sie sie in Ihrer bestehenden Architektur einsetzen können.

[Beispiele hierfür sind die Verwendung des Amazon Relational Database Service \(AmazonRDS\) für die Bereitstellung von Datenbanken, Amazon Elastic Kubernetes Service \(AmazonEKS\) oder AmazonElastic Container Service \(AmazonECS\) für die Orchestrierung von Containern oder die Verwendung serverloser Optionen.](#) Überlegen Sie bei der Entwicklung neuer Anwendungen,

welche Services dazu beitragen können, den Zeit- und Kostenaufwand für die Implementierung und Verwaltung von Sicherheitskontrollen zu reduzieren.

Auch Compliance-Anforderungen können bei der Auswahl von Services eine Rolle spielen. Managed Services können die Einhaltung einiger Anforderungen auf Folgendes verlagern. AWS Erkundigen Sie sich mit Ihrem Compliance-Team darüber, inwieweit es sich mit der Prüfung der Aspekte der von Ihnen betriebenen und verwalteten Dienste sowie mit der Annahme von Kontrollerklärungen in den entsprechenden AWS Prüfberichten auskennt. Sie können die gefundenen Prüfarbefakte Ihren Prüfern oder Aufsichtsbehörden als Nachweis für AWS Sicherheitskontrollen zur Verfügung stellen. [AWS Artifact](#) Sie können bei der Gestaltung Ihrer Architektur auch die in einigen AWS Prüfarbefakten enthaltenen Hinweise zur Verantwortung sowie die Richtlinien zur [Einhaltung der AWS Kundenvorschriften verwenden](#). Dieser Leitfaden hilft Ihnen, die zusätzlichen Sicherheitskontrollen zu bestimmen, die Sie einrichten sollten, um die spezifischen Anwendungsfälle Ihres Systems zu unterstützen.

Wenn Sie Managed Services verwenden, sollten Sie mit dem Prozess der Aktualisierung ihrer Ressourcen auf neuere Versionen vertraut sein (z. B. mit der Aktualisierung der Version einer von Amazon RDS verwalteten Datenbank oder einer Programmiersprachen-Runtime für eine AWS Lambda Funktion). Auch wenn der verwaltete Service diesen Vorgang für Sie durchführt, sind Sie für die Konfiguration des Zeitpunkts der Aktualisierung und die Auswirkungen auf Ihren Betrieb selbst verantwortlich. Tools wie [AWS Health](#) können Ihnen helfen, diese Updates in Ihren Umgebungen zu verfolgen und zu verwalten.

Implementierungsschritte

1. Bewerten Sie die Komponenten Ihrer Workload, die durch einen verwalteten Service ersetzt werden können.
 - a. Wenn Sie einen Workload auf migrieren AWS, sollten Sie bei der Entscheidung, ob Sie Ihren Workload rehosten, refaktorisieren, plattformübergreifend, neu aufbauen oder ersetzen sollten, den geringeren Verwaltungsaufwand (Zeit und Kosten) und die Verringerung des Risikos berücksichtigen. Manchmal können zusätzliche Investitionen zu Beginn einer Migration auf lange Sicht erhebliche Einsparungen bringen.
2. Erwägen Sie die Implementierung von Managed Services wie AmazonRDS, anstatt Ihre eigenen Technologiebereitstellungen zu installieren und zu verwalten.
3. Finden Sie anhand der Leitlinien AWS Artifact zur Verantwortung heraus, welche Sicherheitskontrollen Sie für Ihren Workload einrichten sollten.

4. Führen Sie eine Bestandsaufnahme der verwendeten Ressourcen und bleiben Sie up-to-date bei neuen Services und Ansätzen, um neue Möglichkeiten zur Reduzierung des Umfangs zu identifizieren.

Ressourcen

Zugehörige bewährte Methoden:

- [PERF02-BP01 Wählen Sie die besten Rechenoptionen für Ihren Workload](#)
- [PERF03-BP01 Verwenden Sie einen speziell entwickelten Datenspeicher, der Ihre Datenzugriffs- und Speicheranforderungen am besten unterstützt](#)
- [SUS05-BP03 Verwenden Sie verwaltete Dienste](#)

Zugehörige Dokumente:

- [Geplante Lebenszyklusereignisse für AWS Health](#)

Zugehörige Tools:

- [AWS Health](#)
- [AWS Artifact](#)
- [AWS Customer Compliance Guides](#)

Zugehörige Videos:

- [Wie migriere ich zu einer Amazon RDS - oder Aurora My SQL DB-Instance mit AWS DMS?](#)
- [AWS re:Invent 2023 — Verwalten Sie Ereignisse im Ressourcenlebenszyklus in großem Umfang mit AWS Health](#)

SEC01-BP06 Automatisieren Sie die Implementierung von Standardsicherheitskontrollen

Wenden Sie bei der Entwicklung und Implementierung von Sicherheitskontrollen, die in Ihren Umgebungen zum Standard gehören, moderne DevOps Verfahren an. AWS Definieren Sie standardmäßige Sicherheitskontrollen und -konfigurationen mithilfe von Infrastructure as Code (IaC) -Vorlagen, erfassen Sie Änderungen in einem Versionskontrollsystem, testen Sie Änderungen als

Teil einer CI/CD-Pipeline und automatisieren Sie die Implementierung von Änderungen in Ihren Umgebungen. AWS

Gewünschtes Ergebnis: IaC-Vorlagen erfassen standardisierte Sicherheitskontrollen und übergeben sie an ein Versionskontrollsystem. CI/CD-Pipelines befinden sich an Stellen, an denen Änderungen erkannt und das Testen und Bereitstellen Ihrer Umgebungen automatisiert werden. AWS Mechanismen zum Integritätsschutz erkennen und warnen vor Fehlkonfigurationen in Vorlagen, bevor die Bereitstellung erfolgt. Workloads werden in Umgebungen bereitgestellt, in denen Standardkontrollen vorhanden sind. Die Teams können genehmigte Servicekonfigurationen über einen Selfservice-Mechanismus bereitstellen. Die Strategien zur Gewährleistung der Sicherheit bei der Sicherung und Wiederherstellung von Kontrollkonfigurationen, Skripten und zugehörigen Daten sind etabliert.

Typische Anti-Muster:

- Manuelle Änderungen an Ihren Standard-Sicherheitskontrollen über eine Webkonsole oder eine Befehlszeilenschnittstelle.
- Sich darauf verlassen, dass die einzelnen Workload-Teams die von einem zentralen Team festgelegten Kontrollen manuell umsetzen.
- Sich auf ein zentrales Sicherheitsteam verlassen, das auf Anfrage eines Workload-Teams Kontrollen auf Workload-Ebene bereitstellt.
- Erlauben, dass dieselben Personen oder Teams Automatisierungsskripte für die Sicherheitskontrolle entwickeln, testen und bereitstellen, ohne dass eine angemessene Aufgabentrennung oder gegenseitige Kontrolle stattfindet.

Vorteile der Nutzung dieser bewährten Methode: Die Verwendung von Vorlagen zur Definition Ihrer Standard-Sicherheitskontrollen ermöglicht es Ihnen, Änderungen im Laufe der Zeit mithilfe eines Versionskontrollsystems zu verfolgen und zu vergleichen. Der Einsatz von Automatisierung zum Testen und Bereitstellen von Änderungen schafft Standardisierung und Vorhersehbarkeit, erhöht die Chancen auf eine erfolgreiche Bereitstellung und reduziert manuelle, sich wiederholende Aufgaben. Durch die Bereitstellung eines Selfservice-Mechanismus für Workload-Teams zur Bereitstellung genehmigter Services und Konfigurationen wird das Risiko von Fehlkonfigurationen und Missbrauch verringert. Das hilft ihnen auch dabei, Kontrollen früher in den Entwicklungsprozess einzubauen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Wenn Sie die unter [SEC01-BP01 Separate Workloads mithilfe von Konten](#) beschriebenen Verfahren befolgen, werden Sie am Ende mehrere AWS-Konten für verschiedene Umgebungen verwenden, die Sie verwalten. AWS Organizations Auch wenn jede dieser Umgebungen und Workloads unterschiedliche Sicherheitskontrollen erfordert, können Sie einige Sicherheitskontrollen in Ihrer Organisation standardisieren. Beispiele hierfür sind die Integration zentraler Identitätsanbieter, die Definition von Netzwerken und Firewalls und die Konfiguration von Standardorten für die Speicherung und Analyse von Protokollen. Analog zur Anwendung von Infrastructure as Code (IaC) zur Anwendung der gleichen strikten Vorgehensweise bei der Entwicklung von Anwendungscode auf die Bereitstellung der Infrastruktur können Sie IaC auch zur Definition und Bereitstellung Ihrer Standard-Sicherheitskontrollen verwenden.

Definieren Sie Ihre Sicherheitskontrollen nach Möglichkeit deklarativ, wie z. B. in [AWS CloudFormation](#), und speichern Sie sie in einem Versionskontrollsystem. Verwenden Sie DevOps Methoden, um die Implementierung Ihrer Steuerungen zu automatisieren, um besser vorhersehbare Versionen zu erzielen. Automatisieren Sie Tests mithilfe von Tools wie [AWS CloudFormation Guard](#) und erkennen Sie Abweichungen zwischen Ihren eingesetzten Steuerungen und Ihrer gewünschten Konfiguration. Sie können Services wie [AWS CodePipeline](#), [AWS CodeBuild](#) und [AWS CodeDeploy](#) verwenden, um eine CI/CD-Pipeline zu erstellen. Beachten Sie die Hinweise unter [Organisieren Ihrer AWS Umgebung mithilfe mehrerer Konten](#), um diese Dienste in eigenen Konten zu konfigurieren, die von anderen Bereitstellungspipelines getrennt sind.

Sie können auch Vorlagen definieren, um die Definition und Bereitstellung AWS-Konten, Dienste und Konfigurationen zu standardisieren. Diese Technik ermöglicht es einem zentralen Sicherheitsteam, diese Definitionen zu verwalten und sie den Workload-Teams über einen Selfservice-Ansatz zur Verfügung zu stellen. Eine Möglichkeit, dies zu erreichen, ist die Verwendung von [Service Catalog](#), wo Sie Vorlagen als Produkte veröffentlichen können, die Workload-Teams in ihre eigenen Pipeline-Bereitstellungen einbinden können. Wenn Sie [AWS Control Tower](#) verwenden, sind einige Vorlagen und Kontrollen als Ausgangspunkt verfügbar. Control Tower bietet zudem die Funktion [Account Factory](#), mit der Workload-Teams neue AWS-Konten -Konten unter Verwendung der von Ihnen definierten Standards erstellen können. Mit dieser Funktion sind Sie nicht mehr auf ein zentrales Team angewiesen, das neue Konten genehmigt und anlegt, wenn diese von Ihren Workload-Teams als notwendig erachtet werden. Sie benötigen diese Konten möglicherweise, um verschiedene Workload-Komponenten zu isolieren, z. B. aufgrund ihrer Funktion, der Sensibilität der verarbeiteten Daten oder ihres Verhaltens.

Implementierungsschritte

1. Legen Sie fest, wie Sie Ihre Vorlagen in einem Versionskontrollsystem speichern und pflegen wollen.
2. Erstellen Sie CI/CD-Pipelines zum Testen und Bereitstellen Ihrer Vorlagen. Definieren Sie Tests, um zu prüfen, ob Fehlkonfigurationen vorliegen und ob die Vorlagen den Standards Ihres Unternehmens entsprechen.
3. Erstellen Sie einen Katalog mit standardisierten Vorlagen, die Workload-Teams entsprechend Ihren Anforderungen bereitstellen AWS-Konten und Services bereitstellen können.
4. Implementieren Sie sichere Sicherungs- und Wiederherstellungsstrategien für die Konfiguration Ihrer Kontrollen, Skripte und zugehörigen Daten.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS05-BP01 Verwenden Sie die Versionskontrolle](#)
- [OPS05-BP04 Verwenden Sie Build- und Deployment-Management-Systeme](#)
- [REL08-BP05 Implementieren Sie Änderungen mit Automatisierung](#)
- [SUS06-BP01 Wenden Sie Methoden an, mit denen schnell Verbesserungen der Nachhaltigkeit eingeführt werden können](#)

Zugehörige Dokumente:

- [Organisieren Sie Ihre AWS Umgebung mithilfe mehrerer Konten](#)

Zugehörige Beispiele:

- [Automatisieren Sie die Kontoerstellung und die Bereitstellung von Ressourcen mithilfe von Service Catalog AWS Organizations, und AWS Lambda](#)
- [Stärken Sie die DevOps Pipeline und schützen Sie Daten mit AWS Secrets Manager AWS KMS, und AWS Certificate Manager](#)

Zugehörige Tools:

- [AWS CloudFormation Guard](#)

- [Landezone Accelerator aktiviert AWS](#)

SEC01-BP07 Identifizieren von Bedrohungen und Priorisieren von Abhilfemaßnahmen unter Verwendung eines Bedrohungsmodells

Führen Sie Bedrohungsmodellierungen zur Identifizierung und Pflege eines aktuellen Registers potenzieller Bedrohungen und entsprechender Abhilfemaßnahmen für Ihre Workload durch. Priorisieren Sie Ihre Bedrohungen und passen Sie Ihre Sicherheitskontrollen an, um zu verhindern, zu erkennen und zu reagieren. Überarbeiten und halten Sie diese Methoden im Kontext Ihrer Workload und der sich entwickelnden Sicherheitslandschaft aktuell.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Was versteht man unter Bedrohungsmodellierung?

„Bedrohungsmodellierung dient der Identifizierung, Kommunikation und dem Verständnis von Bedrohungen und Abhilfemaßnahmen im Kontext des Schutzes von etwas Wertvollem.“ – [The Open Web Application Security Project \(OWASP\) über Bedrohungsmodellierung für Anwendungen](#)

Wozu dient die Bedrohungsmodellierung?

Systeme sind komplex und werden mit der Zeit immer komplexer und leistungsfähiger. Gleichzeitig liefern sie immer mehr geschäftlichen Wert und verbessern die Kundenzufriedenheit und -bindung. Dies bedeutet, dass Entscheidungen zum IT-Design immer mehr Anwendungsfälle berücksichtigen müssen. Diese Komplexität und die zunehmende Zahl der Anwendungsfälle macht unstrukturierte Konzepte ineffektiv, wenn es um das Erkennen und Bekämpfen von Bedrohungen geht. Stattdessen wird ein systematisches Konzept benötigt, das die potenziellen Bedrohungen für ein System aufführen und Abhilfemaßnahmen benennen und priorisieren kann, um sicherzustellen, dass die begrenzten Ressourcen einer Organisation in maximaler Weise in der Lage sind, die Sicherheitslage des Systems insgesamt zu verbessern.

Die Bedrohungsmodellierung dient zum Aufbau eines solchen systematischen Konzepts, damit Probleme frühzeitig im Designprozess erkannt und angegangen werden können, so lange Abhilfemaßnahmen noch mit niedrigen relativen Kosten und geringem Aufwand verbunden sind, was später im Lebenszyklus nicht mehr der Fall ist. Dieses Konzept entspricht dem Branchenprinzip des [Shift-Left-Sicherheitsansatzes](#). Letztendlich ist die Bedrohungsmodellierung in den Risikomanagementprozess einer Organisation integriert und hilft mit einem auf Bedrohungen

ausgerichteten Konzept bei Entscheidungen dazu, welche Kontrollmechanismen zu implementieren sind.

Wann sollte eine Bedrohungsmodellierung durchgeführt werden?

Beginnen Sie mit der Bedrohungsmodellierung so früh wie möglich im Lebenszyklus Ihrer Workload. Dies gibt Ihnen die benötigte Flexibilität im Umgang mit den identifizierten Bedrohungen. Wie bei Softwarebugs gilt auch hier: Je früher Sie Bedrohungen identifizieren, desto kostengünstiger ist es, sie zu beheben. Ein Bedrohungsmodell ist ein lebendiges Dokument, dass stetig weiterentwickelt werden sollte, während sich Ihre Workloads verändern. Überprüfen Sie regelmäßig Ihre Bedrohungsmodelle, vor allem bei größeren Änderungen, bei Änderungen der Bedrohungslandschaft, oder wenn Sie neue Features oder Services einführen.

Implementierungsschritte

Wie wird die Bedrohungsmodellierung durchgeführt?

Es gibt viele verschiedene Möglichkeiten zur Durchführung von Bedrohungsmodellierungen. Ähnlich wie bei Programmiersprachen gibt es Vor- und Nachteile und Sie sollten den Ansatz wählen, der für Sie am besten funktioniert. Ein Konzept besteht darin, mit [Shostack's 4 Question Frame for Threat Modeling](#) zu beginnen, das aus offenen Fragen besteht, die Ihre Bedrohungsmodellierung strukturieren:

1. Woran arbeiten wir?

Diese Frage dient dazu, das von Ihnen aufgebaute System sowie die sicherheitsrelevanten Details zu diesem System zu verstehen. Für die Beantwortung dieser Frage ist es üblich, ein Modell oder Diagramm zur Visualisierung dessen aufzustellen, was aufgebaut wird, etwa in Gestalt eines [Datenflussdiagramms](#). Das Aufschreiben von Annahmen und wichtigen Details zum System hilft ebenfalls beim Verständnis des Umfangs. Dadurch können sich alle, die zum Bedrohungsmodell beitragen, auf dasselbe konzentrieren und zeitraubende Umwege über irrelevante Themen (wie etwa veraltete Versionen des Systems) vermeiden. Wenn Sie beispielsweise eine Web-Anwendung erstellen, ist es wahrscheinlich nicht relevant, sich um die Bedrohungsmodellierung im Zusammenhang mit der Bootsequenz für Browser-Clients in vertrauenswürdigen Betriebssystemen zu kümmern, da Sie darauf ohnehin keinen Einfluss haben.

2. Was kann schief gehen?

Hier identifizieren Sie die Bedrohungen für Ihr System. Bedrohungen sind versehentliche oder beabsichtigte Handlungen oder Ereignisse, die unerwünschte Folgen haben und die Sicherheit

Ihres Systems beeinträchtigen können. Ohne ein klares Verständnis dessen, was schief gehen kann, haben Sie keine Möglichkeit, etwas dagegen zu unternehmen.

Es gibt keine kanonische Liste dessen, was schief gehen kann. Die Erstellung dieser Liste erfordert Brainstorming und die Zusammenarbeit all Ihrer Teammitglieder und der [relevanten Beteiligten](#) an der Bedrohungsmodellierung. Sie können das Brainstorming unterstützen, indem Sie ein Modell zur Identifizierung von Bedrohungen verwenden, z. B. [STRIDE](#), das verschiedene Kategorien zur Bewertung anbietet: Spoofing, Manipulation, Zurückweisung, Offenlegung von Informationen, Denial of Service und Erhöhung der Berechtigung. Dazu sollten Sie zur Inspiration vorhandene Listen und Forschungsergebnisse heranziehen, etwa die [OWASP Top 10](#), den [HiTrust Threat Catalog](#) und den eigenen Bedrohungskatalog Ihrer Organisation.

3. Wie gehen wir damit um?

Wie schon bei der vorherigen Frage gibt es auch hier keine kanonische Liste möglicher Abhilfemaßnahmen. Die Inputs für diesen Schritt sind die identifizierten Bedrohungen, Akteure und Verbesserungsbereiche aus dem vorherigen Schritt.

Sicherheit und Compliance unterliegen der [geteilten Verantwortung zwischen Ihnen und AWS](#). Der Frage „Wie gehen wir damit um?“ sollte unbedingt die Frage „Wer ist für die Maßnahmen verantwortlich?“ angeschlossen werden. Das Verständnis der Verantwortungsverteilung zwischen Ihnen und AWS hilft Ihnen bei der Anpassung der Bedrohungsmodellierung an die Abhilfemaßnahmen, die Ihrer Kontrolle unterliegen und in der Regel aus einer Kombination aus AWS-Servicekonfigurationsoptionen und Ihren eigenen systemspezifischen Abhilfemaßnahmen bestehen.

Für den AWS-Teil der geteilten Verantwortung werden Sie feststellen, dass [AWS-Services in den Bereich vieler Compliance-Programme fallen](#). Diese Programme helfen Ihnen, sich mit den zuverlässigen Kontrollmöglichkeiten bei AWS zur Sicherheitswahrung und Compliance in der Cloud vertraut zu machen. Die Audit-Berichte dieser Programme stehen für AWS-Kunden von [AWS Artifact](#) zum Download zur Verfügung.

Unabhängig davon, welche AWS-Services Sie nutzen, gibt es immer ein Element der Kundenverantwortung, und an diese Verantwortungen angepasste Abhilfemaßnahmen sollten Teil Ihres Bedrohungsmodells sein. Für Sicherheitskontrollabhilfen für die AWS-Services selbst sollten Sie die Implementierung von Sicherheitskontrollen über Domains hinweg erwägen, einschließlich Domains wie Identitäts- und Zugriffsmanagement (Authentifizierung und Autorisierung), Datenschutz (im Ruhezustand und während der Übertragung), Infrastruktursicherheit, Protokollierung und Überwachung. Die Dokumentation für jeden AWS-Service enthält ein

[spezielles Sicherheitskapitel](#) mit Anleitungen zu den Sicherheitskontrollen, die Abhilfemaßnahmen unterstützen können. Wichtig ist, dass Sie den Code, den Sie schreiben, und dessen Abhängigkeiten berücksichtigen und an Kontrollen denken, die Sie für den Umgang mit den damit verbundenen Bedrohungen implementieren können. Bei diesen Kontrollen könnte es sich um Dinge wie [Eingabevalidierung](#), [Sitzungsabwicklung](#) und [Umgang mit Grenzen](#) handeln. Oft ist der Löwenanteil der Bedrohungen mit benutzerdefiniertem Code verbunden, konzentrieren Sie sich also besonders darauf.

4. Haben wir gute Arbeit geleistet?

Ihr Team und die Organisation verfolgen das Ziel, die Qualität der Bedrohungsmodelle und die Geschwindigkeit zu verbessern, mit der Sie die Bedrohungsmodellierung im Laufe der Zeit durchführen. Diese Verbesserungen werden durch eine Kombination von Praxis, Lernen, Lehren und Prüfen ermöglicht. Um dies zu vertiefen und praktisch umzusetzen, sollten Sie und Ihr Team den Trainingskurs zum Thema [Korrekte Bedrohungsmodellierung für Builder](#) oder den dazugehörigen [Workshop](#) absolvieren. Wenn Sie nach Anleitungen zur Integration der Bedrohungsmodellierung in den Anwendungsentwicklungslebenszyklus Ihrer Organisation suchen, beachten Sie auch den Beitrag zum Thema [Bedrohungsmodellierungskonzepte](#) im AWS-Blog zu Sicherheit.

Threat Composer

Zur Unterstützung und Anleitung bei der Erstellung von Bedrohungsmodellen können Sie das [Threat Composer](#)-Tool verwenden, das darauf ausgerichtet ist, bei der Erstellung von Bedrohungsmodellen die Zeit bis zur Wertschöpfung zu verkürzen. Das Tool hilft Ihnen bei den folgenden Aufgaben:

- Verfassen nützlicher, an [Bedrohungsgrammatik](#) ausgerichtete Bedrohungsanweisungen, die in einem natürlichen, nicht-linearen Arbeitsablauf funktionieren.
- Generieren Sie ein für Menschen lesbares Bedrohungsmodell.
- Generieren Sie ein maschinenlesbares Bedrohungsmodell, damit Sie Bedrohungsmodelle wie Code behandeln können.
- Mit dem Insights-Dashboard können Sie schnell Bereiche identifizieren, in denen die Qualität und die Abdeckung verbessert werden müssen.

Für weitere Informationen rufen Sie Threat Composer auf und wechseln Sie zum systemdefinierten Beispielarbeitsbereich.

Ressourcen

Zugehörige bewährte Methoden:

- [SEC01-BP03 Identifizieren und Validieren von Kontrollzielen](#)
- [SEC01-BP04 Bleiben Sie über Sicherheitsbedrohungen und Empfehlungen auf dem Laufenden](#)
- [SEC01-BP05 Reduzieren Sie den Umfang des Sicherheitsmanagements](#)
- [SEC01-BP08 Regelmäßiges Bewerten und Implementieren neuer Sicherheitsservices und -features](#)

Zugehörige Dokumente:

- [How to approach threat modeling](#) (AWS-Blog zum Thema Sicherheit)
- [NIST: Guide to Data-Centric System Threat modeling](#)

Zugehörige Videos:

- [AWS Summit ANZ 2021 - How to approach threat modeling](#)
- [AWS Summit ANZ 2.022 - Scaling security – Optimise for fast and secure delivery](#)

Zugehörige Schulungen:

- [Threat modeling the right way for builders – virtuelle AWS Skill Builder-Schulung zum Selbststudium](#)
- [Threat modeling the right way for builders – AWS Workshop](#)

Zugehörige Tools:

- [Threat Composer](#)

SEC01-BP08 Regelmäßiges Bewerten und Implementieren neuer Sicherheitsservices und -features

Bewerten und implementieren Sie Sicherheitsservices und -features von AWS und AWS-Partnern, mit denen Sie die Sicherheitsstrategie für Ihre Workload weiterentwickeln können.

Gewünschtes Ergebnis: Sie verfügen über eine Standardmethode, die Sie über neue Features und Services informiert, die von AWS und AWS-Partnern veröffentlicht werden. Sie bewerten, wie sich

diese neuen Funktionen auf das Design der aktuellen und neuen Kontrollen für Ihre Umgebungen und Workloads auswirken.

Typische Anti-Muster:

- Sie abonnieren keine Blogs und RSS-Feeds von AWS, um schnell von relevanten neuen Features und Services zu erfahren
- Sie verlassen sich auf Nachrichten und Updates über Sicherheitsservices und Features aus zweiter Hand
- Sie halten AWS-Benutzer in Ihrer Organisation nicht dazu an, sich über die neuesten Updates zu informieren

Vorteile der Nutzung dieser bewährten Methode: Indem Sie sich über neue Sicherheitsservices und Features auf dem Laufenden halten, können Sie fundierte Entscheidungen über die Implementierung von Kontrollen in Ihren Cloud-Umgebungen und Workloads treffen. Diese Quellen tragen dazu bei, das Bewusstsein für die sich entwickelnde Sicherheitslandschaft zu schärfen und zu zeigen, wie AWS-Services zum Schutz vor neuen und aufkommenden Bedrohungen genutzt werden können.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Niedrig

Implementierungsleitfaden

AWS informiert Kunden über neue Sicherheitsservices und Features über verschiedene Kanäle:

- [AWS Neuerungen bei](#)
- [AWS News Blog](#)
- [AWS Blog zum Thema Sicherheit](#)
- [AWS Sicherheitsberichte](#)
- [AWS Überblick über die -Dokumentation](#)

Sie können ein Thema der [AWS Daily Feature Updates](#) mit Amazon Simple Notification Service (Amazon SNS) abonnieren, um eine umfassende tägliche Zusammenfassung der Updates zu erhalten. Einige Sicherheitsservices wie [Amazon GuardDuty](#) und [AWS Security Hub CSPM](#) bieten ihre eigenen SNS-Themen an, um über neue Standards, Erkenntnisse und andere Aktualisierungen für diese speziellen Services informiert zu bleiben.

Neue Services und Features werden auch auf [Konferenzen, Veranstaltungen und Webinaren](#), die jedes Jahr rund um den Globus stattfinden, angekündigt und im Detail beschrieben. Besonders

interessant ist dabei die jährliche Sicherheitskonferenz [AWS re:Inforce](#) und die breiter angelegte Konferenz [AWS re:Invent](#). In den bereits erwähnten AWS-Nachrichtenkanälen werden diese Konferenzankündigungen über Sicherheit und andere Services geteilt, und Sie können sich Deep Dive Breakout Sessions online auf dem YouTube-Kanal [AWS Events](#) ansehen.

Sie können auch Ihr [AWS-Konto-Team](#) nach den neuesten Updates und Empfehlungen für Sicherheitsservices fragen. Sie können Ihr Team über das [Verkaufssupport-Formular](#) erreichen, wenn Ihnen dessen direkte Kontaktinformationen nicht vorliegen. Gleichmaßen erhalten Sie, wenn Sie [AWS Enterprise-Support](#) abonniert haben, wöchentliche Updates von Ihrem Technical Account Manager (TAM) und können ein regelmäßiges Review-Meeting mit ihm vereinbaren.

Implementierungsschritte

1. Abonnieren Sie die verschiedenen Blogs und Bulletins mit Ihrem bevorzugten RSS-Reader oder die SNS-Thema Daily Features Updates.
2. Überlegen Sie, welche AWS-Veranstaltungen Sie besuchen sollten, um sich aus erster Hand über neue Features und Services zu informieren.
3. Vereinbaren Sie Besprechungen mit Ihrem AWS-Konto-Team für alle Fragen zur Aktualisierung von Sicherheitsservices und -features.
4. Ziehen Sie in Erwägung, den Enterprise Support zu abonnieren, um regelmäßige Konsultationen mit einem Technical Account Manager (TAM) zu erhalten.

Ressourcen

Zugehörige bewährte Methoden:

- [PERF01-BP01 Informieren über verfügbare Cloud-Services und -Features](#)
- [COST01-BP07 Verfolgen neuer Serviceversionen](#)

Identity and Access Management

Fragen

- [SEC 2. Was ist bei der Verwaltung der Authentifizierung für Personen und Rechner zu beachten?](#)
- [SEC 3. Wie werden Berechtigungen für Personen und Computer verwaltet?](#)

SEC 2. Was ist bei der Verwaltung der Authentifizierung für Personen und Rechner zu beachten?

Es gibt zwei Arten von Identitäten, die Sie für den Betrieb von sicheren AWS-Workloads verwalten müssen.

- **Menschliche Identitäten:** Die menschlichen Identitäten, die Zugriff auf Ihre AWS-Umgebungen und -Anwendungen benötigen, können in drei Gruppen eingeteilt werden: Belegschaft, Dritte und Benutzer.

Zur Gruppe Belegschaft gehören Administratoren, Entwickler und Betreiber, die Mitglieder Ihrer Organisation sind. Diese benötigen Zugriff, um Ihre AWS-Ressourcen verwalten, erstellen und betreiben zu können.

Dritte sind externe Mitarbeiter wie Auftragnehmer, Anbieter oder Partner. Diese interagieren im Rahmen ihrer Zusammenarbeit mit Ihrer Organisation mit Ihren AWS-Ressourcen.

Benutzer sind die Nutzer Ihrer Anwendungen. Diese greifen über Webbrowser, Client-Anwendungen, Mobil-Apps oder interaktive Befehlszeilentools auf Ihre AWS-Ressourcen zu.

- **Maschinenidentitäten:** Ihre Workload-Anwendungen, betrieblichen Tools und Komponenten benötigen eine Identität, um Anforderungen an AWS-Services zu stellen, z. B. um Daten zu lesen. Zu diesen Identitäten gehören auch Maschinen, die in Ihrer AWS-Umgebung ausgeführt werden, wie z. B. Amazon-EC2-Instances oder AWS Lambda-Funktionen. Sie können auch Maschinenidentitäten für externe Parteien oder Maschinen außerhalb von AWS verwalten, die Zugriff auf Ihre AWS-Umgebung benötigen.

Best Practices

- [SEC02-BP01 Verwenden von starken Anmeldemechanismen](#)
- [SEC02-BP02 Verwenden von temporären Anmeldeinformationen](#)
- [SEC02-BP03 Sicheres Speichern und Verwenden von Secrets](#)
- [SEC02-BP04 Verlassen auf einen zentralen Identitätsanbieter](#)
- [SEC02-BP05 Regelmäßiges Überprüfen und Rotieren von Anmeldeinformationen](#)
- [SEC02-BP06 Nutzen von Benutzergruppen und Attributen](#)

SEC02-BP01 Verwenden von starken Anmeldemechanismen

Anmeldungen (Authentifizierung unter Verwendung von Anmeldeinformationen) können risikobehaftet sein, wenn nicht Mechanismen wie die Multi-Faktor-Authentifizierung (MFA) verwendet werden, besonders in Situationen, in denen Anmeldeinformationen unbeabsichtigt offengelegt wurden oder leicht zu erraten sind. Verwenden Sie starke Anmeldemechanismen in Form von MFA und Richtlinien für sichere Passwörter, um diese Risiken zu reduzieren.

Gewünschtes Ergebnis: Reduzieren Sie das Risiko eines unbeabsichtigten Zugriffs auf Anmeldeinformationen in AWS, indem Sie starke Anmeldemechanismen für [AWS Identity and Access Management \(IAM\)](#)-Benutzer, den [AWS-KontoRoot-Benutzer](#), [AWS IAM Identity Center](#) und externe Identitätsanbieter verwenden. Dies bedeutet das Erfordern von MFA, das Durchsetzen von Richtlinien zur Verwendung starker Passwörter und das Erkennen anomaler Anmeldeverhaltensweisen.

Typische Anti-Muster:

- Keine Durchsetzung einer Richtlinie zur Verwendung starker Passwörter für Ihre Identitäten, einschließlich komplexer Passwörter und MFA.
- Gemeinsame Nutzung derselben Anmeldeinformationen durch mehrere Benutzer.
- Keine Verwendung von detektivischen Kontrollen für verdächtige Anmeldevorgänge.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Es gibt mehrere Möglichkeiten zur Anmeldung bei AWS für menschliche Identitäten. Eine bewährte AWS-Methode besteht darin, einen zentralisierten Identitätsanbieter mit Verbundverfahren (direkter SAML-2.0-Verbund zwischen AWS IAM und dem zentralisierten Identitätsanbieter oder Verwendung von AWS IAM Identity Center) für die Authentifizierung bei AWS zu verwenden. Richten Sie in diesem Fall einen sicheren Anmeldevorgang mit Ihrem Identitätsanbieter oder Microsoft Active Directory ein.

Wenn Sie ein AWS-Konto zum ersten Mal einrichten, beginnen Sie mit einem Root-Benutzer für das AWS-Konto. Sie sollten den Root-Benutzer nur verwenden, um den Zugriff für Ihre Benutzer einzurichten (und für [Aufgaben, für die der Root-Benutzer erforderlich ist](#)). Es ist wichtig, die Multi-Faktor-Authentifizierung (MFA) für den Root-Benutzer des Kontos sofort nach dem Öffnen Ihres AWS-Kontos zu aktivieren und den Root-Benutzer unter Berücksichtigung des [AWS-Leitfadens für bewährte Methoden](#) zu sichern.

AWS IAM Identity Center wurde für Belegschaftsbenutzer konzipiert. Sie können Benutzeridentitäten innerhalb des Service erstellen und verwalten und den Anmeldevorgang mit MFA sichern. AWS Cognito dagegen wurde für Customer Identity and Access Management (CIAM) entwickelt, das Benutzerpools und Identitätsanbieter für externe Benutzeridentitäten in Ihren Anwendungen bereitstellt.

Wenn Sie in AWS IAM Identity Center Benutzer erstellen, sollten Sie den Anmeldevorgang in diesem Service schützen und [MFA aktivieren](#). Für externe Benutzeridentitäten in Ihren Anwendungen können Sie [Amazon-Cognito-Benutzerpools](#) verwenden und den Anmeldevorgang in diesem Service oder über einen der unterstützten Identitätsanbieter in Amazon-Cognito-Benutzerpools sichern.

Darüber hinaus können Sie für Benutzer in AWS IAM Identity Center unter Verwendung von [AWS Verified Access](#) eine zusätzliche Sicherheitsebene bereitstellen, indem Sie die Identität der Benutzers und den Gerätestatus überprüfen, bevor ihnen Zugriff auf AWS-Ressourcen gewährt wird.

Wenn Sie [AWS Identity and Access Management \(IAM\)](#)-Benutzer verwenden, sichern Sie den Anmeldevorgang mit IAM.

Sie können AWS IAM Identity Center und den direkten IAM-Verbund gleichzeitig verwenden, um den Zugriff auf AWS zu verwalten. Sie können den IAM-Verbund für die Verwaltung des Zugriffs auf die AWS-Managementkonsole und die Services und IAM Identity Center für die Verwaltung des Zugriffs auf Geschäftsanwendungen wie QuickSight oder Amazon Q Business verwenden.

Unabhängig vom Anmeldeverfahren ist es wichtig, eine strenge Anmelderichtlinie durchzusetzen.

Implementierungsschritte

Es folgen allgemeine Empfehlungen für starke Anmeldeverfahren. Die tatsächlichen Einstellungen, die Sie konfigurieren, sollten Ihren Unternehmensrichtlinien oder einem Standard wie [NIST 800-63](#) entsprechen.

- MFA erforderlich. Es ist eine [bewährte IAM-Methode, MFA für menschliche Identitäten und Workloads zu erfordern](#). Die Aktivierung von MFA bietet eine zusätzliche Sicherheitsebene, die verlangt, dass Benutzer Anmeldeinformationen und ein Einmalpasswort (OTP) oder eine kryptographisch verifizierte und generierte Zeichenfolge von einem Hardware-Gerät vorlegen.
- Verlangen Sie eine Mindestlänge für Passwörter als primären Faktor für die Passwortstärke.
- Verlangen Sie Passwortkomplexität, um das Erraten von Passwörtern zu erschweren.
- Ermöglichen Sie Benutzern, ihre eigenen Passwörter zu ändern.

- Erstellen Sie individuelle Identitäten anstelle gemeinsam genutzter Anmeldeinformationen. Da Sie individuelle Identitäten erstellen, können Sie jedem Benutzer eindeutige Anmeldeinformationen zuordnen. Individuelle Benutzer bieten die Möglichkeit, die Aktivität der einzelnen Benutzer zu prüfen.

Empfehlungen für IAM Identity Center:

- IAM Identity Center bietet bei Verwendung des Standardverzeichnisses eine vordefinierte [Passwortrichtlinie](#), die Anforderungen an die Länge, Komplexität und Wiederverwendung von Passwörtern festlegt.
- [Aktivieren Sie MFA](#) und konfigurieren Sie die kontextsensitive oder „always-on“-Einstellung für MFA, wenn die Identitätsquelle das Standardverzeichnis, AWS Managed Microsoft AD oder AD Connector ist.
- Erlauben Sie Benutzern, [ihre eigenen MFA-Geräte zu registrieren](#).

Empfehlungen für Verzeichnisse der Amazon Cognito-Benutzerpools:

- Konfigurieren Sie die Einstellungen für die [Passwortstärke](#).
- [Erfordern Sie MFA](#) für Benutzer.
- Verwenden Sie die [erweiterten Sicherheitseinstellungen](#) der Amazon Cognito-Benutzerpools für Features wie die [adaptive Authentifizierung](#), mit der verdächtige Anmeldungen blockiert werden können.

Empfehlungen für IAM-Benutzer:

- Idealerweise verwenden Sie IAM Identity Center oder den direkten Verbund. Möglicherweise benötigen Sie aber auch IAM-Benutzer. [Legen Sie in diesem Fall eine Passwortrichtlinie für IAM-Benutzer fest](#). Sie können die Passwortrichtlinie verwenden, um Anforderungen wie die Mindestlänge zu definieren oder ob das Passwort nicht-alphanumerische Zeichen beinhalten sollte.
- Erstellen Sie eine IAM-Richtlinie, um [die MFA-Anmeldung zu erzwingen](#), sodass Benutzer ihre eigenen Passwörter und MFA-Geräte verwalten können.

Ressourcen

Zugehörige bewährte Methoden:

- [SEC02-BP03 Sicheres Speichern und Verwenden von Secrets](#)
- [SEC02-BP04 Verlassen auf einen zentralen Identitätsanbieter](#)
- [SEC03-BP08 Sicheres gemeinsames Nutzen von Ressourcen in Ihrer Organisation](#)

Zugehörige Dokumente:

- [Passwortrichtlinie von AWS IAM Identity Center](#)
- [Passwortrichtlinie für IAM-Benutzer](#)
- [Festlegen des Passworts des AWS-Konto-Root-Benutzers](#)
- [Amazon-Cognito-Passwortrichtlinie](#)
- [AWS-Anmeldeinformationen](#)
- [Bewährte IAM-Sicherheitsmethoden](#)

Zugehörige Videos:

- [Verwaltung von Benutzerberechtigungen in großem Umfang mit AWS IAM Identity Center](#)
- [Beherrschen der Identität auf jeder Ebene](#)

SEC02-BP02 Verwenden von temporären Anmeldeinformationen

Bei Authentifizierungen jeder Art, sollten am besten temporäre anstelle langfristiger Anmeldeinformationen verwendet werden, um Risiken zu reduzieren oder zu eliminieren, etwa durch die unbeabsichtigte Offenlegung, die Weitergabe oder den Diebstahl von Anmeldeinformationen.

Gewünschtes Ergebnis: Um das Risiko langfristiger Anmeldeinformationen zu verringern, sollten Sie nach Möglichkeit sowohl für menschliche als auch für maschinelle Identitäten temporäre Anmeldeinformationen verwenden. Langfristige Anmeldeinformationen sind mit vielen Risiken verbunden. So kann es beispielsweise zu einer Offenlegung durch Uploads in öffentliche Repositories kommen. Durch die Verwendung temporärer Anmeldeinformationen können Sie die Gefahr, dass Anmeldeinformationen kompromittiert werden, deutlich senken.

Typische Anti-Muster:

- Entwickler verwenden langfristige Zugriffsschlüssel von IAM-Benutzern, anstatt sich temporäre Anmeldeinformationen per Verbund von der CLI zu beschaffen.

- Entwickler betten langfristige Zugriffsschlüssel in ihren Code ein und laden diese in öffentliche Git-Repositorys hoch.
- Entwickler betten langfristige Zugriffsschlüssel in Mobil-Apps ein, die dann in App-Stores verfügbar gemacht werden.
- Benutzer geben langfristige Zugriffsschlüssel an andere Benutzer weiter, oder Mitarbeiter verlassen das Unternehmen und besitzen weiterhin langfristige Zugriffsschlüssel.
- Es werden langfristige Zugriffsschlüssel für Maschinenidentitäten genutzt, obwohl temporäre Anmeldeinformationen verwendet werden könnten.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Verwenden Sie temporäre anstelle langfristiger Anmeldeinformationen für alle AWS-API- und -CLI-Anfragen. API- und CLI-Anfragen an AWS-Services müssen in fast allen Fällen mit [AWS-Zugriffsschlüsseln](#) signiert werden. Diese Anfragen können mit temporären oder langfristigen Anmeldeinformationen signiert werden. Langfristige Anmeldeinformationen – auch als langfristige Zugriffsschlüssel bezeichnet – sollten Sie nur verwenden, wenn Sie einen [IAM-Benutzer](#) oder den [AWS-Konto-Root-Benutzer](#) verwenden. Wenn Sie auf andere Weise einen Verbund mit AWS herstellen oder eine [IAM-Rolle](#) übernehmen, werden temporäre Anmeldeinformationen generiert. Selbst wenn Sie mit Anmeldeinformationen auf die AWS-Managementkonsole zugreifen, werden für Sie temporäre Anmeldeinformationen für Aufrufe von AWS-Services generiert. Es gibt nur wenige Situationen, in denen Sie langfristige Anmeldeinformationen benötigen, und fast alle Aufgaben lassen sich mit temporären Anmeldeinformationen erledigen.

Das Vermeiden der Verwendung langfristiger zugunsten temporärer Anmeldeinformationen sollte von einer Strategie zur Reduzierung der Verwendung von IAM-Benutzern gegenüber Verbundverfahren und IAM-Rollen begleitet werden. Zwar wurden früher IAM-Benutzer für menschliche und maschinelle Identitäten verwendet, wir empfehlen heute jedoch, dies nicht mehr zu tun, um die mit der Verwendung langfristiger Zugriffsschlüssel verbundenen Risiken zu vermeiden.

Implementierungsschritte

Menschliche Identitäten

Für Identitäten der Belegschaft wie Mitarbeiter, Administratoren, Entwickler und Bediener:

- Wir empfehlen Ihnen, [sich auf einen zentralen Identitätsanbieter zu verlassen](#) und [menschliche Benutzer aufzufordern, den Verbund mit einem Identitätsanbieter zu verwenden, um mit](#)

[temporären Anmeldeinformationen auf AWS zuzugreifen](#). Der Verbund für Ihre Benutzer kann entweder in Form eines [direkten Verbunds mit jedem AWS-Konto](#) oder unter Verwendung von [AWS IAM Identity Center](#) und dem Identitätsanbieter Ihrer Wahl erfolgen. Ein Verbund bietet eine Reihe von Vorteilen gegenüber der Verwendung von IAM-Benutzern und eliminiert langfristige Anmeldeinformationen. Ihre Benutzer können auch temporäre Anmeldeinformationen über die Befehlszeile für den [direkten Verbund](#) oder mithilfe von [IAM Identity Center](#) anfordern. Dies bedeutet, dass es nur wenige Anwendungsfälle gibt, für die IAM-Benutzer oder langfristige Anmeldeinformationen für Ihre Benutzer erforderlich sind.

Für externe Identitäten:

- Wenn Sie Dritten – etwa Software as a Service (SaaS)-Anbietern – Zugriff auf Ressourcen in Ihrem AWS-Konto gewähren, können Sie [kontoübergreifende Rollen](#) und [ressourcenbasierte Richtlinien](#) verwenden. Darüber hinaus können Sie den Flow zum Erteilen von Client-Anmeldeinformationen in [Amazon Cognito OAuth 2.0](#) für B2B-SaaS-Kunden oder -Partner verwenden.

Benutzeridentitäten, die über Web-Browser, Client-Anwendungen, mobile Apps oder interaktive Befehlszeilentools auf Ihre AWS-Ressourcen zugreifen:

- Wenn Sie Anwendungen für Verbraucher oder Kunden Zugriff auf Ihre AWS-Ressourcen gewähren müssen, können Sie [Amazon Cognito-Identitätspools](#) oder [Amazon Cognito-Benutzerpools](#) verwenden, um temporäre Anmeldeinformationen bereitzustellen. Die Berechtigungen für die Anmeldeinformationen werden über IAM-Rollen konfiguriert. Darüber hinaus können Sie eine separate IAM-Rolle mit beschränkten Berechtigungen für Gastbenutzer anlegen, die nicht authentifiziert wurden.

Maschinenidentitäten

Für Maschinenidentitäten müssen Sie möglicherweise langfristige Anmeldeinformationen verwenden. In diesen Fällen sollten Sie [Workloads auffordern, temporäre Anmeldeinformationen mit IAM-Rollen für den Zugriff auf AWS zu verwenden](#).

- Für [Amazon Elastic Compute Cloud](#) (Amazon EC2) können Sie [Rollen für Amazon EC2](#) verwenden.
- [AWS Lambda](#) ermöglicht es Ihnen, eine [Lambda-Ausführungsrolle zu konfigurieren, um dem Service Berechtigungen zu gewähren](#), die die Ausführung von AWS-Aktionen mit temporären

Anmeldeinformationen erlauben. Es gibt zahlreiche ähnliche Modelle für AWS-Services zum Gewähren temporärer Anmeldeinformationen mit IAM-Rollen.

- Für IoT-Geräte können Sie den [AWS IoT Core-Anmeldeinformationsanbieter](#) verwenden, um temporäre Anmeldeinformationen anzufordern.
- Für On-Premises-Systeme oder Systeme, die außerhalb von AWS ausgeführt werden und Zugriff auf AWS-Ressourcen benötigen, können Sie [IAM Roles Anywhere](#) verwenden.

Es gibt Szenarien, in denen temporäre Anmeldeinformationen nicht unterstützt werden und langfristige Anmeldeinformationen verwendet werden müssen. In diesen Situationen sollten [diese Anmeldeinformationen regelmäßig überprüft und rotiert](#) und [Zugriffsschlüssel regelmäßig rotiert](#) werden. Bei stark eingeschränkten Zugriffsschlüsseln für IAM-Benutzer sollten Sie die folgenden zusätzlichen Sicherheitsmaßnahmen in Betracht ziehen:

- Erteilung stark eingeschränkter Berechtigungen:
 - Halten Sie sich an das Prinzip der geringsten Berechtigung (machen Sie konkrete Angaben zu Aktionen, Ressourcen und Bedingungen).
 - Erwägen Sie, dem IAM-Benutzer nur die Operation „AssumeRole“ für eine bestimmte Rolle zu gewähren. Abhängig von der On-Premises-Architektur hilft dieser Ansatz, die langfristigen IAM-Anmeldeinformationen zu isolieren und zu sichern.
- Beschränken Sie die zulässigen Netzwerkquellen und IP-Adressen in der Vertrauensrichtlinie für IAM-Rollen.
- Überwachen Sie die Nutzung und richten Sie Warnmeldungen bei ungenutzten Berechtigungen oder missbräuchlicher Verwendung ein (unter Verwendung der Metrikfilter und Alarme von AWS CloudWatch Logs).
- Setzen Sie [Berechtigungsgrenzen](#) durch (Service-Kontrollrichtlinien (SCPs) und Berechtigungsgrenzen ergänzen sich gegenseitig – SCPs sind weniger stark differenziert, Berechtigungsgrenzen dagegen stärker differenziert).
- Implementieren Sie einen Prozess zur Bereitstellung und sicheren Speicherung der Anmeldeinformationen (in einem On-Premises-Tresor).

Einige weitere Optionen für Szenarien, in denen langfristige Anmeldeinformationen erforderlich sind:

- Erstellen Ihrer eigenen API für die Token-Vergabe (mit Amazon API Gateway).
- In Situationen, in denen Sie langfristige Anmeldeinformationen oder andere Anmeldeinformationen als AWS-Zugriffsschlüssel verwenden müssen (z. B. Datenbankmeldungen), können Sie

einen Service verwenden, der für die Verwaltung von Secrets konzipiert ist, wie etwa [AWS Secrets Manager](#). Secrets Manager vereinfacht die Verwaltung, Rotation und sichere Speicherung verschlüsselter Secrets. Viele AWS-Services unterstützen eine [direkte Integration](#) mit Secrets Manager.

- Für Multi-Cloud-Integrationen können Sie einen Identitätsverbund auf der Grundlage Ihrer Quell-CSP-Anmeldeinformationen (CSP = Credential Service Provider) verwenden (siehe [AWS STS AssumeRoleWithWebIdentity](#)).

Weitere Informationen zum Austauschen von langfristigen Anmeldeinformationen finden Sie unter [Rotieren der Zugriffsschlüssel](#)

Ressourcen

Zugehörige bewährte Methoden:

- [SEC02-BP03 Sicheres Speichern und Verwenden von Secrets](#)
- [SEC02-BP04 Verlassen auf einen zentralen Identitätsanbieter](#)
- [SEC03-BP08 Sicheres gemeinsames Nutzen von Ressourcen in Ihrer Organisation](#)

Zugehörige Dokumente:

- [Temporäre Sicherheitsanmeldeinformationen](#)
- [AWS Anmeldedaten](#)
- [Bewährte IAM-Sicherheitsmethoden](#)
- [IAM-Rollen](#)
- [IAM Identity Center](#)
- [Identitätsanbieter und Verbund](#)
- [Rotieren der Zugriffsschlüssel](#)
- [Partnerlösungen im Bereich Sicherheit: Zugriff und Zugriffssteuerung](#)
- [Der Stammbenutzer des AWS-Kontos](#)
- [Zugriff auf AWS über eine native Workload-Identität der Google Cloud Platform](#)
- [Zugriff auf AWS-Ressourcen von Mandanten von Microsoft Entra ID mit AWS -Security-Token-Service](#)

Zugehörige Videos:

- [Verwaltung von Benutzerberechtigungen in großem Umfang mit AWS IAM Identity Center](#)
- [Beherrschen der Identität auf jeder Ebene](#)

SEC02-BP03 Sicheres Speichern und Verwenden von Secrets

Eine Workload muss ihre Identität automatisch gegenüber Datenbanken, Ressourcen und Services von Drittanbietern authentifizieren können. Dazu dienen geheime Zugriffsanmeldeinformationen wie etwa API-Zugriffsschlüssel, Passwörter und OAuth-Tokens. Die Verwendung eines dedizierten Services zur Speicherung, Verwaltung und Rotation der Anmeldeinformationen hilft dabei, die Gefahr der Kompromittierung dieser Anmeldeinformationen zu verringern.

Gewünschtes Ergebnis: Implementierung eines Mechanismus zur sicheren Verwaltung von Anmeldeinformationen für Anwendungen, mit dem die folgenden Ziele erreicht werden:

- Identifikation der für die Workload erforderlichen Secrets
- Reduzierung der Anzahl der erforderlichen langfristigen Anmeldeinformationen durch ihren Austausch gegen kurzfristige Anmeldeinformationen, wo dies möglich ist
- Einrichtung der sicheren Speicherung und der automatischen Rotation der verbleibenden langfristigen Anmeldeinformationen
- Überwachung des Zugriffs auf in der Workload vorhandene Secrets
- Kontinuierliche Beobachtung, um sicherzustellen, dass im Rahmen des Entwicklungsprozesses keine Secrets in den Quellcode eingebettet werden
- Reduzieren der Gefahr unbeabsichtigter Offenlegungen von Anmeldeinformationen

Typische Anti-Muster:

- Keine Rotation der Anmeldeinformationen
- Speichern langfristiger Anmeldeinformationen in Quellcode oder Konfigurationsdateien
- Speichern von Anmeldeinformationen im Ruhezustand ohne Verschlüsselung

Vorteile der Nutzung dieser bewährten Methode:

- Secrets werden im Ruhezustand und während der Übertragung verschlüsselt gespeichert.

- Der Zugriff auf Anmeldeinformationen erfolgt über eine API (stellen Sie sich das als Automaten zum Verkauf von Anmeldeinformationen vor).
- Der Zugriff (Lese- und Schreibzugriff) auf Anmeldeinformationen wird geprüft und protokolliert.
- Trennung möglicher Problemquellen: Die Rotation der Anmeldeinformationen wird von einer separaten Komponente vorgenommen, die vom Rest der Architektur isoliert werden kann.
- Secrets werden automatisch bei Bedarf an Softwarekomponenten verteilt und die Rotation erfolgt an einem zentralen Ort.
- Der Zugriff auf Anmeldeinformationen kann detailliert kontrolliert werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Früher wurden Anmeldeinformationen für die Authentifizierung bei Datenbanken, APIs von Dritten, Tokens und andere Secrets möglicherweise in eingebettetem Quellcode oder in Umgebungsdateien gespeichert. AWS bietet mehrere Mechanismen, um diese Anmeldeinformationen sicher zu speichern, sie automatisch zu rotieren und ihre Verwendung zu prüfen.

Das beste Verfahren für die Verwaltung von Secrets besteht darin, den Anweisungen zum Entfernen, Ersetzen und Rotieren zu folgen. Die sichersten Anmeldeinformationen sind diejenigen, die Sie nicht speichern, verwalten oder handhaben müssen. Möglicherweise gibt es Anmeldeinformationen, die für die Funktion der Workload nicht mehr benötigt werden und sicher entfernt werden können.

Bei Anmeldeinformationen, die für die korrekte Funktion der Workload weiterhin benötigt werden, besteht die Möglichkeit, langfristige Anmeldeinformationen durch temporäre oder kurzfristige zu ersetzen. So könnten Sie beispielsweise anstelle der Hartkodierung eines geheimen AWS-Zugriffsschlüssels diese langfristigen Anmeldeinformationen durch temporäre unter Verwendung von IAM-Rollen ersetzen.

Manche langfristigen Secrets können möglicherweise nicht entfernt oder ersetzt werden. Diese Secrets können in einem Service wie [AWS Secrets Manager](#) gespeichert werden, wo sie zentral gespeichert, verwaltet und regelmäßig rotiert werden können.

Eine Prüfung des Quellcodes und der Konfigurationsdateien des Workloads kann verschiedene Arten von Anmeldeinformationen aufdecken. Die folgende Tabelle fasst Strategien für den Umgang mit gängigen Arten von Anmeldeinformationen zusammen:

Anmeldeinformationstyp	Beschreibung	Empfohlene Strategie
IAM-Zugriffsschlüssel	AWS-IAM-Zugriff und geheime Schlüssel, die zur Übernahme von IAM-Rollen innerhalb einer Workload verwendet werden	Ersetzen: Verwenden Sie stattdessen IAM-Rollen , die den Datenverarbeitungs-Instances zugewiesen sind (z. B. Amazon EC2 oder AWS Lambda). Fragen Sie zwecks Interoperabilität mit Drittanbietern, die Zugriff auf Ressourcen in Ihrem AWS-Konto benötigen, ob diese den kontoübergreifenden AWS-Zugriff unterstützen. Erwägen Sie für mobile Apps die Verwendung temporärer Anmeldeinformationen über Amazon-Cognito-Identitätspools (Verbundidentitäten) . Für Workloads, die außerhalb von AWS ausgeführt werden, sollten Sie IAM Roles Anywhere oder AWS Systems Manager Hybrid Activations in Betracht ziehen. Beachten Sie für Container die Informationen unter IAM-Rolle für Amazon-ECS-Aufgaben oder IAM-Rolle für Amazon-EKS-Knoten .
SSH-Schlüssel	Private Secure-Shell-Schlüssel, mit denen Sie sich manuell oder im Rahmen eines automatisierten	Ersetzen: Verwenden Sie AWS Systems Manager oder EC2 Instance Connect , um mithilfe von IAM-Rollen programmgesteuerten und

Anmeldeinformationstyp	Beschreibung	Empfohlene Strategie
	Prozesses bei Linux EC2-Instances anmelden können	menschlichen Zugriff auf EC2-Instances zu ermöglichen.
Anwendungs- und Datenbank anmeldeinformationen	Passwörter – einfache Textzeichenfolge	Rotation: Speichern Sie Anmeldeinformationen in AWS Secrets Manager und richten Sie nach Möglichkeit eine automatische Rotation ein.
Amazon-RDS- und Aurora-Administratordatenbank-Anmeldeinformationen	Passwörter – einfache Textzeichenfolge	Ersetzen: Verwenden Sie die Secrets Manager-Integration mit Amazon RDS oder Amazon Aurora . Darüber hinaus können einige RDS-Datenbanktypen in einigen Anwendungsfällen IAM-Rollen anstelle von Passwörtern verwenden. Weitere Informationen hierzu finden Sie unter IAM-Datenbankauthentifizierung .
OAuth-Token	Geheime Token – einfache Textzeichenfolge	Rotation: Speichern Sie Token in AWS Secrets Manager und konfigurieren Sie die automatische Rotation.
API-Token und Schlüssel	Geheime Token – einfache Textzeichenfolge	Rotation: Speichern Sie diese Daten in AWS Secrets Manager und richten Sie nach Möglichkeit eine automatische Rotation ein.

Ein typisches Anti-Muster ist die Einbettung von IAM-Zugriffsschlüsseln in Quellcode, Konfigurationsdateien oder Mobil-Apps. Wenn für die Kommunikation mit einem AWS-

Service ein IAM-Zugriffsschlüssel erforderlich ist, verwenden Sie [temporäre \(kurzfristige\) Sicherheitsanmeldeinformationen](#). Diese kurzfristigen Anmeldeinformationen können über [IAM-Rollen für EC2-Instances](#), [Ausführungsrollen](#) für Lambda-Funktionen, [Cognito-IAM-Rollen](#) für den mobilen Benutzerzugriff und [IoT-Core-Richtlinien](#) für IoT-Geräte bereitgestellt werden. Wenn Sie Schnittstellen zu Drittanbietern nutzen, sollten Sie eher [den Zugriff auf eine IAM-Rolle mit dem erforderlichen Zugriff auf die Ressourcen Ihres Kontos delegieren](#), als einen IAM-Benutzer zu konfigurieren und dem Drittanbieter den geheimen Zugriffsschlüssel für diesen Benutzer zu senden.

Es gibt viele Fälle, in denen die Workload die Speicherung von Secrets erfordert, die für die Zusammenarbeit mit anderen Services und Ressourcen erforderlich sind. [AWS Secrets Manager](#) wurde speziell für die sichere Verwaltung dieser Anmeldeinformationen sowie für die Speicherung, Verwendung und Rotation von API-Token, Passwörtern und anderen Anmeldeinformationen entwickelt.

AWS Secrets Manager bietet fünf wichtige Funktionen, um die sichere Speicherung und Verarbeitung vertraulicher Anmeldeinformationen zu gewährleisten: [Verschlüsselung im Ruhezustand](#), [Verschlüsselung während der Übertragung](#), [umfassende Prüfungen](#), [detaillierte Zugriffskontrolle](#) und [erweiterbare Rotation von Anmeldeinformationen](#). Andere Secret-Verwaltungsservices von AWS-Partnern oder lokal entwickelte Lösungen mit ähnlichen Funktionen und Sicherungen sind ebenfalls akzeptabel.

Wenn Sie ein Secret abrufen, können Sie die clientseitigen Caching-Komponenten von Secrets Manager verwenden, um es für die zukünftige Verwendung zwischenspeichern. Das Abrufen eines gecachten Secrets ist schneller als das Abrufen aus Secrets Manager. Da für den Aufruf von Secrets-Manager-APIs Kosten anfallen, kann die Verwendung eines Caches zudem Ihre Kosten senken. Alle Möglichkeiten zum Abrufen von Secrets finden Sie unter [Abrufen von Secrets](#).

Note

Bei einigen Sprachen müssen Sie möglicherweise Ihre eigene In-Memory-Verschlüsselung für das clientseitige Caching implementieren.

Implementierungsschritte

1. Identifizieren Sie Codepfade, die hartkodierte Anmeldeinformationen enthalten, mithilfe von automatisierten Tools wie [Amazon CodeGuru](#).

- a. Scannen Sie Ihre Code-Repositorys mit Amazon CodeGuru. Sobald die Überprüfung abgeschlossen ist, filtern Sie in CodeGuru nach Type=Secrets, um problematische Codezeilen zu finden.
2. Identifizieren Sie Anmeldeinformationen, die entfernt oder ersetzt werden können.
 - a. Identifizieren Sie Anmeldeinformationen, die nicht mehr benötigt werden, und markieren Sie sie zum Entfernen.
 - b. Ersetzen Sie AWS-Geheimschlüssel, die in Quellcode eingebettet sind, durch IAM-Rollen, die mit den erforderlichen Ressourcen verbunden sind. Wenn sich ein Teil Ihres Workloads außerhalb von AWS befindet, für den Zugriff auf AWS-Ressourcen jedoch IAM-Anmeldeinformationen erforderlich sind, sollten Sie [IAM Roles Anywhere](#) oder [AWS Systems Manager Hybrid Activations](#) in Betracht ziehen.
3. Integrieren Sie für andere langfristige Secrets von Dritten, die die Rotationsstrategie erfordern, Secrets Manager in Ihren Code, um die externen Secrets zur Laufzeit abzurufen.
 - a. Die CodeGuru-Konsole kann auf der Grundlage der erkannten Anmeldeinformationen automatisch [ein Secret in Secrets Manager erstellen](#).
 - b. Integrieren Sie den Secret-Abruf von Secrets Manager in Ihren Anwendungscode.
 - i. Serverless-Lambda-Funktionen können eine sprachunabhängige [Lambda-Erweiterung](#) verwenden.
 - ii. Für EC2-Instances oder -Container stellt AWS [clientseitigen Beispielcode zum Abrufen von Secrets aus Secrets Manager](#) in verschiedenen gängigen Programmiersprachen bereit.
4. Prüfen Sie Ihre Codebasis regelmäßig und wiederholen Sie dies, um sicherzustellen, dass dem Code keine neuen Secrets hinzugefügt wurden.
 - a. Erwägen Sie die Verwendung eines Tools wie [git-secrets](#), um zu verhindern, dass neue Secrets in Ihr Quellcode-Repository geladen werden.
5. [Überwachen Sie die Aktivitäten von Secrets Manager](#) auf Anzeichen für eine unerwartete Nutzung, unangemessenen Secret-Zugriff oder Versuche, Secrets zu löschen.
6. Reduzieren Sie menschliche Interaktionen mit Anmeldeinformationen. Schränken Sie den Zugriff zum Lesen, Schreiben und Ändern von Anmeldeinformationen auf eine für diesen Zweck dedizierte IAM-Rolle ein und erlauben Sie die Übernahme dieser Rolle nur einem kleinen Teil der betrieblichen Nutzer.

Ressourcen

Zugehörige bewährte Methoden:

- [SEC02-BP02 Verwenden von temporären Anmeldeinformationen](#)
- [SEC02-BP05 Regelmäßiges Überprüfen und Rotieren von Anmeldeinformationen](#)

Zugehörige Dokumente:

- [Erste Schritte mit AWS Secrets Manager](#)
- [Identitätsanbieter und Verbund](#)
- [Amazon CodeGuru: Einführung in Secrets Detector](#)
- [Wie AWS Secrets Manager AWS Key Management Service verwendet](#)
- [Ver- und Entschlüsselung von Secrets in Secrets Manager](#)
- [Blogeinträge zu Secrets Manager](#)
- [Amazon RDS kündigt Integration mit AWS Secrets Manager an](#)

Zugehörige Videos:

- [Bewährte Methoden zum Verwalten, Abrufen und Rotieren von Secrets in großem Maßstab](#)
- [Finden von hartkodierten Secrets mit Amazon CodeGuru Secrets Detector](#)
- [Sicherung von Secrets für hybride Workloads mit AWS Secrets Manager](#)

Zugehörige Workshops:

- [Speichern, Abrufen und Verwalten von vertraulichen Anmeldeinformationen in AWS Secrets Manager](#)
- [AWS-Systems-Manager-Hybridaktivierungen](#)

SEC02-BP04 Verlassen auf einen zentralen Identitätsanbieter

Verlassen Sie sich im Zusammenhang mit Identitäten für Ihre Belegschaft (Mitarbeiter und Auftragnehmer) auf einen Identitätsanbieter, mit dem Sie Identitäten zentral verwalten können. Dadurch ist es einfacher, den Zugriff über mehrere Anwendungen und Systeme hinweg zu verwalten, da Sie den Zugriff von einem einzigen Standort aus erstellen, zuweisen, verwalten, widerrufen und überwachen.

Gewünschtes Ergebnis: Sie verfügen über einen zentralen Identitätsanbieter, mit dem Sie Benutzer im Unternehmen, Authentifizierungsrichtlinien (z. B. die Anforderung einer Multi-Faktor-

Authentifizierung, MFA) und die Autorisierung für Systeme und Anwendungen zentral verwalten (z. B. die Zuweisung von Zugriffsberechtigungen auf Grundlage der Gruppenmitgliedschaft oder der Attribute eines Benutzers). Die Benutzer in Ihrer Belegschaft melden sich beim zentralen Identitätsanbieter an und bilden einen Verbund (Single Sign-On) mit internen und externen Anwendungen, sodass sich die Benutzer nicht mehrere Anmeldeinformationen merken müssen. Ihr Identitätsanbieter ist in Ihre Personalverwaltungssysteme integriert, sodass Personaländerungen automatisch mit Ihrem Identitätsanbieter synchronisiert werden. Wenn beispielsweise jemand Ihr Unternehmen verlässt, können Sie den Zugriff auf alle Anwendungen und Systeme im Verbund (einschließlich AWS) widerrufen. Sie haben die detaillierte Auditprotokollierung in Ihrem Identitätsanbieter aktiviert und überwachen diese Protokolle auf ungewöhnliches Benutzerverhalten.

Typische Anti-Muster:

- Sie verwenden keinen Verbund mit Single-Sign-On. Die Benutzer in Ihrer Belegschaft erstellen separate Benutzerkonten und Anmeldeinformationen für mehrere Anwendungen und Systeme.
- Sie haben den Lebenszyklus von Identitäten für Benutzer in Ihrer Belegschaft nicht automatisiert, indem Sie beispielsweise Ihren Identitätsanbieter in Ihre Personalverwaltungssysteme integriert haben. Wenn ein Benutzer Ihre Organisation verlässt oder die Position wechselt, folgen Sie einem manuellen Prozess, um seine Datensätze in mehreren Anwendungen und Systemen zu löschen oder zu aktualisieren.

Vorteile der Nutzung dieser bewährten Methode: Durch die Verwendung eines zentralen Identitätsanbieters haben Sie die Möglichkeit, Benutzeridentitäten und Richtlinien für Ihre Mitarbeiter von einem zentralen Ort aus zu verwalten, Benutzern und Gruppen Zugriff auf Anwendungen zuzuweisen und die Anmeldeaktivitäten der Benutzer zu überwachen. Wenn ein Benutzer die Position wechselt, werden durch die Integration in Ihre Personalverwaltungssysteme Änderungen mit dem Identitätsanbieter synchronisiert und die ihm zugewiesenen Anwendungen und Berechtigungen werden automatisch aktualisiert. Wenn ein Benutzer Ihre Organisation verlässt, wird seine Identität automatisch im Identitätsanbieter deaktiviert, wodurch ihm der Zugriff auf Anwendungen und Systeme im Verbund entzogen wird.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Leitfaden für Benutzer in der Belegschaft, die auf AWS zugreifen Benutzer in der Belegschaft, wie z. B. Mitarbeiter und Auftragnehmer in Ihrer Organisation, benötigen möglicherweise Zugriff auf AWS über die AWS-Managementkonsole oder die AWS Command Line Interface (AWS CLI), um ihre

Aufgaben auszuführen. Sie können diesen Benutzern Zugriff auf AWS gewähren, indem Sie einen Verbund von Ihrem zentralen Identitätsanbieter zu AWS auf zwei Ebenen einrichten: ein direkter Verbund mit jedem AWS-Konto oder ein Verbund mit mehreren Konten in Ihrer [AWS-Organisation](#).

Um die Benutzer in Ihrem Unternehmen direkt mit jedem AWS-Konto zu verbinden, können Sie einen zentralen Identitätsanbieter für den Verbund mit [AWS Identity and Access Management](#) in diesem Konto verwenden. Dank der Flexibilität von IAM können Sie für jedes AWS-Konto einen separaten [SAML-2.0](#)- oder [OpenID Connect \(OIDC\)](#)-Identitätsanbieter aktivieren und Verbundbenutzerattribute für die Zugriffskontrolle verwenden. Die Benutzer in Ihrer Belegschaft verwenden ihren Webbrowser, um sich beim Identitätsanbieter anzumelden, indem sie ihre Anmeldeinformationen (wie Passwörter und MFA-Tokencodes) angeben. Der Identitätsanbieter gibt eine SAML-Zusicherung an den Browser aus, die an die Anmelde-URL der AWS-Managementkonsole gesendet wird. Dies ermöglicht den Benutzern das Single Sign-On (SSO) bei der [AWS-Managementkonsole, indem sie eine IAM-Rolle annehmen](#). Ihre Benutzer können auch temporäre AWS-API-Anmeldeinformationen für die Verwendung in der [AWS CLI](#) oder [AWS-SDKs](#) aus [AWS STS](#) abrufen, indem sie [mithilfe einer SAML-Zusicherung des Identitätsanbieters die IAM-Rolle übernehmen](#).

Um die Benutzer Ihrer Belegschaft mit mehreren Konten in Ihrer AWS-Organisation zu verbinden, können Sie mithilfe von [AWS IAM Identity Center](#) den Zugriff Ihrer Mitarbeiter auf AWS-Konten und Anwendungen zentral verwalten. Sie aktivieren Identity Center für Ihre Organisation und konfigurieren Ihre Identitätsquelle. IAM Identity Center stellt ein Standard-Identitätsquellenverzeichnis bereit, mit dem Sie Ihre Benutzer und Gruppen verwalten können. Alternativ können Sie eine externe Identitätsquelle auswählen, indem Sie mithilfe von SAML 2.0 [eine Verbindung zu Ihrem externen Identitätsanbieter herstellen](#) und Benutzer und Gruppen mithilfe von SCIM [automatisch bereitstellen](#) oder mithilfe von [Directory Service eine Verbindung zu Ihrem Microsoft AD-Verzeichnis herstellen](#). Sobald eine Identitätsquelle konfiguriert wurde, können Sie Benutzern und Gruppen Zugriff auf AWS-Konten zuweisen, indem Sie Richtlinien nach dem Prinzip der geringsten Berechtigung in Ihren [Berechtigungssätzen](#) definieren. Die Benutzer Ihrer Belegschaft können sich über Ihren zentralen Identitätsanbieter authentifizieren, um sich beim [AWS-Zugriffsportal](#) anzumelden und sich per Single Sign-On bei AWS-Konten und den ihnen zugewiesenen Cloud-Anwendungen zu authentifizieren. Ihre Benutzer können [AWS CLI v2](#) konfigurieren, um sich bei Identity Center zu authentifizieren und Anmeldeinformationen für die Ausführung von AWS CLI-Befehlen zu erhalten. Identity Center ermöglicht auch den Single-Sign-On-Zugriff auf AWS-Anwendungen wie [Amazon SageMaker AI Studio](#) und [AWS IoT-SiteWise-Monitor-Portalen](#).

Nachdem Sie die obigen Anweisungen befolgt haben, müssen die Benutzer in Ihrer Belegschaft bei der Verwaltung von Workloads in AWS für den normalen Betrieb keine IAM-Benutzer und -Gruppen mehr verwenden. Stattdessen werden Ihre Benutzer und Gruppen außerhalb von AWS verwaltet,

und Benutzer können als Verbundidentität auf AWS-Ressourcen zugreifen. Verbundidentitäten verwenden die von ihrem zentralen Identitätsanbieter definierten Gruppen. Sie sollten IAM-Gruppen, IAM-Benutzer und langlebige Benutzeranmeldeinformationen (Passwörter und Zugriffsschlüssel), die in Ihren AWS-Konten nicht mehr benötigt werden, identifizieren und entfernen. Sie können mithilfe von [IAM-Berichten zu Anmeldeinformationen nach ungenutzten Anmeldeinformationen suchen](#), [die entsprechenden IAM-Benutzer löschen](#) und [IAM-Gruppen löschen](#). Sie können auf Ihre Organisation eine [Service-Kontrollrichtlinie \(SCP\)](#) anwenden, die die Erstellung neuer IAM-Benutzer und -Gruppen verhindert, und so den Zugriff auf AWS über Verbundidentitäten erzwingen.

Note

Sie sind für die Rotation der SCIM-Zugriffstoken verantwortlich, wie in der Dokumentation zur [automatischen Bereitstellung](#) beschrieben. Darüber hinaus liegt die Rotation der Zertifikate, die Ihren Identitätsverbund unterstützen, in Ihrer Verantwortung.

Leitfaden für Benutzer Ihrer Anwendungen Sie können die Identitäten der Benutzer Ihrer Anwendungen, z. B. einer mobilen App, mithilfe von [Amazon Cognito](#) als zentralem Identitätsanbieter verwalten. Amazon Cognito ermöglicht die Authentifizierung, Autorisierung und Benutzerverwaltung für Ihre Web- und Mobil-Apps. Amazon Cognito bietet einen Identitätsspeicher, der auf Millionen von Benutzern skaliert werden kann, unterstützt den Identitätsverbund für soziale Netzwerke und Unternehmen und bietet erweiterte Sicherheitsfeatures zum Schutz Ihrer Benutzer und Ihres Unternehmens. Sie können Ihre benutzerdefinierte Web- oder Mobilanwendung in Amazon Cognito integrieren, um Ihren Anwendungen innerhalb von Minuten Benutzerauthentifizierung und Zugriffskontrolle hinzuzufügen. Amazon Cognito basiert auf offenen Identitätsstandards wie SAML und Open ID Connect (OIDC), unterstützt verschiedene Compliance-Vorschriften und lässt sich in Frontend- und Backend-Entwicklungsressourcen integrieren.

Implementierungsschritte

Schritte für Benutzer im Unternehmen, die auf AWS zugreifen

- Erstellen Sie für die Benutzer in Ihrer Belegschaft unter Verwendung eines zentralen Identitätsanbieters einen Verbund mit AWS. Nutzen Sie dabei einen der folgenden Ansätze:
 - Verwenden Sie IAM Identity Center, um Single Sign-On für mehrere AWS-Konten in Ihrer AWS-Organisation zu aktivieren, indem Sie einen Verbund mit Ihrem Identitätsanbieter erstellen.
 - Verwenden Sie IAM, um Ihren Identitätsanbieter direkt mit jedem AWS-Konto zu verbinden und so einen differenzierten Verbundzugriff zu ermöglichen.

- Identifizieren und entfernen Sie IAM-Benutzer und -Gruppen, die durch Verbundidentitäten ersetzt werden.

Schritte für Benutzer Ihrer Anwendungen

- Verwenden Sie Amazon Cognito als zentralen Identitätsanbieter für Ihre Anwendungen.
- Integrieren Sie Ihre benutzerdefinierten Anwendungen mithilfe von OpenID Connect und OAuth mit Amazon Cognito. Sie können Ihre benutzerdefinierten Anwendungen mithilfe der Amplify-Bibliotheken entwickeln, die einfache Schnittstellen für die Integration in eine Vielzahl von AWS-Services bieten, z. B. Amazon Cognito für die Authentifizierung.

Ressourcen

Zugehörige bewährte Methoden:

- [SEC02-BP06 Nutzen von Benutzergruppen und Attributen](#)
- [SEC03-BP02 Gewähren des Zugriffs mit den geringsten Berechtigungen](#)
- [SEC03-BP06 Zugriffsverwaltung basierend auf dem Lebenszyklus](#)

Zugehörige Dokumente:

- [Identitätsverbund in AWS](#)
- [Bewährte Methoden für die Sicherheit in IAM](#)
- [Bewährte Methoden für AWS Identity and Access Management](#)
- [Erste Schritte mit der delegierten Administration von IAM Identity Center](#)
- [Verwenden von vom Kunden verwalteten Richtlinien in IAM Identity Center für fortgeschrittene Anwendungsfälle](#)
- [AWS CLI v2: Anbieter von IAM Identity Center-Anmeldeinformationen](#)

Zugehörige Videos:

- [AWS re:Inforce 2022 – AWS Identity and Access Management \(IAM\) Vertiefung](#)
- [AWS re:Invent 2022 – Vereinfachen des vorhandenen Mitarbeiterzugriffs mit IAM Identity Center](#)
- [AWS re:Invent 2018: Beherrschen der Identität auf jeder Ebene](#)

Zugehörige Beispiele:

- [Workshop: Verwenden von AWS IAM Identity Center für eine robuste Identitätsverwaltung](#)

Zugehörige Tools:

- [AWS-Kompetenzpartner für Sicherheit: Identitäts- und Zugriffsverwaltung](#)
- [saml2aws](#)

SEC02-BP05 Regelmäßiges Überprüfen und Rotieren von Anmeldeinformationen

Prüfen und rotieren Sie Anmeldeinformationen regelmäßig, um die Zeit zu begrenzen, für die diese zum Zugriff auf Ihre Ressourcen genutzt werden können. Langfristig gültige Anmeldeinformationen sind mit Risiken verbunden, die durch die regelmäßige Rotation dieser Informationen reduziert werden können.

Gewünschtes Ergebnis: Implementieren Sie die Rotation von Anmeldeinformationen, um die Risiken zu verringern, die mit der Nutzung von langfristigen Anmeldeinformationen verbunden sind. Prüfen und korrigieren Sie regelmäßig fehlende Compliance mit Richtlinien zur Rotation von Anmeldeinformationen.

Typische Anti-Muster:

- Keine Prüfung der Verwendung von Anmeldeinformationen
- Unnötiges Verwenden langfristiger Anmeldeinformationen
- Verwendung langfristiger Anmeldeinformationen, ohne diese regelmäßig zu rotieren

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Wenn Sie sich nicht auf temporäre Anmeldeinformationen verlassen können und langfristige Anmeldeinformationen benötigen, prüfen Sie die Anmeldeinformationen, um sicherzustellen, dass definierte Kontrollen wie [Multi-Faktor-Authentifizierung](#) (MFA)) erzwungen und regelmäßig rotiert werden sowie über die entsprechende Zugriffsebene verfügen.

Eine regelmäßige Validierung, vorzugsweise durch ein automatisiertes Tool, ist notwendig, um zu überprüfen, ob die richtigen Kontrollen angewendet werden. Für Personenidentitäten

sollten Sie festlegen, dass Benutzer ihre Passwörter regelmäßig ändern und anstelle von Zugriffsschlüsseln temporäre Anmeldeinformationen verwenden. Wenn Sie von AWS Identity and Access Management (IAM) Benutzern zu zentralen Identitäten wechseln, können Sie [einen Bericht zu Anmeldeinformationen erstellen](#), um Ihre Benutzer zu überprüfen.

Wir empfehlen außerdem, dass Sie MFA in Ihrem Identitätsanbieter erzwingen. Sie können [AWS-Config-Regeln](#) einrichten oder [AWS Security Hub CSPM Security Standards](#) verwenden, um zu überwachen, ob Benutzer MFA konfiguriert haben. Erwägen Sie die Nutzung von [IAM Roles Anywhere](#) zur Bereitstellung temporärer Anmeldeinformationen für Maschinenidentitäten. In Situationen, in denen die Verwendung von IAM-Rollen und temporären Anmeldeinformationen nicht möglich ist, ist eine häufige Prüfung und Rotation von Zugriffsschlüsseln erforderlich.

Implementierungsschritte

- **Regelmäßige Prüfung der Anmeldeinformationen:** Die Prüfung der in Ihrem Identitätsanbieter und in IAM konfigurierten Identitäten hilft bei der Sicherstellung, dass nur autorisierte Identitäten Zugriff auf Ihre Workload haben. Solche Identitäten können unter anderem IAM-Benutzer, Benutzer von AWS IAM Identity Center, Active-Directory-Benutzer oder Benutzer in einem anderen vorgelagerten Identitätsanbieter sein. Entfernen Sie beispielsweise Personen, die die Organisation verlassen. Entfernen Sie auch kontoübergreifende Rollen, die nicht mehr erforderlich sind. Sie benötigen einen Prozess zum regelmäßigen Prüfen von Berechtigungen für die Services, auf die eine IAM-Entität zugreift. Dadurch können Sie die Richtlinien identifizieren, die Sie ändern müssen, um nicht genutzte Berechtigungen zu entfernen. Verwenden Sie Berichte zu Anmeldeinformationen und [AWS Identity and Access Management Access Analyzer](#), um IAM-Anmeldeinformationen und -Berechtigungen zu überprüfen. Sie können [Amazon CloudWatch verwenden, um Alarme für bestimmte API-Aufrufe einzurichten](#), die in Ihrer AWS-Umgebung erfolgen. [Amazon GuardDuty kann Sie auch vor unerwarteten Aktivitäten warnen](#), die auf einen übermäßig freizügigen Zugriff oder einen unbeabsichtigten Zugriff auf IAM-Anmeldeinformationen hindeuten können.
- **Anmeldeinformationen regelmäßig rotieren:** Wenn Sie keine temporären Anmeldeinformationen verwenden können, rotieren Sie langfristige IAM-Zugriffsschlüssel regelmäßig (spätestens nach jeweils 90 Tagen). Wenn ein Zugriffsschlüssel ohne Ihr Wissen kompromittiert wurde, wird dadurch begrenzt, für wie lange die Anmeldeinformationen zum Zugriff auf Ihre Ressourcen genutzt werden können. Informationen zum Austauschen von Zugriffsschlüsseln für IAM-Benutzer finden Sie unter [Rotieren der Zugriffsschlüssel](#).
- **IAM-Berechtigungen überprüfen:** Um die Sicherheit Ihres AWS-Kontos zu erhöhen, sollten Sie Ihre IAM-Richtlinien regelmäßig überprüfen und überwachen. Stellen Sie sicher, dass die Richtlinien dem Prinzip der geringsten Berechtigung entsprechen.

- Automatisierung der Erstellung und Aktualisierung von IAM-Ressourcen erwägen: [IAM Identity Center](#) automatisiert viele IAM-Aufgaben, etwa die Rollen- und Richtlinienverwaltung. Alternativ können Sie mit AWS CloudFormation die Bereitstellung von IAM-Ressourcen – einschließlich Rollen und Richtlinien – automatisieren. So lässt sich die Zahl menschlicher Fehler verringern, da die Vorlagen verifiziert und ihre Versionen kontrolliert werden können.
- IAM Roles Anywhere verwenden, um IAM-Benutzer für Maschinenidentitäten zu ersetzen: Mit [IAM Roles Anywhere](#) können Sie Rollen in Bereichen verwenden, in denen dies bisher nicht möglich war, z. B. auf On-Premises-Servern. IAM Roles Anywhere verwendet ein vertrauenswürdiges [X.509-Zertifikat](#) zur Authentifizierung gegenüber AWS und zum Erhalt temporärer Anmeldeinformationen. Mit IAM Roles Anywhere müssen Sie diese Anmeldeinformationen nicht mehr rotieren, da sie nicht mehr in Ihrer On-Premises-Umgebung gespeichert werden. Beachten Sie, dass Sie das X.509-Zertifikat beobachten und gegen Ende seiner Gültigkeitsdauer austauschen müssen.

Ressourcen

Zugehörige bewährte Methoden:

- [SEC02-BP02 Verwenden von temporären Anmeldeinformationen](#)
- [SEC02-BP03 Sicheres Speichern und Verwenden von Secrets](#)

Zugehörige Dokumente:

- [Erste Schritte mit AWS Secrets Manager](#)
- [IAM Best Practices](#)
- [Identitätsanbieter und Verbund](#)
- [Partnerlösungen im Bereich Sicherheit: Zugriff und Zugriffssteuerung](#)
- [Temporäre Sicherheitsanmeldeinformationen](#)
- [Abrufen von Berichten zu Anmeldeinformationen für Ihr AWS-Konto](#)

Zugehörige Videos:

- [Bewährte Methoden zum Verwalten, Abrufen und Rotieren von Secrets in großem Maßstab](#)
- [Verwaltung von Benutzerberechtigungen in großem Umfang mit AWS IAM Identity Center](#)
- [Beherrschen der Identität auf jeder Ebene](#)

SEC02-BP06 Nutzen von Benutzergruppen und Attributen

Die Definition von Berechtigungen nach Benutzergruppen und Attributen trägt dazu bei, die Anzahl und Komplexität von Richtlinien zu reduzieren, so dass das Prinzip der geringsten Berechtigung einfacher umgesetzt werden kann. Sie können Benutzergruppen verwenden, um die Berechtigungen für viele Personen an einem Ort zu verwalten, basierend auf der Funktion, die sie in Ihrer Organisation innehaben. Attribute, wie z. B. Abteilung, Projekt oder Standort, können eine zusätzliche Ebene des Berechtigungsumfangs bieten, wenn Personen eine ähnliche Funktion ausüben, jedoch für unterschiedliche Teilmengen von Ressourcen.

Gewünschtes Ergebnis: Sie können Änderungen der Berechtigungen auf alle Benutzer anwenden, die eine bestimmte Funktion ausführen. Die Gruppenzugehörigkeit und -attribute regeln die Benutzerberechtigungen, sodass Sie die Berechtigungen nicht mehr auf der Ebene der einzelnen Benutzer verwalten müssen. Die Gruppen und Attribute, die Sie in Ihrem Identitätsanbieter (IDP) definieren, werden automatisch an Ihre AWS-Umgebungen weitergegeben.

Typische Anti-Muster:

- Verwaltung von Berechtigungen für einzelne Benutzer und Duplizierung für viele Benutzer.
- Definition von Gruppen auf einer zu hohen Ebene, Gewährung von zu weitreichenden Berechtigungen.
- Die Definition von Gruppen auf einer zu granularen Ebene, was zu Doppelarbeit und Verwirrung über die Mitgliedschaft führt.
- Verwendung von Gruppen mit doppelten Berechtigungen für Teilmengen von Ressourcen, wenn stattdessen Attribute verwendet werden können.
- Keine Verwaltung von Gruppen, Attributen und Mitgliedschaften über einen standardisierten Identitätsanbieter, der in Ihre AWS-Umgebungen integriert ist.
- Verwenden von Rollenverkettung bei der Verwendung von Sitzungen von AWS IAM Identity Center

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

AWS-Berechtigungen werden in Dokumenten definiert, die als Richtlinien bezeichnet werden und einem Prinzipal zugeordnet sind, z. B. einem Benutzer, einer Gruppe, einer Rolle oder einer Ressource. Sie können das Berechtigungsmanagement skalieren, indem Sie die Zuweisungen von Berechtigungen (Gruppe, Berechtigungen, Konto) auf der Grundlage von Aufgabenfunktion, Workload und SDLC-Umgebung organisieren. So können Sie für Ihre Mitarbeiter Gruppen

definieren, die auf der Funktion basieren, die Ihre Benutzer in Ihrer Organisation innehaben, und nicht auf den Ressourcen, auf die sie zugreifen. Beispielsweise kann einer WebAppDeveloper-Gruppe eine Richtlinie für die Konfiguration von Services wie Amazon CloudFront innerhalb eines Entwicklungskontos angehängt sein. Eine AutomationDeveloper-Gruppe hat möglicherweise einige Berechtigungen, die sich mit der WebAppDeveloper-Gruppe überschneiden. Diese gemeinsamen Berechtigungen können in einer separaten Richtlinie erfasst und beiden Gruppen zugeordnet werden. Dadurch ist es nicht erforderlich, dass Benutzer beider Funktionen zu einer CloudFrontAccess-Gruppe gehören.

Zusätzlich zu Gruppen können Sie Attribute verwenden, um den Zugriff festzulegen. Sie können beispielsweise ein Projekt-Attribut für Benutzer in Ihrer WebAppDeveloper-Gruppe nutzen, damit die Benutzer nur auf Ressourcen ihres Projekts zugreifen können. Mit dieser Technik entfällt die Notwendigkeit, für Anwendungsentwickler, die an verschiedenen Projekten arbeiten, unterschiedliche Gruppen einzurichten, wenn ihre Berechtigungen ansonsten identisch sind. Die Art und Weise, wie Sie sich auf Attribute in Berechtigungsrichtlinien beziehen, hängt von deren Quelle ab, d. h. ob sie als Teil Ihres Verbundprotokolls (wie SAML, OIDC oder SCIM), als benutzerdefinierte SAML-Assertions oder innerhalb von IAM Identity Center definiert sind.

Implementierungsschritte

1. Legen Sie fest, wo Sie Gruppen und Attribute definieren wollen:

- a. Anhand der Anleitung unter [SEC02-BP04 Verlassen auf einen zentralen Identitätsanbieter](#) können Sie festlegen, ob Sie Gruppen und Attribute innerhalb Ihres Identitätsanbieters, innerhalb von IAM Identity Center oder mithilfe von IAM-Benutzergruppen in einem bestimmten Konto definieren müssen.

2. Definieren von Gruppen:

- a. Legen Sie Ihre Gruppen je nach Funktion und Umfang des erforderlichen Zugriffs fest. Erwägen Sie die Verwendung einer hierarchischen Struktur oder von Benennungskonventionen, um Gruppen effektiv zu organisieren.
- b. Wenn Sie innerhalb von IAM Identity Center definieren, erstellen Sie Gruppen und ordnen Sie die gewünschte Zugriffsebene mithilfe von Berechtigungsgruppen zu.
- c. Wenn Sie die Definition innerhalb eines externen Identitätsanbieters vornehmen, stellen Sie fest, ob der Anbieter das SCIM-Protokoll unterstützt und erwägen Sie die Aktivierung der automatischen Bereitstellung innerhalb von IAM Identity Center. Diese Funktion synchronisiert die Erstellung, Mitgliedschaft und Löschung von Gruppen zwischen Ihrem Anbieter und IAM Identity Center.

3. Definieren von Attributen:

- a. Wenn Sie einen externen Identitätsanbieter verwenden, bieten sowohl das SCIM- als auch das SAML 2.0-Protokoll standardmäßig bestimmte Attribute. Zusätzliche Attribute können mithilfe von SAML-Zusicherungen unter Verwendung des `https://aws.amazon.com/SAML/Attributes/PrincipalTag`-Attributnamens definiert und übergeben werden. Schritte zum Definieren und Konfigurieren von benutzerdefinierten Attributen finden Sie in der Dokumentation Ihres Identitätsanbieters.
- b. Wenn Sie Rollen innerhalb von IAM Identity Center definieren, aktivieren Sie das Feature attributbasierte Zugriffskontrolle (ABAC) und definieren Sie die Attribute nach Bedarf. Ziehen Sie Attribute in Betracht, die zur Struktur oder zur Ressourcen-Tagging-Strategie Ihres Unternehmens passen.

Wenn Sie eine IAM-Rollenverkettung von IAM-Rollen benötigen, die über das IAM Identity Center übernommen wurden, werden Werte wie `source-identity` und `principal-tags` nicht weitergegeben. Weitere Informationen finden Sie unter [Aktivieren und Konfigurieren von Attributen für die Zugriffskontrolle](#).

1. Legen Sie den Umfang von Berechtigungen basierend auf Gruppen und Attributen fest:
 - a. Erwägen Sie, Bedingungen in Ihre Genehmigungsrichtlinien aufzunehmen, die die Attribute Ihres Prinzipals mit den Attributen der Ressourcen vergleichen, auf die zugegriffen wird. Sie können beispielsweise eine Bedingung so definieren, dass der Zugriff auf eine Ressource nur dann gewährt wird, wenn der Wert eines `PrincipalTag`-Bedingungsschlüssels mit dem Wert eines `ResourceTag`-Schlüssels mit demselben Namen übereinstimmt.
 - b. Beachten Sie bei der Definition von ABAC-Richtlinien die Hinweise in den bewährten Methoden und Beispielen für die [ABAC-Autorisierung](#).
 - c. Überprüfen und aktualisieren Sie Ihre Gruppen- und Attributstruktur regelmäßig, wenn sich die Anforderungen Ihres Unternehmens weiterentwickeln, um ein optimales Berechtigungsmanagement sicherzustellen.

Ressourcen

Zugehörige bewährte Methoden:

- [SEC02-BP04 Verlassen auf einen zentralen Identitätsanbieter](#)
- [SEC03-BP02 Gewähren des Zugriffs mit den geringsten Berechtigungen](#)
- [COST02-BP04 Implementieren von Gruppen und Rollen](#)

Zugehörige Dokumente:

- [IAM Best Practices](#)
- [Verwaltung von Identitäten in IAM Identity Center](#)
- [Wofür wird ABAC in AWS verwendet?](#)
- [ABAC in IAM Identity Center](#)
- [Beispiele für ABAC-Richtlinien](#)

Zugehörige Videos:

- [Verwaltung von Benutzerberechtigungen in großem Umfang mit AWS IAM Identity Center](#)
- [Beherrschen der Identität auf jeder Ebene](#)

SEC 3. Wie werden Berechtigungen für Personen und Computer verwaltet?

Verwalten Sie Berechtigungen zum Steuern des Zugriffs für menschliche Identitäten und Maschinenidentitäten, die Zugriff auf AWS und Ihre Workloads benötigen. Berechtigungen steuern, wer unter welchen Bedingungen worauf zugreifen kann. Legen Sie Berechtigungen für bestimmte menschliche oder Maschinenidentitäten fest, um Zugriff auf bestimmte Service-Aktionen für bestimmte Ressourcen zu gewähren. Sie können auch Bedingungen angeben, die erfüllt sein müssen, damit der Zugriff gewährt wird.

Best Practices

- [SEC03-BP01 Definieren von Zugriffsanforderungen](#)
- [SEC03-BP02 Gewähren des Zugriffs mit den geringsten Berechtigungen](#)
- [SEC03-BP03 Einrichtung eines Notfallzugriffprozesses](#)
- [SEC03-BP04 Kontinuierliche Reduzierung der Berechtigungen](#)
- [SEC03-BP05 Definieren eines Integritätsschutzes für Berechtigungen in Ihrer Organisation](#)
- [SEC03-BP06 Zugriffsverwaltung basierend auf dem Lebenszyklus](#)
- [SEC03-BP07 Analysieren des öffentlichen und kontoübergreifenden Zugriffs](#)
- [SEC03-BP08 Sicheres gemeinsames Nutzen von Ressourcen in Ihrer Organisation](#)
- [SEC03-BP09 Sicheres Teilen von Ressourcen mit Dritten](#)

SEC03-BP01 Definieren von Zugriffsanforderungen

Auf jede Komponente oder Ressource Ihrer Workload müssen Administratoren, Endbenutzer oder andere Komponenten zugreifen können. Definieren Sie klar, wer oder was Zugriff auf die einzelnen Komponenten haben soll, und wählen Sie den geeigneten Identitätstyp und die Methode für die Authentifizierung und Autorisierung aus.

Typische Anti-Muster:

- Hartkodierung oder Speicherung von geheimen Daten in Ihrer Anwendung
- Gewähren individueller Berechtigungen für jeden Benutzer
- Verwendung langlebiger Anmeldeinformationen

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Auf jede Komponente oder Ressource Ihrer Workload müssen Administratoren, Endbenutzer oder andere Komponenten zugreifen können. Definieren Sie klar, wer oder was Zugriff auf die einzelnen Komponenten haben soll, und wählen Sie den geeigneten Identitätstyp und die Methode für die Authentifizierung und Autorisierung aus.

Der reguläre Zugriff auf AWS-Konten innerhalb einer Organisation sollte über den [Verbundzugriff](#) oder einen zentralen Identitätsanbieter bereitgestellt werden. Sie sollten auch Ihr Identitätsmanagement zentralisieren und sicherstellen, dass es ein etabliertes Verfahren zur Integration des AWS-Zugriffs in den Zugriffslebenszyklus der Mitarbeiter gibt. Wenn beispielsweise ein Mitarbeiter in eine Rolle mit einer anderen Zugriffsstufe wechselt, sollte sich auch dessen Gruppenmitgliedschaft so ändern, dass die neuen Zugriffsanforderungen berücksichtigt werden.

Legen Sie bei der Definition der Zugriffsanforderungen für nicht menschliche Identitäten fest, welche Anwendungen und Komponenten Zugriff benötigen und wie die Berechtigungen gewährt werden. Eine empfohlene Vorgehensweise ist die Verwendung von nach dem Modell der geringsten Berechtigung entwickelten IAM-Rollen. [AWS Verwaltete Richtlinien](#) bieten vordefinierte IAM-Richtlinien für die meisten typischen Anwendungsfälle.

AWS-Services wie [AWS Secrets Manager](#) und [AWS Systems Manager Parameter Store](#) können Ihnen dabei helfen, Secrets sicher von der Anwendung oder dem Workload zu entkoppeln. In Secrets Manager können Sie die automatische Rotation Ihrer Anmeldeinformationen einrichten. Mit Systems Manager können Sie auf Parameter in Ihren Skripten, Befehlen, SSM-Dokumenten, Konfigurations-

und Automatisierungsworkflows verweisen, indem Sie den bei der Erstellung des Parameters angegebenen eindeutigen Namen verwenden.

Sie können [AWS IAM Roles Anywhere](#) verwenden, um [temporäre Sicherheitsanmeldeinformationen in IAM](#) für Workloads abzurufen, die außerhalb von AWS ausgeführt werden. Ihre Workloads können dieselben [IAM-Richtlinien](#) und [IAM-Rollen](#) verwenden, die Sie auch bei AWS-Anwendungen für den Zugriff auf AWS-Ressourcen verwenden.

Verwenden Sie nach Möglichkeit kurzfristige temporäre anstelle langfristiger statischer Anmeldeinformationen. Für Szenarien, in denen Sie -Benutzer mit programmgeteuertem Zugriff und langfristigen Anmeldeinformationen benötigen, verwenden Sie die [Informationen über die letzte Nutzung von Zugriffsschlüsseln](#), um die Zugriffsschlüssel zu rotieren und zu entfernen.

Benutzer benötigen programmgesteuerten Zugriff, wenn sie außerhalb der AWS-Managementkonsole mit AWS interagieren möchten. Die Vorgehensweise, um programmgesteuerten Zugriff zu gewähren, hängt davon ab, welcher Benutzertyp auf zugreift AWS.

Um Benutzern programmgesteuerten Zugriff zu gewähren, wählen Sie eine der folgenden Optionen.

Welcher Benutzer benötigt programmgesteuerten Zugriff?	Bis	Von
IAM	(Empfohlen) Verwenden Sie Konsolen-Anmeldeinformationen als temporäre Anmeldeinformationen, um programmgesteuerte Anforderungen an die AWS CLI, AWS-SDKs oder AWS-APIs zu signieren.	Befolgen Sie die Anweisungen für die Schnittstelle, die Sie verwenden möchten. - Informationen zur AWS CLI finden Sie unter Anmeldung für lokale AWS-Entwicklung im AWS Command Line Interface-Benutzerhandbuch. - Informationen zu AWS-SDKs finden Sie unter Anmeldung für lokale AWS-Entwicklung im Referenzhandbuch zu AWS-SDKs und -Tools.

Welcher Benutzer benötigt programmgesteuerten Zugriff?	Bis	Von
Mitarbeiteridentität (Benutzer, die in IAM Identity Center verwaltet werden)	Verwenden Sie temporäre Anmeldeinformationen, um programmgesteuerte Anforderungen an die AWS CLI, AWS-SDKs oder AWS-APIs zu signieren.	Befolgen Sie die Anweisungen für die Schnittstelle, die Sie verwenden möchten. • Informationen zur AWS CLI finden Sie unter Konfigurieren der AWS CLI für die Verwendung von AWS IAM Identity Center im AWS Command Line Interface-Benutzerhandbuch. • Informationen zu AWS-SDKs, Tools und AWS-APIs finden Sie unter IAM-Identity-Center-Authentifizierung im Referenzhandbuch zu AWS-SDKs und Tools.
IAM	Verwenden Sie temporäre Anmeldeinformationen, um programmgesteuerte Anforderungen an die AWS CLI, AWS-SDKs oder AWS-APIs zu signieren.	Folgen Sie den Anweisungen unter Verwenden temporärer Anmeldeinformationen mit AWS-Ressourcen im IAM-Benutzerhandbuch.

Welcher Benutzer benötigt programmgesteuerten Zugriff?	Bis	Von
IAM	(Nicht empfohlen) Verwenden Sie langfristige Anmeldeinformationen, um programmgesteuerte Anforderungen an die AWS CLI, AWS-SDKs oder AWS-APIs zu signieren.	Befolgen Sie die Anweisungen für die Schnittstelle, die Sie verwenden möchten. • Informationen zur AWS CLI finden Sie unter Authentifizierung mit IAM-Benutzer-Anmeldeinformationen im AWS Command Line Interface-Benutzerhandbuch. • Informationen zu AWS-SDKs und Tools finden Sie unter Authentifizierung mit langfristigen Anmeldeinformationen im Referenzhandbuch zu AWS-SDKs und Tools. • Informationen zu AWS-APIs finden Sie unter Verwalten von Zugriffsschlüsseln für IAM-Benutzer im IAM-Benutzerhandbuch.

Ressourcen

Zugehörige Dokumente:

- [Attributbasierte Zugriffskontrolle \(ABAC\)](#)
- [AWS IAM Identity Center](#)
- [IAM Roles Anywhere](#)
- [AWS Verwaltete -Richtlinien für IAM Identity Center](#)
- [AWS IAM-Richtlinienbedingungen](#)

- [IAM-Anwendungsfälle](#)
- [Entfernen unnötiger Anmeldeinformationen](#)
- [Arbeiten mit -Richtlinien](#)
- [Steuerung des Zugriffs auf AWS-Ressourcen auf der Grundlage von AWS-Konto, OU oder Organisation](#)
- [Identifizieren, Arrangieren und Verwalten von geheimen Daten mithilfe der erweiterten Suche in AWS Secrets Manager](#)

Zugehörige Videos:

- [Experte für IAM-Richtlinien in unter 60 Minuten](#)
- [Trennung von Pflichten, geringste Berechtigung, Delegation und CI/CD](#)
- [Optimieren des Identitäts- und Zugriffsmanagements für Innovation](#)

SEC03-BP02 Gewähren des Zugriffs mit den geringsten Berechtigungen

Gewähren Sie nur den Zugriff, den Benutzer benötigen, um bestimmte Aktionen auf bestimmten Ressourcen unter bestimmten Bedingungen durchzuführen. Nutzen Sie Gruppen und Identitätsattribute, um Berechtigungen dynamisch in großem Umfang festzulegen, anstatt Berechtigungen für einzelne Benutzer zu definieren. Sie können beispielsweise einer Gruppe von Entwicklern den Zugriff erlauben, nur die Ressourcen für ihr Projekt zu verwalten. So ist sichergestellt, dass einem Entwickler, der nicht mehr am Projekt arbeitet, automatisch der Zugriff entzogen wird, ohne dass die zugrunde liegenden Zugriffsrichtlinien geändert werden müssen.

Gewünschtes Ergebnis: Benutzer verfügen nur über die Berechtigungen, die für ihre Arbeit erforderlich sind. Sie verwenden separate AWS-Konten, um Entwickler von Produktionsumgebungen zu isolieren. Wenn Entwickler für bestimmte Aufgaben auf Produktionsumgebungen zugreifen müssen, wird ihnen nur für die Dauer dieser Aufgaben eingeschränkter und kontrollierter Zugriff gewährt. Ihr Zugriff auf die Produktion wird sofort aufgehoben, nachdem sie die erforderlichen Arbeiten abgeschlossen haben. Sie führen regelmäßige Überprüfungen der Berechtigungen durch und widerrufen sie umgehend, wenn sie nicht mehr benötigt werden, z. B. wenn ein Benutzer die Rolle wechselt oder das Unternehmen verlässt. Sie beschränken Administratorrechte auf eine kleine, vertrauenswürdige Gruppe, um das Risiko zu verringern. Sie gewähren Maschinen- oder Systemkonten nur die Mindestberechtigungen, die zur Ausführung der vorgesehenen Aufgaben erforderlich sind.

Typische Anti-Muster:

- Sie gewähren Benutzern standardmäßig Administratorberechtigungen.
- Sie verwenden das Root-Benutzerkonto für tägliche Aktivitäten.
- Sie erstellen übermäßig permissive Richtlinien ohne angemessene Beschränkung des Geltungsbereichs.
- Ihre Berechtigungen werden nur selten überprüft, was dazu führt, dass sie ständig erweitert werden.
- Sie verlassen sich ausschließlich auf die attributbasierte Zugriffskontrolle, wenn es um die Isolierung von Umgebungen oder die Verwaltung von Berechtigungen geht.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Das Prinzip der [geringsten Berechtigung](#) besagt, dass Identitäten nur die kleinstmögliche Menge von Aktionen ausführen dürfen, die zur Durchführung einer bestimmten Aufgabe erforderlich sind. Dies schafft ein Gleichgewicht zwischen Benutzerfreundlichkeit, Effizienz und Sicherheit. Die Anwendung dieses Prinzips trägt dazu bei, den unbeabsichtigten Zugriff zu beschränken und nachzuverfolgen, wer auf welche Ressourcen zugreifen kann. IAM-Benutzer und -Rollen verfügen standardmäßig über keine Berechtigungen. Der Root-Benutzer hat standardmäßig vollen Zugriff und sollte streng kontrolliert, überwacht und nur für [Aufgaben verwendet werden, die Root-Zugriff erfordern](#).

Mithilfe von IAM-Richtlinien können ausdrücklich Berechtigungen für IAM-Rollen oder bestimmte Ressourcen erteilt werden. So können beispielsweise identitätsbasierte Richtlinien an IAM-Gruppen angefügt werden, während S3-Buckets von ressourcenbasierten Richtlinien kontrolliert werden können.

Wenn Sie eine IAM-Richtlinie erstellen, können Sie die Serviceaktionen, Ressourcen und Bedingungen angeben, die erfüllt sein müssen, damit AWS den Zugriff erlaubt oder verweigert. AWS unterstützt eine Vielzahl von Bedingungen, mit denen Sie den Zugriff einschränken können. Mithilfe des [Bedingungsschlüssels](#) `PrincipalOrgID` können Sie beispielsweise Aktionen ablehnen, wenn der Anforderer nicht zu Ihrer AWS-Organisation gehört.

Sie können auch Anforderungen kontrollieren, die AWS-Services in Ihrem Namen stellen, etwa das Erstellen einer AWS Lambda-Funktion durch AWS CloudFormation. Dazu verwenden Sie den Bedingungsschlüssel `CalledVia`. Sie können unterschiedliche Richtlinientypen in Ebenen organisieren, um ein umfassendes Verteidigungskonzept aufzubauen und die Berechtigungen

Ihrer Benutzer insgesamt zu begrenzen. Sie können auch Beschränkungen in Bezug darauf festlegen, welche Berechtigungen unter welchen Umständen erteilt werden können. Sie können Ihren Workload-Teams beispielsweise gestatten, ihre eigenen IAM-Richtlinien für die von ihnen erstellten Systeme zu erstellen; es muss aber auch eine [Berechtigungsgrenze](#) festgelegt werden, um die maximalen Berechtigungen zu beschränken, die sie gewähren können.

Implementierungsschritte

- Richtlinien für die geringste Berechtigung implementieren: Weisen Sie IAM-Gruppen und -Rollen Zugriffsrichtlinien mit geringsten Berechtigungen zu, die an den von Ihnen definierten Tätigkeitsbereich der Benutzer angepasst sind.
- Entwicklungs- und Produktionsumgebungen durch separate AWS-Konten trennen: Verwenden Sie separate AWS-Konten für Entwicklungs- und Produktionsumgebungen und kontrollieren Sie den Zugriff zwischen diesen Umgebungen mithilfe von [Servicekontrollrichtlinien](#), Ressourcen- und Identitätsrichtlinien.
- Grundlegende Richtlinien für die API-Nutzung: Eine Möglichkeit, die erforderlichen Berechtigungen zu ermitteln, ist die Überprüfung von AWS CloudTrail-Protokollen. Diese Prüfung ermöglicht es Ihnen, Berechtigungen zu erstellen, die auf die Aktionen zugeschnitten sind, die der Benutzer tatsächlich in AWS ausführt. [IAM Access Analyzer](#) kann [automatisch](#) IAM-Richtlinien auf der Grundlage von Zugriffsaktivitäten generieren. Sie können IAM Access Advisor auf Organisations- oder Kontoebene verwenden, um [die zuletzt abgerufenen Informationen für eine bestimmte Richtlinie nachzuverfolgen](#).
- Die Verwendung von [AWS-verwalteten Richtlinien für Tätigkeitsbereiche erwägen](#): Wenn Sie mit der Erstellung detaillierter Berechtigungsrichtlinien beginnen, kann es nützlich sein, AWS-verwaltete Richtlinien für gängige Positionen wie Fakturierungsmitarbeiter, Datenbankadministratoren und Datenwissenschaftler zu verwenden. Diese Richtlinien können helfen, den Zugriff der Benutzer einzuschränken, während Sie festlegen, wie die Richtlinien für die geringste Berechtigung implementiert werden sollen.
- Unnötige Berechtigungen entfernen: Erkennen und entfernen Sie nicht genutzte IAM-Entitäten, Anmeldeinformationen und Berechtigungen, um das Prinzip der geringsten Berechtigung durchzusetzen. Sie können [IAM Access Analyzer](#) verwenden, um externen und nicht genutzten Zugriff zu identifizieren, und die [Richtliniengenerierung von IAM Access Analyzer](#) kann zur Optimierung der Berechtigungsrichtlinien beitragen.
- Sicherstellen, dass Benutzer eingeschränkten Zugriff auf Produktionsumgebungen haben: Benutzer sollten nur Zugriff auf Produktionsumgebungen haben, wenn es sich um einen gültigen Anwendungsfall handelt. Nachdem der Benutzer die konkreten Aufgaben ausgeführt hat, für die

Zugriff auf die Produktionsumgebung erforderlich war, sollte der Zugriff widerrufen werden. Die Beschränkung des Zugriffs auf Produktionsumgebungen hilft, unbeabsichtigte Vorkommnisse mit Auswirkungen auf die Produktion zu verhindern und das Ausmaß der Auswirkungen eines unbeabsichtigten Zugriffs zu verringern.

- **Berechtigungsgrenzen erwägen::** Eine [Berechtigungsgrenze](#) ist ein Feature für eine verwaltete Richtlinie, das die maximalen Berechtigungen festlegt, die mit einer identitätsbasierten Richtlinie einer IAM-Entität erteilt werden können. Durch eine Berechtigungsgrenze kann eine Entität nur die Aktionen durchführen, die sowohl von den identitätsbasierten Richtlinien als auch den Berechtigungsgrenzen erlaubt werden.
- **Den Zugriff mithilfe von attributbasierter Zugriffskontrolle und Ressourcentags verfeinern** Die [attributbasierte Zugriffskontrolle \(ABAC\)](#) mithilfe von Ressourcentags kann, sofern sie unterstützt wird, zur Verfeinerung von Berechtigungen verwendet werden. Sie können ein ABAC-Modell verwenden, das Prinzipal-Tags mit Ressourcen-Tags vergleicht, um den Zugriff auf der Grundlage der von Ihnen definierten benutzerdefinierten Dimensionen zu verfeinern. Dieses Konzept kann die Berechtigungsrichtlinien in Ihrer Organisation vereinfachen und ihre Anzahl reduzieren.
- **Es wird empfohlen, ABAC nur für die Zugriffskontrolle zu verwenden, wenn sowohl die Prinzipale als auch die Ressourcen zu Ihrer AWS-Organisation gehören.** Externe Parteien können dieselben Tag-Namen und Werte wie Ihre Organisation für ihre eigenen Prinzipale und Ressourcen verwenden. Wenn Sie sich bei der Gewährung des Zugriffs für Prinzipale oder Ressourcen von externen Parteien ausschließlich auf diese Name-Wert-Paare stützen, kann es vorkommen, dass Sie nicht beabsichtigte Berechtigungen erteilen.
- **Service-Kontrollrichtlinien für AWS Organizations verwenden:** Service-Kontrollrichtlinien steuern zentral die maximal verfügbaren Berechtigungen für Mitgliedskonten in Ihrer Organisation. Wichtig ist, dass Sie mithilfe von Service-Kontrollrichtlinien die Root-Benutzerberechtigungen in Mitgliedskonten einschränken können. Ziehen Sie auch die Verwendung von AWS Control Tower in Betracht, das präskriptive verwaltete Kontrollen zur Bereicherung von AWS Organizations bietet. Sie können auch Ihre eigenen Kontrollen in Control Tower definieren.
- **Eine Lebenszyklusrichtlinie für Benutzer für Ihre Organisation einrichten:** Benutzer-Lebenszyklusrichtlinien definieren Aufgaben, die ausgeführt werden, wenn Benutzer in AWS hinzugefügt werden, ihre Rolle oder ihren Aufgabenbereich ändern oder sie keinen Zugriff auf AWS mehr benötigen. Führen Sie bei jedem Schritt im Lebenszyklus eines Benutzers Berechtigungsprüfungen durch, um sicherzustellen, dass die Berechtigungen angemessen restriktiv sind und keine schleichenden Berechtigungserweiterungen stattfinden.
- **Einen regelmäßigen Zeitplan einrichten, um die Berechtigungen zu überprüfen und alle nicht benötigten Berechtigungen zu entfernen:** Sie sollten den Benutzerzugriff regelmäßig überprüfen,

um sicherzustellen, dass Benutzer keinen übermäßigen Zugriff haben. [AWS Config](#) und IAM Access Analyzer können Sie bei der Prüfung der Benutzerberechtigungen unterstützen.

- **Job-Rollen-Matrix erstellen:** Eine Job-Rollen-Matrix stellt die verschiedenen Rollen und Zugriffsebenen, die in Ihrem AWS-System erforderlich sind, grafisch dar. Mithilfe einer Job-Rollen-Matrix können Sie Berechtigungen auf der Grundlage von Benutzerzuständigkeiten in Ihrer Organisation definieren und trennen. Verwenden Sie Gruppen, anstatt Berechtigungen direkt auf einzelne Benutzer oder Rollen anzuwenden.

Ressourcen

Zugehörige Dokumente:

- [Gewähren der geringsten Berechtigung](#)
- [Berechtigungsgrenzen für IAM-Entitäten](#)
- [Techniken zum Erstellen von IAM-Richtlinien mit geringsten Berechtigungen](#)
- [IAM Access Analyzer erleichtert die Implementierung geringster Berechtigungen durch die Generierung von IAM-Richtlinien auf der Grundlage der Zugriffsaktivitäten](#)
- [Delegieren der Berechtigungsverwaltung an Entwickler mithilfe von IAM-Berechtigungsgrenzen](#)
- [Verfeinern von Berechtigungen mithilfe der Informationen zum letzten Zugriff](#)
- [IAM-Richtlinienarten und wann sie verwendet werden sollten](#)
- [Testen von IAM-Richtlinien mit dem IAM-Richtliniensimulator](#)
- [Integritätsschutz in AWS Control Tower](#)
- [Zero-Trust-Architekturen: Eine AWS-Perspektive](#)
- [Implementieren des Prinzips der geringsten Berechtigung mit CloudFormation StackSets](#)
- [Attributbasierte Zugriffskontrolle \(ABAC\)](#)
- [Reduzieren des Richtlinienbereichs durch Anzeigen der Benutzeraktivität](#)
- [Anzeigen des Rollenzugriffs](#)
- [Verwenden Sie Tagging, um Ihre Umgebung zu organisieren und die Verantwortlichkeit zu fördern](#)
- [AWS-Strategien für das Tagging](#)
- [Taggen von AWS-Ressourcen](#)

Zugehörige Videos:

- [Berechtigungsmanagement der nächsten Generation](#)
- [Zero Trust: Eine AWS-Perspektive](#)

SEC03-BP03 Einrichtung eines Notfallzugriffsprozesses

Erstellen Sie einen Prozess, der im unwahrscheinlichen Fall eines Problems mit Ihrem zentralen Identitätsanbieter den Notfallzugriff auf Ihre Workloads ermöglicht.

Sie müssen Prozesse für verschiedene Ausfallmodi entwerfen, die zu einem Notfallereignis führen können. Unter normalen Umständen verbinden sich die Benutzer Ihrer Belegschaft beispielsweise über einen zentralen Identitätsanbieter ([SEC02-BP04](#)) mit der Cloud, um ihre Workloads zu verwalten. Wenn der zentrale Identitätsanbieter jedoch ausfällt oder die Konfiguration für den Verbund in der Cloud geändert wird, können sich die Benutzer in Ihrem Unternehmen möglicherweise nicht mit der Cloud verbinden. Ein Prozess für den Notfallzugriff ermöglicht autorisierten Administratoren den Zugriff auf Ihre Cloud-Ressourcen über alternative Verfahren (z. B. eine alternative Form des Verbunds oder direkter Benutzerzugriff), um Probleme mit Ihrer Verbundkonfiguration oder Ihren Workloads zu beheben. Der Prozess für den Notfallzugriff wird verwendet, bis der normale Verbundmechanismus wiederhergestellt ist.

Gewünschtes Ergebnis:

- Sie haben die Ausfallmodi definiert und dokumentiert, die als Notfall gelten: Berücksichtigen Sie dabei Ihre normalen Abläufe und die Systeme, auf die Ihre Benutzer angewiesen sind, um ihre Workloads zu verwalten. Überlegen Sie, wie jede dieser Abhängigkeiten ausfallen und zu einer Notsituation führen kann. Möglicherweise finden Sie die Fragen und bewährten Methoden in der [Säule der Zuverlässigkeit](#) hilfreich, um Ausfallarten zu identifizieren und widerstandsfähigere Systeme zu entwickeln, bei denen die Wahrscheinlichkeit von Ausfällen geringer ist.
- Sie haben die Schritte dokumentiert, die befolgt werden müssen, um einen Ausfall als Notfall zu identifizieren. Sie können beispielsweise festlegen, dass Ihre Identitätsadministratoren den Status Ihrer primären und Standby-Identitätsanbieter überprüfen müssen und, falls beide nicht verfügbar sind, ein Notfallereignis für den Ausfall eines Identitätsanbieters feststellen.
- Sie haben einen Prozess für den Notfallzugriff definiert, der für jeden Notfall- oder Ausfallmodus spezifisch ist. Wenn Sie hier möglichst detaillierte Informationen angeben, kann dies der Neigung Ihrer Benutzer entgegenwirken, einen allgemeinen Prozess für alle Arten von Notfällen zu stark zu nutzen. Ihre Prozesse für den Notfallzugriff beschreiben die Umstände, unter denen ein Prozess jeweils verwendet werden sollte, und umgekehrt Situationen, in denen der Prozess nicht verwendet werden sollte. In diesem Fall wird auf alternative Prozesse hingewiesen, die zutreffen können.

- Ihre Prozesse sind mit detaillierten Anweisungen und Playbooks, die schnell und effizient befolgt werden können, gut dokumentiert. Denken Sie daran, dass ein Notfallereignis Stress für Ihre Benutzer bedeuten kann und dass sie unter extremem Zeitdruck stehen können. Gestalten Sie Ihren Prozess daher so einfach wie möglich.

Typische Anti-Muster:

- Sie verfügen nicht über gut dokumentierte und gut getestete Prozesse für den Notfallzugriff. Ihre Benutzer sind nicht auf einen Notfall vorbereitet und nutzen improvisierte Prozesse, wenn er eintritt.
- Ihre Prozesse für den Notfallzugriff hängen von denselben Systemen (z. B. einem zentralen Identitätsanbieter) ab wie Ihre normalen Zugriffsmechanismen. Das bedeutet, dass der Ausfall eines solchen Systems sowohl Ihre normalen Zugriffsmechanismen als auch Ihre Notfallzugriffsmechanismen betrifft und Ihre Fähigkeit zur Wiederherstellung nach dem Ausfall beeinträchtigen kann.
- Ihre Prozesse für den Notfallzugriff werden in Situationen verwendet, die keine Notfälle sind. Ein Beispiel könnte sein, dass Ihre Benutzer Prozesse für den Notfallzugriff häufig missbrauchen, da es für sie einfacher ist, Änderungen direkt vorzunehmen, als Änderungen über eine Pipeline einzureichen.
- Ihre Prozesse für den Notfallzugriff generieren nicht genügend Protokolle, um sie zu überwachen, oder die Protokolle werden nicht so überwacht, dass Sie bei einem möglichen Missbrauch der Prozesse gewarnt werden.

Vorteile der Nutzung dieser bewährten Methode:

- Durch gut dokumentierte und gut getestete Prozesse für den Notfallzugriff können Sie die Zeit reduzieren, die Ihre Benutzer benötigen, um auf ein Notfallereignis zu reagieren und es zu beheben. Dies kann zu kürzeren Ausfallzeiten und einer höheren Verfügbarkeit der Services führen, die Sie für Ihre Kunden bereitstellen.
- Sie können jede Notfallzugriffsanfrage verfolgen und unbefugte Versuche, den Prozess für Nicht-Notfallereignisse zu missbrauchen, erkennen und darauf hinweisen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Dieser Abschnitt enthält Richtlinien zur Erstellung von Prozessen für den Notfallzugriff für verschiedene Ausfallmodi im Zusammenhang mit Workloads, die in AWS bereitgestellt werden. Zunächst finden Sie allgemeine Leitlinien, die für alle Ausfallmodi gelten, und danach spezifische Anleitungen für die verschiedenen Arten von Ausfallmodi.

Allgemeine Leitlinien für alle Ausfallmodi

Beachten Sie beim Entwerfen eines Prozesses für den Notfallzugriff für einen Ausfallmodus Folgendes:

- Dokumentieren Sie die Voraussetzungen und Annahmen für den Prozess: Wann soll der Prozess verwendet werden und wann nicht? Es ist hilfreich, den Ausfallmodus detailliert zu beschreiben und Annahmen zu dokumentieren, z. B. den Zustand anderer verwandter Systeme. Der Prozess für den Ausfallmodus 2 geht beispielsweise davon aus, dass der Identitätsanbieter verfügbar ist, aber die Konfiguration in AWS geändert wurde oder abgelaufen ist.
- Erstellen Sie im Voraus Ressourcen, die für den Notfallzugriffsprozess benötigt werden ([SEC10-BP05](#)). Erstellen Sie beispielsweise vorab das AWS-Konto für den Notfallzugriff mit IAM-Benutzern und -Rollen sowie die kontoübergreifenden IAM-Rollen in allen Workload-Konten. So wird sichergestellt, dass diese Ressourcen bereit und verfügbar sind, wenn ein Notfallereignis eintritt. Durch die Vorab-Erstellung von Ressourcen sind Sie nicht von APIs der AWS-[Steuerebene](#) (zum Erstellen und Ändern von AWS-Ressourcen verwendet) abhängig, die im Notfall möglicherweise nicht verfügbar sind. Darüber hinaus müssen Sie durch die Vorab-Erstellung von IAM-Ressourcen keine [potenziellen Verzögerungen aufgrund der letztendlichen Datenkonsistenz](#) berücksichtigen.
- Schließen Sie Prozesse für den Notfallzugriff in Ihre Vorfalldmanagementpläne ein ([SEC10-BP02](#)). Dokumentieren Sie, wie Notfallereignisse nachverfolgt und an andere in Ihrem Unternehmen, z. B. an Peer-Teams, Führungskräfte und gegebenenfalls extern an Kunden und Geschäftspartner, kommuniziert werden sollen.
- Definieren Sie den Prozess für Notfallzugriffsanfragen in Ihrem bestehenden Workflow-System für Serviceanfragen, falls eines vorhanden ist. In der Regel können Sie mit solchen Workflow-Systemen Eingabeformulare erstellen, um Informationen zur Anfrage zu erfassen, die Anfrage in jeder Phase des Workflows zu verfolgen und sowohl automatisierte als auch manuelle Genehmigungsschritte hinzuzufügen. Ordnen Sie jede Anfrage einem entsprechenden Notfallereignis zu, das in Ihrem Vorfalldmanagement-System verfolgt wird. Mit einem einheitlichen System für Notfallzugriffe können Sie diese Anfragen in einem zentralen System verfolgen, Nutzungstrends analysieren und Ihre Prozesse verbessern.

- Stellen Sie sicher, dass Ihre Notfallzugriffsprozesse nur von autorisierten Benutzern initiiert werden können, und legen Sie fest, dass Genehmigungen von Kollegen oder Führungskräften des Benutzers erforderlich sind. Der Genehmigungsprozess sollte sowohl während als auch außerhalb der Geschäftszeiten funktionieren. Definieren Sie, wie Genehmigungsanfragen sekundäre Genehmiger berücksichtigen, falls die primären Genehmiger nicht verfügbar sind, und wie sie in Ihrer Managementkette nach oben eskaliert werden, bis sie genehmigt wurden.
- Implementieren Sie robuste Protokollierungs-, Überwachungs- und Warnmechanismen für den Notfallzugriffsprozess und die entsprechenden Mechanismen. Generieren Sie detaillierte Auditprotokolle für alle erfolgreichen und fehlgeschlagenen Versuche, Notfallzugriff zu erhalten. Korrelieren Sie die Aktivität mit laufenden Notfallereignissen aus Ihrem Vorfalldmanagement-System und senden Sie Benachrichtigungen, wenn Aktionen außerhalb der erwarteten Zeiträume erfolgen oder wenn das Notfallzugriffskonto während des normalen Betriebs verwendet wird. Auf das Notfallkonto sollte nur in Notfällen zugegriffen werden, da Break-Glass-Verfahren als Hintertür betrachtet werden sollten. Integrieren Sie dies in Ihr SIEM-Tool (Security Information and Event Management) oder [AWS Security Hub CSPM](#), um alle Aktivitäten während der Notfallzugriffsphase zu melden und zu überprüfen. Sobald Sie zum normalen Betrieb zurückkehren, rotieren Sie die Anmeldeinformationen für den Notfallzugriff automatisch und benachrichtigen Sie die zuständigen Teams.
- Testen Sie die Notfallzugriffsprozesse regelmäßig, um sicherzustellen, dass die Schritte klar sind und die richtigen Zugriffsebenen schnell und effizient gewährt werden. Ihre Notfallzugriffsprozesse sollten im Rahmen von Incident-Response-Simulationen ([SEC10-BP07](#)) und Notfallwiederherstellungs-Tests ([REL13-BP03](#)) getestet werden.

Ausfallmodus 1: Der für den Verbund mit AWS verwendete Identitätsanbieter ist nicht verfügbar

Wie in [SEC02-BP04 Verlassen auf einen zentralen Identitätsanbieter](#) beschrieben, wird empfohlen, sich auf einen zentralen Identitätsanbieter zu verlassen, der die Benutzer Ihres Unternehmens verbindet, um den Zugriff auf AWS-Konten zu gewähren. Sie können mit IAM Identity Center einen Verbund für mehrere AWS-Konten in Ihrer AWS-Organisation implementieren oder einzelne AWS-Konten mit IAM verbinden. In beiden Fällen authentifizieren sich die Benutzer in Ihrer Belegschaft beim zentralen Identitätsanbieter, bevor sie zu einem AWS-Anmeldeendpunkt für das Single Sign-On weitergeleitet werden.

Im unwahrscheinlichen Fall, dass der zentrale Identitätsanbieter nicht verfügbar ist, können sich die Benutzer Ihrer Belegschaft nicht mit AWS-Konten verbinden oder ihre Workloads verwalten. In einem solchen Notfall können Sie einen Notfallzugriffsprozess für eine kleine Gruppe von Administratoren einrichten, die auf AWS-Konten zugreifen dürfen, um kritische Aufgaben auszuführen, die nicht

warten können, bis die zentralen Identitätsanbieter wieder online sind. Nehmen Sie beispielsweise an, dass Ihr Identitätsanbieter für 4 Stunden nicht verfügbar ist und während dieses Zeitraums die Obergrenzen einer Amazon EC2 Auto Scaling-Gruppe in einem Produktionskonto geändert werden müssen, um einen unerwarteten Anstieg des Kundendatenverkehrs zu bewältigen. Ihre Notfalladministratoren sollten den Notfallzugriffsprozess befolgen, um Zugriff auf das spezifische AWS-Konto in der Produktion zu erhalten und die erforderlichen Änderungen vorzunehmen.

Der Notfallzugriffsprozess basiert auf einem vorab erstellten AWS-Konto für den Notfallzugriff, das ausschließlich für den Notfallzugriff verwendet wird und über AWS-Ressourcen (wie IAM-Rollen und IAM-Benutzer) zur Unterstützung des Notfallzugriffsprozesses verfügt. Während des normalen Betriebs sollte niemand auf das Notfallzugriffskonto zugreifen. Sie müssen dieses Konto auf Missbrauch überwachen und ggf. Warnungen senden (weitere Informationen finden Sie im vorherigen Abschnitt mit allgemeinen Leitlinien).

Das Notfallzugriffskonto verfügt über IAM-Notfallzugriffsrollen mit der Berechtigung, kontoübergreifende Rollen in den AWS-Konten anzunehmen, für die Notfallzugriff erforderlich ist. Diese IAM-Rollen sind vordefiniert und mit Vertrauensrichtlinien konfiguriert, die den IAM-Rollen des Notfallkontos vertrauen.

Der Notfallzugriffsprozess kann einen der folgenden Ansätze verwenden:

- Sie können im Notfallzugriffskonto vorab eine Gruppe von [IAM-Benutzern](#) mit zugehörigen sicheren Passwörtern und MFA-Token für Ihre Notfalladministratoren erstellen. Diese IAM-Benutzer verfügen über Berechtigungen, die IAM-Rollen anzunehmen, die dann den kontoübergreifenden Zugriff auf das AWS-Konto ermöglichen, für das der Notfallzugriff erforderlich ist. Wir empfehlen, so wenige solcher Benutzer wie möglich zu erstellen und jeden Benutzer einem einzelnen Notfalladministrator zuzuweisen. Während eines Notfalls meldet sich ein Notfalladministrator mit seinem Passwort und seinem MFA-Tokencode beim Notfallzugriffskonto an, wechselt zur IAM-Notfallzugriffsrolle im Notfallkonto und wechselt schließlich zur IAM-Notfallzugriffsrolle im Workload-Konto, um die für den Notfall erforderliche Änderungsaktion durchzuführen. Der Vorteil dieses Ansatzes besteht darin, dass jeder IAM-Benutzer einem Notfalladministrator zugewiesen ist und Sie anhand der CloudTrail-Ereignisse feststellen können, welcher Benutzer sich angemeldet hat. Der Nachteil ist, dass Sie mehrere IAM-Benutzer mit den zugehörigen langlebigen Passwörtern und MFA-Token verwalten müssen.
- Sie können den [Root-Benutzer des AWS-Kontos](#) für den Notfallzugriff verwenden, um sich beim Notfallzugriffskonto anzumelden, die IAM-Rolle für den Notfallzugriff anzunehmen und dann die kontoübergreifende Rolle im Workload-Konto anzunehmen. Wir empfehlen, ein sicheres Passwort und mehrere MFA-Token für den Root-Benutzer festzulegen. Wir empfehlen außerdem, das

Passwort und die MFA-Token in einem sicheren Vault für Unternehmensanmeldeinformationen zu speichern, der eine starke Authentifizierung und Autorisierung erzwingt. Sie sollten das Passwort und die Faktoren zum Zurücksetzen des MFA-Tokens sichern: Legen Sie die E-Mail-Adresse für das Konto auf eine E-Mail-Verteilerliste fest, die von Ihren Cloud-Sicherheitsadministratoren überwacht wird. Legen Sie die Telefonnummer des Kontos auf eine gemeinsam genutzte Telefonnummer fest, die ebenfalls von Sicherheitsadministratoren überwacht wird. Der Vorteil dieses Ansatzes besteht darin, dass nur ein Satz von Root-Benutzeranmeldeinformationen verwaltet werden muss. Der Nachteil ist, dass sich mehrere Administratoren als Root-Benutzer anmelden können, da es sich um einen gemeinsam genutzten Benutzer handelt. Sie müssen die Protokollereignisse für den Unternehmens-Vault überprüfen, um festzustellen, welcher Administrator das Passwort für den Root-Benutzer ausgecheckt hat.

Ausfallmodus 2: Die Konfiguration des Identitätsanbieters in AWS wurde geändert oder ist abgelaufen

Um den Verbund der Benutzer in Ihrem Unternehmen mit AWS-Konten zu ermöglichen, können Sie IAM Identity Center mit einem externen Identitätsanbieter konfigurieren oder einen IAM-Identitätsanbieter erstellen ([SEC02-BP04](#)). In der Regel konfigurieren Sie diese, indem Sie ein XML-Dokument mit SAML-Metadaten importieren, das von Ihrem Identitätsanbieter bereitgestellt wird. Das XML-Metadatendokument enthält ein X.509-Zertifikat, das einem privaten Schlüssel entspricht, mit dem der Identitätsanbieter seine SAML-Zusicherungen signiert.

Diese Konfigurationen auf AWS-Seite können versehentlich von einem Administrator geändert oder gelöscht werden. In einem anderen Szenario läuft das in AWS importierte X.509-Zertifikat möglicherweise ab und eine neue XML-Metadatendatei mit einem neuen Zertifikat wurde noch nicht in AWS importiert. In beiden Szenarien kann der Verbund mit AWS für die Benutzer Ihrer Belegschaft unterbrochen werden, was zu einem Notfall führt.

In einem solchen Notfall können Sie Ihren Identitätsadministratoren Zugriff auf AWS gewähren, um die Verbundprobleme zu beheben. Ihr Identitätsadministrator verwendet beispielsweise den Notfallzugriffsprozess, um sich beim AWS-Konto für den Notfallzugriff anzumelden. Er wechselt zu einer Rolle im Identity Center-Administratorkonto und aktualisiert die Konfiguration des externen Identitätsanbieters, indem er das aktuelle XML-Dokument mit SAML-Metadaten von Ihrem Identitätsanbieter importiert, um den Verbund wieder zu aktivieren. Sobald der Verbund wiederhergestellt ist, verwenden die Benutzer in Ihrer Belegschaft weiter den normalen Betriebsprozess, um sich mit ihren Workload-Konten zu verbinden.

Sie können die oben für Ausfallmodus 1 beschriebenen Vorgehensweisen befolgen, um einen Notfallzugriffsprozess zu erstellen. Sie können Ihren Identitätsadministratoren Berechtigungen

nach dem Prinzip der geringsten Berechtigung gewähren, sodass sie nur auf das Identity Center-Administratorkonto zugreifen und nur in diesem Konto Aktionen für Identity Center ausführen können.

Ausfallmodus 3: Störung von Identity Center

Für den unwahrscheinlichen Fall einer Störung von IAM Identity Center oder einer AWS-Region empfehlen wir, eine Konfiguration einzurichten, mit der Sie temporären Zugriff auf die AWS-Managementkonsole gewähren können.

Der Notfallzugriffsprozess verwendet einen direkten Verbund von Ihrem Identitätsanbieter zu IAM in einem Notfallkonto. Einzelheiten zu den Prozess- und Entwurfsüberlegungen finden Sie im [Artikel zum Einrichten des Notfallzugriffs auf die AWS-Managementkonsole](#).

Implementierungsschritte

Allgemeine Schritte für alle Ausfallmodi

- Erstellen Sie ein AWS-Konto speziell für Notfallzugriffsprozesse. Erstellen Sie vorab die für das Konto benötigten IAM-Ressourcen wie IAM-Rollen oder IAM-Benutzer und optional IAM-Identitätsanbieter. Erstellen Sie außerdem vorab kontoübergreifende IAM-Rollen in den AWS-Konten für die Workload mit Vertrauensbeziehungen zu den entsprechenden IAM-Rollen im Notfallzugriffskonto. Sie können [CloudFormation-StackSets mit AWS Organizations](#) verwenden, um solche Ressourcen in den Mitgliedskonten Ihrer Organisation zu erstellen.
- Erstellen Sie AWS Organizations-[Service-Kontrollrichtlinien](#) (SCP), um das Löschen und Ändern der kontoübergreifenden IAM-Rollen in den AWS-Konten der Mitglieder zu verweigern.
- Aktivieren Sie CloudTrail für das AWS-Konto für den Notfallzugriff und senden Sie die Trail-Ereignisse an einen zentralen S3-Bucket im AWS-Konto für die Protokollerfassung. Wenn Sie AWS Control Tower verwenden, um Ihre AWS-Umgebung mit mehreren Konten einzurichten und zu verwalten, ist für jedes Konto, das Sie mit AWS Control Tower erstellen oder in AWS Control Tower registrieren, CloudTrail standardmäßig aktiviert und wird an einen S3-Bucket in einem dedizierten AWS-Konto für das Protokollarchiv gesendet.
- Überwachen Sie die Aktivitäten des Notfallzugriffskontos, indem Sie EventBridge-Regeln erstellen, die bei der Anmeldung in der Konsole und bei API-Aktivitäten durch die IAM-Notfallrollen greifen. Senden Sie Benachrichtigungen an Ihr Security Operations Center, wenn Aktivitäten außerhalb eines laufenden Notfallereignisses stattfinden, das in Ihrem Vorfalldmanagement-System nachverfolgt wurde.

Zusätzliche Schritte für Ausfallmodus 1 (Der für den Verbund mit AWS verwendete Identitätsanbieter ist nicht verfügbar) und Ausfallmodus 2 (Die Konfiguration des Identitätsanbieters in AWS wurde geändert oder ist abgelaufen)

- Erstellen Sie vorab Ressourcen, je nachdem, welchen Mechanismus Sie für den Notfallzugriff wählen:
 - IAM-Benutzer verwenden: Erstellen Sie vorab die IAM-Benutzer mit sicheren Passwörtern und den zugehörigen MFA-Geräten.
 - Root-Benutzer des Notfallkontos verwenden: Konfigurieren Sie den Root-Benutzer mit einem sicheren Passwort und speichern Sie das Passwort im Unternehmens-Vault für Anmeldeinformationen. Ordnen Sie dem Root-Benutzer mehrere physische MFA-Geräte zu und bewahren Sie die Geräte an Orten auf, zu denen die Mitglieder Ihres Notfalladministratorteam schnell Zugang haben.

Zusätzliche Schritte für den Ausfallmodus 3 (Störung von Identity Center)

- Erstellen Sie wie im [Artikel zum Einrichten des Notfallzugriffs auf die AWS-Managementkonsole](#) erläutert im AWS-Konto für den Notfallzugriff einen IAM-Identitätsanbieter, um den direkten SAML-Verbund von Ihrem Identitätsanbieter aus zu ermöglichen.
- Erstellen Sie Notfalleinsatzgruppen in Ihrem Identitätsanbieter ohne Mitglieder.
- Erstellen Sie IAM-Rollen, die den Notfalleinsatzgruppen im Notfallzugriffskonto entsprechen.

Ressourcen

Zugehörige bewährte Methoden für Well-Architected:

- [SEC02-BP04 Verlassen auf einen zentralen Identitätsanbieter](#)
- [SEC03-BP02 Gewähren des Zugriffs mit den geringsten Berechtigungen](#)
- [SEC10-BP02 Entwickeln von Vorfalmanagementplänen](#)
- [SEC10-BP07 Durchführen von Gamedays](#)

Zugehörige Dokumente:

- [Artikel zum Einrichten des Notfallzugriffs auf die AWS-Managementkonsole](#)
- [Aktivieren von Zugriff auf die für SAML-2.0-Verbundbenutzer AWS-Managementkonsole](#)

- [Break Glass“-Zugriff](#)

Zugehörige Videos:

- [AWS re:Invent 2022 – Vereinfachen des vorhandenen Mitarbeiterzugriffs mit IAM Identity Center](#)
- [AWS re:Inforce 2022 – AWS Identity and Access Management \(IAM\) Vertiefung](#)

Zugehörige Beispiele:

- [AWS Rolle „Break Glass](#)
- [AWS Customer Playbook Framework](#)
- [AWS Beispiele von Playbooks für die Vorfallsreaktion](#)

SEC03-BP04 Kontinuierliche Reduzierung der Berechtigungen

Wenn Ihre Teams bestimmen, welchen Zugriff sie benötigen, entfernen Sie unnötige Berechtigungen und erstellen Sie Überprüfungsprozesse, damit jederzeit dem Prinzip der geringsten Berechtigung entsprochen wird. Überwachen Sie Ihre Identitäten kontinuierlich und entfernen Sie ungenutzte Identitäten und Berechtigungen für den Zugriff von Menschen und Maschinen.

Gewünschtes Ergebnis: Die Genehmigungsrichtlinien sollten dem Prinzip der geringsten Berechtigung entsprechen. Wenn Zuständigkeiten und Rollen immer besser definiert werden, müssen Sie Ihre Berechtigungsrichtlinien prüfen, um unnötige Berechtigungen zu entfernen. Dieses Konzept verringert die Auswirkungen, wenn Anmeldeinformationen versehentlich offengelegt werden oder wenn anderweitig ohne Genehmigung darauf zugegriffen wird.

Typische Anti-Muster:

- Standardmäßige Gewährung von Administratorberechtigungen für Benutzer
- Erstellung übermäßig großzügiger Richtlinien, jedoch ohne vollständige Administratorberechtigungen
- Aufbewahrung von Berechtigungsrichtlinien, nachdem sie nicht mehr benötigt werden

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Wenn Teams und Projekte gerade erst mit der Arbeit beginnen, können lockere Richtlinien verwendet werden, um Innovationen und Agilität zu unterstützen. So könnten beispielsweise Entwickler in einer Entwicklungs- und Testumgebung Zugang zu einer breiten Palette von AWS-Services erhalten. Wir empfehlen, den Zugriff kontinuierlich zu prüfen und auf Services und Serviceaktionen einzuschränken, die für die anstehende Aufgabe wirklich benötigt werden. Wir empfehlen diese Evaluierung für menschliche und für maschinelle Identitäten. Maschinenidentitäten, manchmal auch als System- oder Servicekonten bezeichnet, sind Identitäten, die AWS den Zugriff auf Anwendungen oder Server ermöglichen. Dieser Zugriff ist besonders in einer Produktionsumgebung wichtig, in der übermäßig lockere Zugriffsregeln weitreichende Auswirkungen haben und möglicherweise Kundendaten offen legen könnten.

AWS bietet mehrere Verfahren zur Unterstützung der Identifizierung nicht verwendeter Benutzer, Rollen, Berechtigungen und Anmeldeinformationen. AWS kann auch bei der Analyse von Zugriffsaktivitäten von IAM-Benutzern und -Rollen helfen, darunter ebenfalls Analysen zu zugehörigen Zugriffsschlüsseln sowie zum Zugriff auf AWS-Ressourcen wie etwa Objekten in Amazon S3-Buckets. Die Generierung von Richtlinien mit AWS Identity and Access Management Access Analyzer kann Ihnen bei der Erstellung restriktiver Berechtigungsrichtlinien auf der Grundlage der Services und Aktionen helfen, mit denen ein Prinzipal tatsächlich interagiert. Die [attributbasierte Zugriffskontrolle \(ABAC\)](#) kann zur Vereinfachung der Berechtigungsverwaltung beitragen, da Sie Benutzern anhand ihrer Attribute Berechtigungen erteilen können, anstatt jedem Benutzer direkt Berechtigungsrichtlinien zuzuweisen.

Implementierungsschritte

- [AWS Identity and Access Management Access Analyzer](#) verwenden: IAM Access Analyzer hilft Ihnen, die Ressourcen in Ihrer Organisation und Ihren Konten zu identifizieren, die [mit einer externen Entität geteilt werden](#) (z. B. Amazon Simple Storage Service (Amazon S3)-Buckets oder IAM-Rollen).
- Das [Generieren von IAM Access Analyzer-Richtlinien](#) verwenden: Mit dem Generieren von IAM Access Analyzer-Richtlinien können Sie [detaillierte Berechtigungsrichtlinien erstellen, die auf der Zugriffsaktivität eines IAM-Benutzers oder einer IAM-Rolle basieren](#).
- Vor der Produktionsphase Berechtigungen in weniger kritischen Umgebungen testen: Verwenden Sie zunächst die [weniger kritischen Sandbox- und Entwicklungsumgebungen](#), um die für verschiedene Tätigkeitsbereiche erforderlichen Berechtigungen mit IAM Access Analyzer zu testen. Verschärfen und validieren Sie dann schrittweise diese Berechtigungen in allen Test-, Qualitätssicherungs- und Staging-Umgebungen, bevor Sie sie für die Produktion anwenden.

In den weniger kritischen Umgebungen können zunächst lockerere Berechtigungen gelten, da Service-Kontrollrichtlinien (SCPs) Integritätsschutz durchsetzen, indem sie die maximal erteilten Berechtigungen einschränken.

- Einen akzeptablen Zeitrahmen und eine akzeptable Nutzungsrichtlinie für IAM-Benutzer und -Rollen festlegen: Verwenden Sie den [Zeitstempel des letzten Zugriffs](#), um [ungenutzte Benutzer und Rollen zu identifizieren](#) und sie zu entfernen. Überprüfen Sie die Informationen zum letzten Service- und Aktionszugriff überprüfen, um [Berechtigungen für bestimmte Benutzer und Rollen zu identifizieren und festzulegen](#). Sie können beispielsweise Informationen zum letzten Zugriff verwenden, um die spezifischen Amazon S3-Aktionen zu identifizieren, die Ihre Anwendungsrolle erfordert, und den Zugriff der Rolle auf diese Aktionen beschränken. Features für die zuletzt abgerufenen Informationen sind in der AWS-Managementkonsole und programmgesteuert verfügbar, damit Sie sie in Ihre Infrastruktur-Workflows und automatisierten Tools integrieren können.
- [Protokollierung von Datenereignissen in AWS CloudTrail](#) erwägen: Standardmäßig protokolliert CloudTrail keine Datenereignisse wie Amazon S3-Aktivitäten auf Objektebene (zum Beispiel GetObject und DeleteObject) oder Amazon DynamoDB-Tabellenaktivitäten (zum Beispiel PutItem und DeleteItem). Erwägen Sie die Verwendung der Protokollierung dieser Ereignisse, um zu ermitteln, welche Benutzer und Rollen Zugriff auf bestimmte Amazon S3-Objekte oder DynamoDB-Tabellenelemente benötigen.

Ressourcen

Zugehörige Dokumente:

- [Gewähren der geringsten Berechtigung](#)
- [Entfernen unnötiger Anmeldeinformationen](#)
- [Was ist AWS CloudTrail?](#)
- [Arbeiten mit -Richtlinien](#)
- [Protokollierung und Überwachung in DynamoDB](#)
- [Verwenden der CloudTrail-Ereignisprotokollierung für Amazon S3-Buckets und -Objekte](#)
- [Abrufen von Berichten zu Anmeldeinformationen für Ihr AWS-Konto](#)

Zugehörige Videos:

- [Experte für IAM-Richtlinien in unter 60 Minuten](#)

- [Trennung von Pflichten, geringste Berechtigung, Delegation und CI/CD](#)
- [AWS re:Inforce 2022 – AWS Identity and Access Management \(IAM\) Vertiefung](#)

SEC03-BP05 Definieren eines Integritätsschutzes für Berechtigungen in Ihrer Organisation

Verwenden Sie Maßnahmen zum Integritätsschutz, um den Umfang der verfügbaren Berechtigungen, die Prinzipalen gewährt werden können, einzuschränken. Die Bewertungskette für Genehmigungsrichtlinien umfasst Ihren Integritätsschutz, um bei Autorisierungsentscheidungen die effektiven Berechtigungen eines Prinzipals zu bestimmen. Sie können Maßnahmen zum Integritätsschutz mit einem ebenenbasierten Ansatz definieren. Wenden Sie einige Maßnahmen zum Integritätsschutz allgemein für Ihre gesamte Organisation an und andere granular auf Sitzungen mit temporärem Zugriff.

Gewünschtes Ergebnis: Die Umgebungen sind durch die Verwendung separater AWS-Konten klar voneinander abgegrenzt. Service-Kontrollrichtlinien (SCP) werden verwendet, um organisationsweite Maßnahmen zum Integritätsschutz zu definieren. Umfassender angelegte Maßnahmen zu Integritätsschutz werden auf den Hierarchieebenen festgelegt, die der Root Ihrer Organisation am nächsten sind, und strengerer Integritätsschutz wird näher an der Ebene der einzelnen Konten festgelegt.

Sofern unterstützt, definieren Ressourcenrichtlinien die Bedingungen, die ein Prinzipal erfüllen muss, um Zugriff auf eine Ressource zu erhalten. Die Ressourcenrichtlinien schränken auch den Umfang der erlaubten Aktionen ein, wo dies angebracht ist. Berechtigungsgrenzen werden auf Prinzipale verteilt, die Workload-Berechtigungen verwalten und die Verwaltung von Berechtigungen an einzelne Workload-Besitzer delegieren.

Typische Anti-Muster:

- AWS-Konten für Mitglieder werden innerhalb einer [AWS-Organisation](#) erstellt, ohne dass SCPs verwendet werden, um die Nutzung und die für ihre Root-Anmeldeinformationen verfügbaren Rechte einzuschränken.
- Zuweisung von Berechtigungen auf der Grundlage der geringsten Berechtigung, aber kein Integritätsschutz für die maximale Anzahl von Berechtigungen, die gewährt werden können
- Sie verlassen sich bei der Einschränkung von Berechtigungen auf die implizite Ablehnungsgrundlage von AWS IAM und vertrauen darauf, dass Richtlinien keine unerwünschte ausdrückliche Genehmigungsberechtigung gewähren.
- Mehrere Workload-Umgebungen im selben AWS-Konto ausführen und sich dann auf Mechanismen wie VPCs, Tags oder Ressourcenrichtlinien verlassen, um Berechtigungsgrenzen durchzusetzen

Vorteile der Nutzung dieser bewährten Methode: Durch den Integritätsschutz für Berechtigungen kann das Vertrauen gestärkt werden, dass keine unerwünschten Berechtigungen erteilt werden können, selbst wenn eine Berechtigungsrichtlinie dies versucht. Dies kann die Definition und Verwaltung von Berechtigungen vereinfachen, da der maximale Umfang der zu berücksichtigenden Berechtigungen reduziert wird.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Wir empfehlen Ihnen, einen ebenenbasierten Ansatz zu verwenden, um für Maßnahmen für den Integritätsschutz für Ihre Organisation zu definieren. Dieser Ansatz reduziert systematisch die maximale Anzahl der möglichen Berechtigungen, wenn weitere Ebenen hinzugefügt werden. So können Sie den Zugriff nach dem Prinzip der geringsten Berechtigung gewähren und das Risiko eines unbeabsichtigten Zugriffs aufgrund einer falschen Konfiguration der Richtlinie verringern.

Der erste Schritt zur Einrichtung zum Integritätsschutz ist die Isolierung Ihrer Workloads und Umgebungen in getrennten AWS-Konten. Prinzipale eines Kontos können ohne ausdrückliche Genehmigung nicht auf Ressourcen in einem anderen Konto zugreifen, selbst wenn sich beide Konten in derselben AWS-Organisation oder derselben [Organisationseinheit \(OE\)](#) befinden. Sie können OEs verwenden, um Konten zu gruppieren, die Sie als eine Einheit verwalten möchten.

Der nächste Schritt besteht darin, die maximale Anzahl von Berechtigungen zu reduzieren, die Sie Prinzipalen innerhalb der Mitgliedskonten Ihrer Organisation erteilen können. Zu diesem Zweck können Sie [Service-Kontrollrichtlinien \(SCP\)](#) verwenden, die Sie entweder auf eine Organisationseinheit oder ein Konto anwenden können. SCPs können allgemeine Zugriffskontrollen durchsetzen, etwa die Beschränkung des Zugriffs auf bestimmte AWS-Regionen, die Verhinderung des Löschens von Ressourcen oder die Deaktivierung potenziell riskanter Serviceaktionen. SCPs, die Sie auf das Root-Verzeichnis Ihrer Organisation anwenden, wirken sich nur auf die Mitgliedskonten aus, nicht auf das Verwaltungskonto. SCPs regeln nur die Prinzipale innerhalb Ihrer Organisation. Ihre SCPs regeln keine Prinzipale außerhalb Ihrer Organisation, die auf Ihre Ressourcen zugreifen.

Wenn Sie [AWS Control Tower](#) verwenden, können Sie die [Steuerungen](#) und [Landing Zones](#) als Grundlage für Ihren Integritätsschutz für Berechtigungen und Ihre Multi-Konten-Umgebung nutzen. Die Landing Zones bieten eine vorkonfigurierte, sichere Basisumgebung mit getrennten Konten für verschiedene Workloads und Anwendungen. Der Integritätsschutz setzt verbindliche Kontrollen in Bezug auf Sicherheit, Betrieb und Compliance durch eine Kombination aus Service-Kontrollrichtlinien (SCPs), AWS Config-Regeln und anderen Konfigurationen durch. Bei der Verwendung von Integritätsschutz und Landing Zones im Control Tower zusammen mit SCPs, die

für die Kundenorganisation spezifisch sind, ist es jedoch kritisch, die in der AWS-Dokumentation beschriebenen Best Practices zu befolgen, um Konflikte zu vermeiden und eine angemessene Steuerung sicherzustellen. Detaillierte Empfehlungen zur Verwaltung von SCPs, Konten und Organisationseinheiten (OUs) in einer Control-Tower-Umgebung finden Sie in der [AWS Control Tower-Anleitung für AWS Organizations](#).

Wenn Sie sich an diese Empfehlungen halten, können Sie den Integritätsschutz, die Landing Zones und die benutzerdefinierten SCPs von Control Tower effektiv nutzen. Gleichzeitig vermeiden Sie potenzielle Konflikte und stellen eine angemessene Verwaltung und Kontrolle Ihrer AWS-Umgebung mit mehreren Konten sicher.

Ein weiterer Schritt besteht darin, mithilfe von [IAM-Ressourcenrichtlinien](#) die verfügbaren Aktionen festzulegen, die Sie für die zugehörigen Ressourcen ausführen können – zusammen mit allen Bedingungen, die der aktuelle Prinzipal erfüllen muss. Dies kann so breit gefasst sein, dass alle Aktionen zugelassen werden, solange der Prinzipal Teil Ihrer Organisation ist (unter Verwendung des [Bedingungsschlüssels](#) PrincipalOrgID), oder so detailliert sein, dass nur bestimmte Aktionen einer bestimmten IAM-Rolle zugelassen werden. Sie können einen ähnlichen Ansatz mit Bedingungen in IAM-Rollenvertrauensrichtlinien verfolgen. Wenn eine Vertrauensrichtlinie für eine Ressource oder Rolle explizit einen Prinzipal im selben Konto wie die Rolle oder Ressource benennt, die sie regelt, benötigt dieser Prinzipal keine angehängte IAM-Richtlinie, die dieselben Berechtigungen gewährt. Wenn der Prinzipal ein anderes Konto hat als die Ressource, dann benötigt der Prinzipal eine angehängte IAM-Richtlinie, die diese Berechtigungen gewährt.

Oft möchte ein Workload-Team die für seine Workload erforderlichen Berechtigungen verwalten. Dazu muss es möglicherweise neue IAM-Rollen und Berechtigungsrichtlinien erstellen. Sie können den maximalen Umfang der Berechtigungen erfassen, die das Team innerhalb einer [IAM-Berechtigungsgrenze](#) gewähren darf, und dieses Dokument einer IAM-Rolle zuordnen, mit der das Team dann seine IAM-Rollen und -Berechtigungen verwalten kann. Dieser Ansatz kann dem Team die nötige Flexibilität bieten, die Aufgaben zu erledigen, und gleichzeitig die Risiken reduzieren, die durch einen IAM-Verwaltungszugriff entstehen.

Ein detaillierterer Schritt ist die Implementierung von Techniken zur Verwaltung des privilegierten Zugriffs (PAM) und zur Verwaltung des vorübergehend erhöhten Zugriffs (TEAM). Ein Beispiel für PAM ist die Anforderung an Prinzipale, sich mehrfach zu authentifizieren, bevor sie privilegierte Aktionen durchführen. Weitere Informationen finden Sie unter [Konfigurieren eines MFA-geschützten API-Zugriffs](#). TEAM benötigt eine Lösung, die die Genehmigung und den Zeitrahmen verwaltet, in dem ein Prinzipal erweiterten Zugriff haben darf. Eine Möglichkeit besteht darin, den Prinzipal vorübergehend in die Vertrauensrichtlinie für eine IAM-Rolle aufzunehmen, die über einen erweiterten

Zugriff verfügt. Ein anderer Ansatz besteht darin, im Normalbetrieb die einem Prinzipal durch eine IAM-Rolle gewährten Berechtigungen mithilfe einer [Sitzungsrichtlinie](#) einzuschränken und diese Einschränkung dann während des genehmigten Zeitfensters vorübergehend aufzuheben. Weitere Informationen zu Lösungen, die AWS und ausgewählte Partner validiert haben, finden Sie unter [Temporärer erweiterter Zugriff](#).

Implementierungsschritte

1. Isolieren Sie Ihre Workloads und Umgebungen in separaten AWS-Konten.
2. Verwenden Sie SCPs, um die maximale Anzahl von Berechtigungen zu reduzieren, die Prinzipalen innerhalb der Mitgliedskonten Ihrer Organisation gewährt werden können.
 - a. Bei der Definition von SCPs zur Reduzierung der maximalen Anzahl von Berechtigungen, die Prinzipalen innerhalb der Mitgliedskonten Ihrer Organisation gewährt werden können, können Sie zwischen einer Zulassungsliste und einer Verweigerungsliste wählen. Die Zulassungslistenstrategie gibt explizit die zulässigen Zugriffe an und blockiert implizit alle anderen Zugriffe. Die Verweigerungslistenstrategie gibt explizit die unzulässigen Zugriffe an und lässt standardmäßig alle anderen Zugriffe zu. Beide Strategien haben Vor- und Nachteile. Die Entscheidung ist von den spezifischen Anforderungen und vom Risikomodell Ihres Unternehmens abhängig. Weitere Informationen finden Sie unter [Strategie für die Verwendung von SCPs](#).
 - b. Sehen Sie sich auch die [Beispiele für Service-Kontrollrichtlinien](#) an, um zu verstehen, wie Sie SCPs effektiv konstruieren können.
3. Verwenden Sie IAM-Ressourcenrichtlinien, um den Geltungsbereich einzugrenzen und Bedingungen für zulässige Aktionen auf Ressourcen festzulegen. Verwenden Sie Bedingungen in IAM-Rollenvertrauensrichtlinien, um Einschränkungen für die Übernahme von Rollen zu erstellen.
4. Weisen Sie IAM-Berechtigungsgrenzen zu IAM-Rollen zu, die Workload-Teams dann zur Verwaltung ihrer eigenen Workload-IAM-Rollen und -Berechtigungen verwenden können.
5. Evaluieren Sie PAM- und TEAM-Lösungen auf der Grundlage Ihrer Bedürfnisse.

Ressourcen

Zugehörige Dokumente:

- [Datenperimeter in AWS](#)
- [Einrichten des Berechtigungs-Integritätsschutzes mithilfe von Datenperimetern](#)
- [Auswertungslogik für Richtlinien](#)

Zugehörige Beispiele:

- [Beispiele für Service-Kontrollrichtlinie](#)

Zugehörige Tools:

- [AWS Lösung: Temporäre erweiterte Zugriffsverwaltung](#)
- [Validierte Sicherheitspartnerlösungen für TEAM](#)

SEC03-BP06 Zugriffsverwaltung basierend auf dem Lebenszyklus

Überwachen Sie die Berechtigungen, die Ihren Prinzipalen (Benutzern, Rollen und Gruppen) während ihres gesamten Lebenszyklus in Ihrer Organisation gewährt werden, und passen Sie sie an. Passen Sie die Gruppenmitgliedschaften an, wenn Benutzer ihre Rolle ändern, und entfernen Sie den Zugriff, wenn ein Benutzer die Organisation verlässt.

Gewünschtes Ergebnis: Sie überwachen und passen die Berechtigungen während des gesamten Lebenszyklus der Prinzipale innerhalb der Organisation an und reduzieren so das Risiko unnötiger Rechte. Sie gewähren den entsprechenden Zugriff, wenn Sie einen Benutzer anlegen. Sie ändern den Zugriff, wenn sich die Aufgaben des Benutzers ändern, und Sie entfernen den Zugriff, wenn der Benutzer nicht mehr aktiv ist oder die Organisation verlassen hat. Sie verwalten Änderungen an Ihren Benutzern, Rollen und Gruppen zentral. Sie verwenden die Automatisierung, um Änderungen in Ihren AWS-Umgebungen zu verbreiten.

Typische Anti-Muster:

- Sie erteilen Identitäten im Voraus übermäßige oder weitreichende Zugriffsrechte, die über das zunächst erforderliche Maß hinausgehen.
- Sie überprüfen und ändern die Zugriffsberechtigungen nicht, wenn sich Rollen und Verantwortlichkeiten der Identitäten im Laufe der Zeit ändern.
- Sie entfernen aktive Zugriffsberechtigungen nicht von inaktiven oder beendeten Identitäten. Dies erhöht das Risiko eines unbefugten Zugriffs.
- Sie nutzen keine Automatisierung, um den Lebenszyklus von Identitäten zu verwalten.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Verwalten Sie die Zugriffsprivilegien, die Sie Identitäten (z. B. Benutzern, Rollen, Gruppen) gewähren, sorgfältig und passen Sie sie im Laufe ihres Lebenszyklus an. Dieser Lebenszyklus umfasst die anfängliche Onboarding-Phase, laufende Änderungen der Rollen und Verantwortlichkeiten und schließlich das Offboarding oder die Kündigung. Verwalten Sie den Zugriff proaktiv je nach Stadium des Lebenszyklus, um die richtige Zugriffsstufe zu erhalten. Halten Sie sich an das Prinzip der geringsten Berechtigung, um Risiken durch übermäßige oder unnötige Zugriffsberechtigungen zu verringern.

Sie können den Lebenszyklus von IAM-Benutzern direkt innerhalb des AWS-Konto oder durch den Verbund Ihres Identitätsanbieters für Mitarbeiter mit [AWS IAM Identity Center](#) verwalten. Für IAM-Benutzer können Sie innerhalb des AWS-Konto Benutzer und die damit verbundenen Berechtigungen erstellen, ändern und löschen. Für Verbundbenutzer können Sie IAM Identity Center verwenden, um ihren Lebenszyklus zu verwalten. Hierzu synchronisieren Sie Benutzer- und Gruppeninformationen aus dem Identitätsanbieter Ihrer Organisation über das Protokoll [System for Cross-domain Identity Management](#) (SCIM).

SCIM ist ein offenes Standardprotokoll für die automatisierte Bereitstellung und Deprovisionierung von Benutzeridentitäten über verschiedene Systeme hinweg. Durch die Integration Ihres Identitätsanbieters mit IAM Identity Center unter Verwendung von SCIM können Sie Benutzer- und Gruppeninformationen automatisch synchronisieren und so sicherstellen, dass Zugriffsberechtigungen auf der Grundlage von Änderungen in der maßgeblichen Identitätsquelle Ihrer Organisation gewährt, geändert oder entzogen werden.

Wenn sich die Rollen und Zuständigkeiten der Mitarbeiter in Ihrer Organisation ändern, passen Sie ihre Zugriffsrechte entsprechend an. Sie können die Berechtigungssätze von IAM Identity Center verwenden, um verschiedene Job-Rollen oder -Verantwortlichkeiten zu definieren und sie mit den entsprechenden IAM-Richtlinien und -Berechtigungen zu verknüpfen. Wenn sich die Rolle eines Mitarbeiters ändert, können Sie die ihm zugewiesenen Berechtigungen aktualisieren, um die neuen Verantwortlichkeiten zu berücksichtigen. Vergewissern Sie sich, dass sie über den erforderlichen Zugriff verfügen, und halten Sie sich dabei an das Prinzip der geringsten Berechtigung.

Implementierungsschritte

1. Definieren und dokumentieren Sie einen Lebenszyklusprozess für die Zugriffsverwaltung, einschließlich Verfahren für die Gewährung des Erstzugriffs, regelmäßige Überprüfungen und das Offboarding.

2. Implementieren Sie [IAM-Rollen, -Gruppen und -Berechtigungsgrenzen](#), um den Zugriff kollektiv zu verwalten und die maximal zulässigen Zugriffsstufen durchzusetzen.
3. Führen Sie eine Integration mit einem [Anbieter von Verbundidentitäten](#) (z. B. Microsoft Active Directory, Okta, Ping Identity) als autoritativer Quelle für Benutzer- und Gruppeninformationen mit IAM Identity Center durch.
4. Verwenden Sie das [SCIM](#)-Protokoll, um Benutzer- und Gruppeninformationen aus dem Identitätsanbieter mit dem Identitätsspeicher von IAM Identity Center zu synchronisieren.
5. Erstellen Sie in IAM Identity Center [Berechtigungssätze](#), die verschiedene Jobrollen oder Verantwortlichkeiten in Ihrer Organisation repräsentieren. Definieren Sie die entsprechenden IAM-Richtlinien und -Berechtigungen für jeden Berechtigungssatz.
6. Führen Sie regelmäßige Zugriffsüberprüfungen, sofortigen Zugriffsentzug und eine kontinuierliche Verbesserung des Lebenszyklusprozesses der Zugriffsverwaltung ein.
7. Schulung und Sensibilisierung der Mitarbeiter für die bewährten Methoden der Zugriffsverwaltung.

Ressourcen

Zugehörige bewährte Methoden:

- [SEC02-BP04 Verlassen auf einen zentralen Identitätsanbieter](#)

Zugehörige Dokumente:

- [Verwaltung Ihrer Identitätsquelle](#)
- [Verwaltung von Identitäten in IAM Identity Center](#)
- [Verwenden von AWS Identity and Access Management Access Analyzer](#)
- [Generieren von IAM Access Analyzer-Richtlinien](#)

Zugehörige Videos:

- [AWS re:Inforce 2023 – Temporäre erweiterte Zugriffsverwaltung mit AWS IAM Identity Center](#)
- [AWS re:Invent 2022 – Vereinfachen des vorhandenen Mitarbeiterzugriffs mit IAM Identity Center](#)
- [AWS re:Invent 2022 – Nutzung der Leistungsfähigkeit von IAM-Richtlinien und Einschränken der Berechtigungen mit Access Analyzer](#)

SEC03-BP07 Analysieren des öffentlichen und kontoübergreifenden Zugriffs

Überwachen Sie kontinuierlich Ergebnisse, die den öffentlichen und kontoübergreifenden Zugriff betreffen. Beschränken Sie den öffentlichen und kontoübergreifenden Zugriff ausschließlich auf Ressourcen, die diese Art von Zugriff benötigen.

Gewünschtes Ergebnis: Sie wissen, welche Ihrer AWS-Ressourcen für welche Benutzer freigegeben sind. Überwachen und prüfen Sie kontinuierlich Ihre freigegebenen Ressourcen, um sicherzustellen, dass sie nur für autorisierte Prinzipale freigegeben sind.

Typische Anti-Muster:

- Fehlendes Inventar gemeinsam genutzter Ressourcen
- Nichtbefolgung eines Prozesses zur Genehmigung von kontoübergreifendem oder öffentlichem Zugriff auf Ressourcen

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Niedrig

Implementierungsleitfaden

Wenn sich Ihr Konto in AWS Organizations befindet, können Sie den Zugriff auf Ressourcen der gesamten Organisation, bestimmten Organisationseinheiten oder einzelnen Konten gewähren. Wenn Ihr Konto nicht zu einer Organisation gehört, können Sie Ressourcen für einzelne Konten freigeben. Sie können direkten kontoübergreifenden Zugriff gewähren, indem Sie ressourcenbasierte Richtlinien verwenden (z. B. die [Richtlinien für Amazon Simple Storage Service \(Amazon S3\)-Buckets](#)) oder indem Sie einem Prinzipal in einem anderen Konto erlauben, eine IAM-Rolle in Ihrem Konto zu übernehmen. Verifizieren Sie bei der Verwendung von Ressourcenrichtlinien, dass der Zugriff nur autorisierten Prinzipalen gewährt wird. Definieren Sie einen Prozess für die Genehmigung aller Ressourcen, die öffentlich verfügbar sein müssen.

[AWS Identity and Access Management Access Analyzer](#) nutzt [nachweisbare Sicherheit](#), um alle Zugriffspfade zu einer Ressource von außerhalb ihres Kontos zu identifizieren. Es überprüft Ressourcenrichtlinien kontinuierlich und meldet Ergebnisse des öffentlichen und kontoübergreifenden Zugriffs, um Ihnen die Analyse potenziell umfassender Zugriffe zu erleichtern. Ziehen Sie die Konfiguration von IAM Access Analyzer mit AWS Organizations in Betracht, um Transparenz für alle Ihre Konten zu gewährleisten. Mit IAM Access Analyzer können Sie zudem [eine Vorschau der Ergebnisse anzeigen](#), bevor Sie Ressourcenberechtigungen bereitstellen. So können Sie sicherstellen, dass mit den Richtlinienänderungen nur der beabsichtigte öffentliche und kontoübergreifende Zugriff auf Ihre Ressourcen gewährt wird. Wenn Sie den Zugriff auf mehrere

Konten planen, können Sie mithilfe von [Vertrauensrichtlinien](#) steuern, in welchen Fällen eine Rolle übernommen werden kann. Sie könnten beispielsweise den [PrincipalOrgId-Bedingungsschlüssel verwenden, um Versuche, eine Rolle von außerhalb Ihres AWS Organizations zu übernehmen, abzulehnen](#).

[AWS Config kann falsch konfigurierte Ressourcen melden](#) und mithilfe von AWS Config-Richtlinienprüfungen Ressourcen erkennen, für die ein öffentlicher Zugriff konfiguriert ist. Services wie [AWS Control Tower](#) und [AWS Security Hub CSPM](#) vereinfachen die Bereitstellung von detektivischen Kontrollen und Integritätsschutz über AWS Organizations hinweg, um öffentlich zugängliche Ressourcen zu identifizieren und zu korrigieren. AWS Control Tower hat beispielsweise einen verwalteten Integritätsschutz, der erkennen kann, ob [Amazon-EBS-Snapshots von AWS-Konten wiederhergestellt werden können](#).

Implementierungsschritte

- Die Verwendung von [AWS Config für AWS Organizations](#) erwägen: Mit AWS Config können Sie die Ergebnisse mehrerer Konten in einem AWS Organizations in einem delegierten Administratorkonto zusammenfassen. Dies bietet einen umfassenden Überblick und ermöglicht die [kontoübergreifende Bereitstellung von AWS-Config-Regeln, um öffentlich zugängliche Ressourcen zu erkennen](#).
- AWS Identity and Access Management Access Analyzer konfigurieren: IAM Access Analyzer hilft Ihnen, die Ressourcen in Ihrer Organisation und in Ihren Konten zu identifizieren, [die für eine externe Entität freigegeben wurden](#), z. B. Amazon S3-Buckets oder IAM-Rollen.
- Automatische Abhilfemaßnahmen in AWS Config verwenden, um auf Änderungen an der Konfiguration des öffentlichen Zugriffs von Amazon S3-Buckets zu reagieren: [Sie können die Einstellungen zum Blockieren des öffentlichen Zugriffs für Amazon S3-Buckets automatisch aktivieren](#).
- Überwachung und Warnmeldungen implementieren, um festzustellen, ob Amazon S3-Buckets öffentlich geworden sind: [Überwachung und Warnmeldungen](#) müssen aktiviert sein, damit erkannt werden kann, wenn Amazon S3 Block Public Access deaktiviert wird und ob Amazon S3-Buckets öffentlich geworden sind. Wenn Sie AWS Organizations verwenden, können Sie außerdem eine [Service-Kontrollrichtlinie](#) erstellen, die Änderungen an den Amazon-S3-Richtlinien für den öffentlichen Zugriff verhindert. [AWS Trusted Advisor](#) sucht nach Amazon-S3-Buckets mit offenen Zugriffsberechtigungen. Bucket-Berechtigungen, die allen Benutzern den Zugriff zum Hochladen/Löschen einräumen, bergen ein hohes Potenzial für Sicherheitsrisiken, da alle Personen Elemente in einem Bucket hinzufügen, ändern oder löschen können. Bei der Prüfung durch Trust Advisor werden explizite Bucket-Berechtigungen und zugeordnete Bucket-Richtlinien geprüft, die die

Bucket-Berechtigungen möglicherweise überschreiben. Sie können auch mit AWS Config Ihre Amazon S3-Buckets für den öffentlichen Zugriff überwachen. Weitere Informationen finden Sie unter [How AWS Config to Monitor for and Respond to Amazon S3 Buckets Allowing Public Access](#).

Bei der Überprüfung der Zugriffskontrollen für Amazon-S3-Buckets ist es wichtig, die Art der darin gespeicherten Daten zu berücksichtigen. [Amazon Macie](#) ist ein Service, mit dem Sie sensible Daten wie persönlich identifizierbare Informationen (PII), geschützte Gesundheitsinformationen (PHI, Protected Health Information) und Anmeldeinformationen wie private Schlüssel oder AWS-Zugriffsschlüssel entdecken und schützen können.

Ressourcen

Zugehörige Dokumente:

- [Verwenden von AWS Identity and Access Management Access Analyzer](#)
- [Bibliothek von AWS Control Tower-Kontrollen](#)
- [AWS Foundational Security Best Practices-Standard](#)
- [AWS Config Von verwaltete Regeln](#)
- [AWS Trusted Advisor-Referenz prüfen](#)
- [Überwachen von AWS Trusted Advisor-Prüfungsergebnissen mit Amazon EventBridge](#)
- [Verwalten von AWS Config-Regeln für alle Konten in Ihrer Organisation](#)
- [AWS Config und AWS Organizations](#)
- [Ihr AMI für die Verwendung in Amazon EC2 öffentlich verfügbar machen](#)

Zugehörige Videos:

- [Bewährte Methoden für den Schutz Ihrer Mehrkonten-Umgebung](#)
- [Tiefer Einblick in IAM Access Analyzer](#)

SEC03-BP08 Sicheres gemeinsames Nutzen von Ressourcen in Ihrer Organisation

Wenn die Anzahl der Workloads zunimmt, müssen Sie möglicherweise den Zugriff auf Ressourcen in diesen Workloads ausweiten oder diese Ressourcen mehrfach über mehrere Konten hinweg zugänglich machen. Möglicherweise haben Sie Konstrukte zur Untergliederung Ihrer Umgebung, etwa für Entwicklungs-, Test- und Produktionsumgebungen. Solche Trennungskonstrukte

schränken Sie jedoch nicht in der Lage ein, sicher zu teilen. Durch die gemeinsame Nutzung sich überschneidender Ressourcen können Sie übermäßigen betrieblichen Aufwand reduzieren und eine konsistente Umgebung schaffen, ohne dass Sie raten müssen, was Sie vielleicht versäumt haben, wenn Sie eine Ressource mehrmals erstellen.

Gewünschtes Ergebnis: Vermeiden Sie den unbeabsichtigten Zugriff, indem Sie sichere Methoden verwenden, um Ressourcen innerhalb Ihrer Organisation zu teilen, und unterstützen Sie Ihre Initiative zur Verhinderung von Datenverlust. Reduzieren Sie Ihren organisatorischen Aufwand gegenüber der Verwaltung einzelner Komponenten, senken Sie die Zahl von Fehlern durch das manuelle mehrmalige Erstellen identischer Ressourcen, und steigern Sie die Skalierbarkeit Ihrer Workloads. Sie können von kürzeren Lösungszeiten in Szenarien mit mehreren Fehlerpunkten profitieren und Ihr Vertrauen in die Bestimmung erhöhen, wann eine Komponente nicht mehr benötigt wird. Verbindliche Anleitungen zur Analyse extern gemeinsam genutzter Ressourcen finden Sie unter [SEC03-BP07 Analysieren des öffentlichen und kontoübergreifenden Zugriffs](#).

Typische Anti-Muster:

- Fehlen eines Prozesses für die kontinuierliche Überwachung und die automatische Benachrichtigung bei unerwarteten externen Freigaben
- Fehlen einer Basislinie dazu, was freigegeben werden sollte und was nicht
- Die standardmäßige Verwendung einer sehr offenen Richtlinie, anstatt Ressourcen explizit freizugeben, wenn sie benötigt werden
- Manuelle Erstellung grundlegender Ressourcen bei Bedarf, die sich überlappen

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Gestalten Sie Ihre Zugriffskontrollen und -muster so, dass die Nutzung freigegebener Ressourcen kontrolliert wird und nur mit vertrauenswürdigen Entitäten möglich ist. Überwachen Sie freigegebene Ressourcen, prüfen Sie kontinuierlich den Zugriff darauf und erhalten Sie Benachrichtigungen bei unangemessenen oder unerwarteten Freigaben. Lesen Sie [Analysieren des öffentlichen und kontoübergreifenden Zugriffs](#), um eine Governance einzurichten, mit der Sie den externen Zugriff auf diejenigen Ressourcen beschränken können, die ihn benötigen, und um einen Prozess zur kontinuierlichen Überwachung und automatischen Warnung einzurichten.

Die kontoübergreifende gemeinsame Nutzung innerhalb von AWS Organizations wird durch [eine Reihe von AWS-Services unterstützt](#), etwa [AWS Security Hub CSPM](#), [Amazon GuardDuty](#) und

[AWS Backup](#). Diese Services ermöglichen die Freigabe von Daten für ein zentrales Konto, ihre Zugänglichkeit von einem zentralen Konto aus sowie die Verwaltung von Ressourcen und Daten von einem zentralen Konto aus. Beispielsweise kann AWS Security Hub CSPM Ergebnisse von einzelnen Konten auf ein zentrales Konto übertragen, wo Sie alle Ergebnisse einsehen können. AWS Backup kann eine Sicherungskopie einer Ressource kontoübergreifend freigeben. Sie können mit [AWS Resource Access Manager](#) (AWS RAM) weitere gängige Ressourcen freigeben, z. B. [VPC-Subnetze und Transit-Gateway-Anhänge](#), [AWS Network Firewall](#) oder [Amazon-SageMaker-AI-Pipelines](#).

Um Ihr Konto so zu beschränken, dass nur Ressourcen innerhalb Ihrer Organisation gemeinsam genutzt werden, verwenden Sie [Service-Kontrollrichtlinien \(SCP\)](#), um den Zugriff auf externe Prinzipale zu verhindern. Kombinieren Sie bei der gemeinsamen Nutzung von Ressourcen identitätsbasierte Kontrollen und Netzwerkkontrollen, um [einen Datenperimeter für Ihre Organisation zu schaffen](#), der zum Schutz vor unbeabsichtigtem Zugriff beiträgt. Ein Datenperimeter ist ein Satz von präventiven Maßnahmen zum Integritätsschutz, die dabei helfen, sicherzustellen, dass nur vertrauenswürdige Identitäten aus erwarteten Netzwerken auf vertrauenswürdige Ressourcen zugreifen. Diese Kontrollen begrenzen, welche Ressourcen gemeinsam genutzt werden, und verhindern die gemeinsame Nutzung oder Offenlegung von Ressourcen, die nicht zugelassen werden sollten. Als Teil Ihres Datenperimeters können Sie beispielsweise VPC-Endpunktrichtlinien und die `AWS:PrincipalOrgId`-Bedingung verwenden, um sicherzustellen, dass die Identitäten, die auf Ihre Amazon S3-Buckets zugreifen, zu Ihrer Organisation gehören. Es ist wichtig zu beachten, dass [SCPs nicht für servicebezogene Rollen oder AWS-Service-Prinzipale gelten](#).

Wenn Sie Amazon S3 verwenden, [deaktivieren Sie ACLs für Ihren Amazon S3-Bucket](#) und definieren Sie die Zugriffskontrolle mithilfe von IAM-Richtlinien. Zum [Beschränken des Zugriffs auf Amazon-S3-Inhalte](#) von [Amazon CloudFront](#) aus migrieren Sie von der Ursprungszugriffsidentität (OAI) zur Ursprungszugriffssteuerung (OAC), die zusätzliche Features wie die serverseitige Verschlüsselung mit [AWS Key Management Service](#) unterstützen.

In manchen Fällen möchten Sie möglicherweise die Freigabe von Ressourcen außerhalb Ihrer Organisation zulassen oder einer Drittpartei den Zugriff auf Ihre Ressourcen gewähren. Verbindliche Anleitungen zur Verwaltung von Berechtigungen für die externe gemeinsame Nutzung von Ressourcen finden Sie unter [Verwaltung von Berechtigungen](#).

Implementierungsschritte

1. Verwendung von AWS Organizations: AWS Organizations ist ein Kontoverwaltungsservice, mit dem Sie mehrere AWS-Konten in einer von Ihnen erstellten und zentral verwalteten Organisation konsolidieren können. Sie können Ihre Konten in Organisationseinheiten (OUs) gruppieren und jeder OU unterschiedliche Richtlinien zuweisen, um Ihre Budget-, Sicherheits- und Compliance-

- Anforderungen zu erfüllen. Sie können auch steuern, wie AWS-Services für künstliche Intelligenz (KI) und Machine Learning (ML) Daten erfassen und speichern können, und die Mehrkonten-Verwaltung der mit Organizations integrierten AWS-Services verwenden.
2. Integration von AWS Organizations mit AWS-Services: Wenn Sie einen AWS-Service zur Ausführung von Aufgaben in Ihrem Namen in den Mitgliedskonten Ihrer Organisation verwenden, erstellt AWS Organizations in jedem Mitgliedskonto eine serviceverknüpfte IAM-Rolle (SLR, Service-linked Role) für den jeweiligen Service. Sie sollten den vertrauenswürdigen Zugriff mit der AWS-Managementkonsole, den AWS-APIs oder der AWS CLI verwalten. Verbindliche Anleitungen zur Aktivierung des vertrauenswürdigen Zugriffs finden Sie unter [Verwendung von AWS Organizations mit anderen AWS-Services](#) und unter [AWS-Services, die Sie mit Organizations verwenden können](#).
 3. Einrichtung eines Datenperimeters: Ein Datenperimeter schafft eine klare Grenze zwischen Vertrauen und Zuständigkeit. In AWS wird er in der Regel als Ihre AWS-Organisation dargestellt, die von AWS Organizations verwaltet wird, zusammen mit allen On-Premises-Netzwerken oder -Systemen, die auf Ihre AWS-Ressourcen zugreifen. Das Ziel des Datenperimeters besteht darin, zu überprüfen, ob der Zugriff erlaubt ist, wenn die Identität und die Ressource vertrauenswürdig sind und es sich um ein erwartetes Netzwerk handelt. Die Einrichtung eines Datenperimeters ist jedoch kein Einheitslösung für alle. Evaluieren und übernehmen Sie die im [Whitepaper „Perimeter in AWS erstellen“](#) erläuterten Kontrollziele auf der Grundlage Ihrer spezifischen Sicherheitsrisikomodelle und -anforderungen. Sie sollten Ihre individuelle Risikosituation sorgfältig abwägen und die Perimeterkontrollen implementieren, die Ihren Sicherheitsanforderungen entsprechen.
 4. Gemeinsame Nutzung von Ressourcen in AWS-Services und entsprechende Einschränkung: Viele AWS-Services ermöglichen es Ihnen, Ressourcen mit einem anderen Konto gemeinsam zu nutzen oder eine Ressource in einem anderen Konto als Ziel zu verwenden, z. B. [Amazon Machine Images \(AMIs\)](#) und [AWS Resource Access Manager \(AWS RAM\)](#). Beschränken Sie die `ModifyImageAttribute`-API auf die Angabe der vertrauenswürdigen Konten, mit denen das AMI geteilt werden soll. Geben Sie bei der Verwendung von AWS RAM die `ram:RequestedAllowsExternalPrincipals`-Bedingung an, um die gemeinsame Nutzung nur auf Ihre Organisation zu beschränken, sodass der Zugriff durch nicht vertrauenswürdige Identitäten verhindert wird. Verbindliche Hinweise und Überlegungen finden Sie unter [Gemeinsame Nutzung von Ressourcen und externe Ziele](#).
 5. Verwendung von AWS RAM für die sichere gemeinsame Nutzung in einem Konto oder mit anderen AWS-Konten: [AWS RAM](#) unterstützt die sichere gemeinsame Nutzung der Ressourcen, die Sie erstellt haben, mit Rollen und Benutzern in Ihrem Konto sowie mit anderen AWS-Konten. In einer Mehrkonten-Umgebung ermöglicht AWS RAM die einmalige Erstellung einer Ressource

und ihre Freigabe für andere Konten. Dies reduziert Ihren operationalen Aufwand und sorgt für Konsistenz, Transparenz und Prüfbarkeit durch Integrationen mit Amazon CloudWatch und AWS CloudTrail, die bei Verwendung eines kontoübergreifenden Zugriffs nicht möglich sind.

Wenn Sie über Ressourcen verfügen, die Sie zuvor mithilfe einer ressourcenbasierten Richtlinie gemeinsam genutzt haben, können Sie die [PromoteResourceShareCreatedFromPolicy-API](#) oder eine gleichwertige Lösung verwenden, um die gemeinsame Nutzung auf eine vollständige AWS RAM-Ressourcenfreigabe hochzustufen.

In manchen Fällen müssen Sie möglicherweise weitere Schritte unternehmen, um Ressourcen freizugeben. Um beispielsweise einen verschlüsselten Snapshot zu teilen, müssen Sie [einen AWS KMS-Schlüssel freigeben](#).

Ressourcen

Zugehörige bewährte Methoden:

- [SEC03-BP07 Analysieren des öffentlichen und kontoübergreifenden Zugriffs](#)
- [SEC03-BP09 Sicheres Teilen von Ressourcen mit Dritten](#)
- [SEC05-BP01 Erstellen von Netzwerkebenen](#)

Zugehörige Dokumente:

- [Bucket-Besitzer gewährt kontoübergreifende Berechtigung für Objekte, die er nicht besitzt](#)
- [Verwendung von Vertrauensrichtlinien mit IAM](#)
- [Erstellen von Datenperimetern in AWS](#)
- [Verwenden einer externen ID, um Dritten Zugriff auf Ihre AWS-Ressourcen zu gewähren](#)
- [AWS-Services, die Sie mit AWS Organizations verwenden können](#)
- [Einrichtung eines Datenperimeters in AWS: Nur vertrauenswürdigen Identitäten den Zugriff auf Unternehmensdaten gestatten](#)

Zugehörige Videos:

- [Granulärer Zugriff mit AWS Resource Access Manager](#)
- [Schutz Ihres Datenperimeters mit VPC-Endpunkten](#)
- [Einrichten eines Datenperimeters in AWS](#)

Zugehörige Tools:

- [Beispiele für Richtlinien für Datenperimeter](#)

SEC03-BP09 Sicheres Teilen von Ressourcen mit Dritten

Die Sicherheit Ihrer Cloud-Umgebung endet nicht bei Ihrer Organisation. Möglicherweise stützt sich Ihre Organisation auf eine Drittpartei, um einen Teil Ihrer Daten zu verwalten. Das Berechtigungsmanagement für das von Dritten verwaltete System sollte dem Prinzip des Just-in-time-Zugriffs und dem der geringsten Berechtigung mit temporären Anmeldeinformationen folgen. Durch die enge Zusammenarbeit mit einer Drittpartei können Sie die möglichen Auswirkungen und das Risiko unbeabsichtigter Zugriffe gemeinsam senken.

Gewünschtes Ergebnis: Sie vermeiden die Verwendung von langfristigen AWS Identity and Access Management (IAM)-Anmeldeinformationen wie Zugriffsschlüsseln und geheimen Schlüsseln, da diese bei Missbrauch ein Sicherheitsrisiko darstellen. Stattdessen verwenden Sie IAM-Rollen und temporäre Anmeldeinformationen, um Ihre Sicherheitslage zu verbessern und den operativen Aufwand für die Verwaltung langfristiger Anmeldeinformationen zu minimieren. Wenn Sie Dritten Zugriff erteilen, verwenden Sie eine universell eindeutige Kennung (UUID, Universally Unique Identifier) als externe ID in der IAM-Vertrauensrichtlinie und lassen die IAM-Richtlinien an die Rolle unter Ihrer Kontrolle angefügt, um einen Zugriff mit geringsten Berechtigungen sicherzustellen. Eine verbindliche Anleitung für die Analyse extern freigegebener Ressourcen finden Sie unter [SEC03-BP07 Analysieren des öffentlichen und kontoübergreifenden Zugriffs](#).

Typische Anti-Muster:

- Verwendung der Standard-IAM-Vertrauensrichtlinie ohne Bedingungen
- Verwenden langfristiger IAM-Anmeldeinformationen und Zugriffsschlüssel
- Wiederverwendung externer IDs

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Möglicherweise möchten Sie die Freigabe von Ressourcen außerhalb von AWS Organizations zulassen oder einer Drittpartei den Zugriff auf Ihr Konto gewähren. So könnte etwa eine Drittpartei eine Überwachungslösung bereitstellen, die auf Ressourcen in Ihrem Konto zugreifen muss. In solchen Fällen sollten Sie eine kontoübergreifende IAM-Rolle erstellen, die nur über die von der

Drittpartei benötigten Berechtigungen verfügt. Definieren Sie außerdem eine Vertrauensrichtlinie mithilfe der [externen ID-Bedingung](#). Wenn eine externe ID verwendet wird, können Sie oder die Drittpartei eine eindeutige ID für jede(n) Kunden, Drittpartei oder Tenancy generieren. Die eindeutige ID sollte nach ihrer Erstellung ausschließlich von Ihnen kontrolliert werden. Die Drittpartei muss einen Prozess implementieren, durch den die externe ID in sicherer, prüfbarer und reproduzierbarer Weise dem Kunden zugeordnet wird.

Sie können [IAM Roles Anywhere](#) auch verwenden, um IAM-Rollen für Anwendungen außerhalb von AWS zu verwalten, die AWS-APIs verwenden.

Wenn die Drittpartei keinen Zugriff mehr auf Ihre Umgebung benötigt, entfernen Sie die Rolle. Vermeiden Sie die Weitergabe langfristiger Anmeldeinformationen an Dritte. Informieren Sie sich über andere AWS-Services zur Unterstützung von Freigaben, z. B. AWS Well-Architected Tool, der das [Freigeben eines Workloads](#) für andere AWS-Konten unterstützt, und [AWSResource Access Manager](#), der Ihnen hilft, eine AWS-Ressource in Ihrem Besitz auf sichere Weise für andere Konten freizugeben.

Implementierungsschritte

1. Verwenden Sie kontoübergreifende Rollen, um Zugriff auf externe Konten zu gewähren. [Kontoübergreifende Rollen](#) reduzieren die Menge vertraulicher Informationen, die von externen Konten und Dritten gespeichert werden, um ihre Kunden zu betreuen. Kontoübergreifende Rollen ermöglichen die sichere Gewährung des Zugriffs auf AWS-Ressourcen in Ihrem Konto für Drittparteien wie AWS-Partner oder andere Konten in Ihrer Organisation. Gleichzeitig wird die Möglichkeit gewahrt, diesen Zugriff zu verwalten und zu überprüfen. Möglicherweise stellt Ihnen die Drittpartei Dienstleistungen aus einer hybriden Infrastruktur heraus bereit oder ruft Daten zu einem anderen Standort ab. Mit [IAM Roles Anywhere](#) können Ihre Drittanbieter-Workloads sicher mit Ihren AWS-Workloads interagieren und die Notwendigkeit für langfristige Anmeldeinformationen weiter reduzieren.

Sie sollten keine langfristigen Anmeldeinformationen oder Benutzern zugeordnete Zugriffsschlüssel verwenden, um externen Zugriff auf Konten zu erteilen. Verwenden Sie stattdessen kontoübergreifende Rollen, um kontoübergreifenden Zugriff zu gewähren.

2. Führen Sie Due-Diligence-Prüfungen durch und sorgen Sie für einen sicheren Zugriff für SaaS-Drittanbieter. Führen Sie bei der Freigabe von Ressourcen für SaaS-Drittanbieter eine gründliche Due-Diligence-Prüfung durch, um sicherzustellen, dass diese beim Zugriff auf Ihre AWS-Ressourcen sicher und verantwortungsbewusst vorgehen. Evaluieren Sie ihr Modell der gemeinsamen Verantwortung, um zu verstehen, welche Sicherheitsmaßnahmen diese

Drittanbieter bereitstellen und für welche Bereiche Sie verantwortlich sind. Stellen Sie sicher, dass die SaaS-Anbieter über einen sicheren und überprüfbaren Prozess für den Zugriff auf Ihre Ressourcen verfügen, einschließlich der Verwendung [externer IDs](#) und des Prinzips des geringsten Zugriffs. Die Verwendung externer IDs trägt dazu bei, das [Confused-Deputy-Problem](#) zu lösen.

Implementieren Sie Sicherheitskontrollen, um einen sicheren Zugriff und die Einhaltung des Prinzips der geringsten Berechtigung sicherzustellen, wenn Sie SaaS-Drittanbietern Zugriff erteilen. Dies kann die Verwendung von externen IDs, Universally Unique Identifiers (UUIDs) und IAM-Vertrauensrichtlinien umfassen, die den Zugriff auf das unbedingt Notwendige einschränken. Arbeiten Sie eng mit dem SaaS-Anbieter zusammen, um sichere Zugriffsmechanismen einzurichten, den Zugriff auf Ihre AWS-Ressourcen regelmäßig zu überprüfen und Audits durchzuführen, um die Einhaltung Ihrer Sicherheitsanforderungen sicherzustellen.

3. Verwenden Sie vom Kunden bereitgestellte langfristige Anmeldeinformationen nicht mehr. Beenden Sie die Verwendung langfristiger Anmeldeinformationen, und verwenden Sie kontoübergreifende Rollen oder IAM Roles Anywhere. Wenn Sie langfristige Anmeldeinformationen verwendet müssen, formulieren Sie einen Plan für die Migration rollenbasierter Zugriffe. Einzelheiten zur Verwaltung von Schlüsseln finden Sie unter [Identitätsverwaltung](#). Sie sollten außerdem zusammen mit Ihrem AWS-Konto-Team und dem Drittanbieter ein Runbook für die Risikominderung erstellen. Verbindliche Anleitungen für Reaktionen auf Sicherheitsvorfälle und die Minderung ihrer potenziellen Auswirkungen finden Sie unter [Vorfallsreaktion](#).
4. Stellen Sie sicher, dass die Einrichtung über verbindliche Anleitungen verfügt oder automatisiert ist. Die externe ID wird nicht als Secret behandelt, ihr Wert darf aber nicht leicht zu erraten sein wie etwa eine Telefonnummer, ein Name oder eine Konto-ID. Machen Sie die externe ID zu einem schreibgeschützten Feld, damit sie nicht für illegitime Einrichtungen geändert werden kann.

Die externe ID kann von Ihnen oder von der Drittpartei generiert werden. Richten Sie einen Prozess ein, um festzulegen, wer für die Generierung der ID verantwortlich ist. Unabhängig von der Entität, die die externe ID erstellt, setzt die Drittpartei Eindeutigkeit und Formate in konsistenter Weise für alle Kunden durch.

Die Richtlinie, die für den kontoübergreifenden Zugriff in Ihren Konten erstellt wurde, muss dem [Prinzip der geringsten Berechtigung](#) entsprechen. Die Drittpartei muss ein Rollenrichtliniendokument oder einen automatisierten Einrichtungsmechanismus bereitstellen, der eine AWS CloudFormation-Vorlage oder ein Äquivalent verwendet. Dies reduziert die Gefahr von Fehlern durch die manuelle Erstellung von Richtlinien und bietet einen Überwachungspfad.

Weitere Informationen zur Verwendung einer AWS CloudFormation-Vorlage zum Erstellen kontoübergreifender Rollen finden Sie unter [Kontoübergreifende Rollen](#).

Die Drittpartei muss einen automatisierten und prüfbaren Einrichtungsmechanismus bereitstellen. Sie sollten jedoch die Einrichtung der Rolle automatisieren, indem Sie das Rollenrichtliniendokument verwenden, das den erforderlichen Zugriff angibt. Sie sollten mithilfe der AWS CloudFormation-Vorlage oder einer gleichwertigen Methode Änderungen überwachen. Die Erkennung von Abweichungen sollte Teil dieser Überwachung sein.

5. Berücksichtigen Sie Änderungen. Ihre Kontostruktur und Ihr Bedarf an einer Drittpartei bzw. deren Serviceangebots können sich über Nacht ändern. Sie sollten Änderungen und Ausfälle antizipieren und mit den richtigen Personen, Prozessen und Technologielösungen entsprechend planen. Prüfen Sie regelmäßig das von Ihnen bereitgestellte Zugriffsniveau und implementieren Sie Erkennungsverfahren, die Sie auf unerwartete Änderungen aufmerksam machen. Überwachen und prüfen Sie die Verwendung der externen Rolle und den Datenspeicher der externen IDs. Sie sollten darauf vorbereitet sein, den Zugriff der Drittpartei temporär oder dauerhaft zu widerrufen, wenn sich unerwartete Änderungen oder Zugriffsmuster ergeben. Messen Sie auch die Auswirkungen Ihrer Widerrufaktion, einschließlich der dafür benötigten Zeit, der involvierten Personen, der Kosten und der Auswirkungen auf andere Ressourcen.

Verbindliche Anleitungen zu Erkennungsmethoden finden Sie unter [Bewährte Methoden zur Erkennung](#).

Ressourcen

Zugehörige bewährte Methoden:

- [SEC02-BP02 Verwenden von temporären Anmeldeinformationen](#)
- [SEC03-BP05 Definieren eines Integritätsschutzes für Berechtigungen in Ihrer Organisation](#)
- [SEC03-BP06 Zugriffsverwaltung basierend auf dem Lebenszyklus](#)
- [SEC03-BP07 Analysieren des öffentlichen und kontoübergreifenden Zugriffs](#)
- [SEC04 Erkennung](#)

Zugehörige Dokumente:

- [Bucket-Besitzer gewährt kontoübergreifende Berechtigung für Objekte, die er nicht besitzt](#)
- [Verwenden von Vertrauensrichtlinien mit IAM-Rollen](#)

- [Delegieren des Zugriffs für AWS-Konten mithilfe von IAM-Rollen](#)
- [Wie greife ich mithilfe von IAM auf Ressourcen in einem anderen AWS-Konto zu?](#)
- [Bewährte Methoden für die Sicherheit in IAM](#)
- [Logik für die kontoübergreifende Richtlinienbewertung](#)
- [Verwenden einer externen ID beim Erteilen von Zugriff auf Ihre AWS-Ressourcen für Dritte](#)
- [Sammeln von Informationen in AWS CloudFormation-Ressourcen, die in externen Konten mit benutzerdefinierten Ressourcen erstellt wurden](#)
- [Sichere Verwendung einer externen ID für den Zugriff auf AWS-Konten im Besitz anderer Benutzer](#)
- [Erweitern von IAM-Rollen auf Workloads außerhalb von IAM mithilfe von IAM Roles Anywhere](#)

Zugehörige Videos:

- [Wie erteile ich Benutzern oder Rollen in einem separaten AWS-Konto Zugriff auf mein AWS-Konto?](#)
- [AWS re:Invent 2018: Experte für IAM-Richtlinien in unter 60 Minuten](#)
- [AWS Knowledge Center Live: Bewährte IAM-Methoden und Entwurfsentscheidungen](#)

Zugehörige Beispiele:

- [Kontenübergreifenden Zugriff auf Amazon DynamoDB konfigurieren](#)
- [AWS STS Network Query Tool](#)

Erkennung

Frage

- [SEC 4. Wie werden Sicherheitsereignisse erkannt und untersucht?](#)

SEC 4. Wie werden Sicherheitsereignisse erkannt und untersucht?

Erfassen und analysieren Sie Ereignisse anhand von Protokollen und Metriken, um mehr Transparenz zu erhalten. Reagieren Sie auf Sicherheitsereignisse und potenzielle Bedrohungen, um Ihre Workload zu schützen.

Best Practices

- [SEC04-BP01 Konfigurieren der Service- und Anwendungsprotokollierung](#)
- [SEC04-BP02 Erfassen von Protokollen, Erkenntnissen und Metriken an standardisierten Orten](#)
- [SEC04-BP03 Korrelieren und Anreichern von Sicherheitswarnmeldungen](#)
- [SEC04-BP04 Initiieren von Abhilfemaßnahmen für nicht konforme Ressourcen](#)

SEC04-BP01 Konfigurieren der Service- und Anwendungsprotokollierung

Bewahren Sie Protokolle zu Sicherheitsereignissen von Services und Anwendungen auf. Dies ist ein grundlegendes Sicherheitsprinzip für Prüfungs-, Untersuchungs- und betriebliche Anwendungsfälle und eine übliche Sicherheitsanforderung gemäß Governance-, Risiko- und Compliance (GRC)-Standards, -Richtlinien und -Prozeduren.

Gewünschtes Ergebnis: Eine Organisation sollte in der Lage sein, Sicherheitsereignisprotokolle in zuverlässiger und konsistenter Weise sowie zeitnah aus AWS-Services und -Anwendungen abzurufen, wenn diese für einen internen Prozess oder eine Verpflichtung wie etwa die Reaktion auf einen Sicherheitsvorfall benötigt werden. Erwägen Sie die Zentralisierung von Protokollen für bessere betriebliche Ergebnisse.

Typische Anti-Muster:

- Protokolle werden dauerhaft gespeichert oder zu früh gelöscht.
- Jeder kann auf die Protokolle zugreifen.
- Für die Verwaltung und Verwendung von Protokollen werden ausschließlich manuelle Prozesse genutzt.
- Alle Arten von Protokollen werden gespeichert, nur für den Fall, dass sie benötigt werden.
- Die Protokollintegrität wird nur bei Bedarf geprüft.

Vorteile der Nutzung dieser bewährten Methode: Implementieren Sie einen Mechanismus für die Ursachenanalyse (RCA) für Sicherheitsvorfälle sowie eine Evidenzquelle für Ihre Governance-, Risiko- und Compliance-Anforderungen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Bei einer Sicherheitsuntersuchung oder in anderen bedarfsabhängigen Anwendungsfällen müssen Sie relevante Protokolle konsultieren können, um alle Aspekte und den Zeitrahmen des Vorfalls

zu verstehen. Protokolle werden auch für die Generierung von Alarmen benötigt, die darauf hinweisen, dass bestimmte Ereignisse vorgekommen sind. Es ist sehr wichtig, Abfrage-, Abruf- sowie Benachrichtigungsmechanismen auszuwählen, zu aktivieren, zu speichern und einzurichten.

Implementierungsschritte

- Wählen und aktivieren Sie Protokollquellen. Vor einer Sicherheitsuntersuchung müssen Sie relevante Protokolle erfassen, um die Aktivitäten in einem AWS-Konto retroaktiv rekonstruieren zu können. Wählen und aktivieren Sie für Ihre Workloads relevante Protokollquellen.

Die Kriterien für die Auswahl der Protokollquelle sollten auf den Anwendungsfällen Ihres Unternehmens basieren. Richten Sie einen Trail für jedes AWS-Konto mit AWS CloudTrail oder einen AWS Organizations-Trail ein, und konfigurieren Sie dafür einen Amazon S3-Bucket.

AWS CloudTrail ist ein Protokollservice, der API-Aufrufe an ein AWS-Konto verfolgt und AWS-Serviceaktivitäten erfasst. Dieser ist standardmäßig mit einer 90-tägigen Aufbewahrung von Managementereignissen aktiviert, die [über den CloudTrail-Ereignisverlauf](#) mit der AWS-Managementkonsole, der AWS CLI oder einem AWS-SDK abgerufen werden können. Für längere Aufbewahrungszeiten und Abrufbarkeit von Datenereignissen [erstellen Sie einen CloudTrail-Trail](#) und verbinden diesen mit einem Amazon S3-Bucket sowie optional mit einer Amazon CloudWatch-Protokollgruppe. Sie können auch einen [CloudTrail-Lake](#) erstellen, der CloudTrail-Protokolle bis zu sieben Jahre lang aufbewahrt und eine SQL-basierte Abfragemöglichkeit bietet.

AWS empfiehlt, dass Kunden, die eine VPC nutzen, Netzwerkdatenverkehr- und DNS-Protokolle mit [VPC Flow Logs](#) und [Amazon Route 53 Resolver Query Logs](#) einrichten und diese per Stream zu einem Amazon S3-Bucket oder einer CloudWatch-Protokollgruppe leiten. Sie können ein VPC-Flow-Protokoll für eine VPC, ein Subnetz oder eine Netzwerkschnittstelle erstellen. Für VPC-Flow-Protokolle können Sie wählen, wie und wo Flow-Protokolle verwendet werden sollen, um Kosten zu sparen.

AWS CloudTrail CloudTrail-Protokolle, VPC-Flow-Protokolle und Route 53 Resolver Query Logs sind die grundlegenden Protokollquellen zur Unterstützung von Sicherheitsuntersuchungen in AWS. Sie können auch [Amazon Security Lake](#) verwenden, um diese Protokolldaten zu erfassen, zu normalisieren und im Apache Parquet-Format und mit dem Open Cybersecurity Schema Framework (OCSF) zu speichern, das Abfragen ermöglicht. Security Lake unterstützt auch andere AWS-Protokolle sowie Protokolle aus Drittquellen.

AWS-Services können Protokolle generieren, die von den grundlegenden Protokollquellen nicht erfasst werden, wie etwa Protokolle von Elastic Load Balancing, AWS WAF-Protokolle, Recorder-

Protokolle von AWS Config, Amazon GuardDuty-Erkenntnisse, Amazon Elastic Kubernetes Service (Amazon EKS)-Prüfprotokolle sowie Instance-Betriebssystem- und Anwendungsprotokolle von Amazon EC2. Eine vollständige Liste von Protokoll- und Überwachungslösungen finden Sie unter [Anhang A: Definitionen der Cloud-Funktionen – Protokollierung und Ereignisse](#) in der [Anleitung zur Reaktion auf AWS-Sicherheitsvorfälle](#).

- Untersuchen Sie die Protokollierungsmöglichkeiten für jede(n) AWS-Service und -Anwendung: Jede(r) AWS-Service und -Anwendung bietet Optionen für die Speicherung von Protokollen, jeweils mit eigenen Aufbewahrungs- und Lebenszyklus-Funktionen. Die beiden verbreitetsten Protokollspeicherservices sind Amazon Simple Storage Service (Amazon S3) und Amazon CloudWatch. Für lange Aufbewahrungszeiten wird die Verwendung von Amazon S3 empfohlen, wegen seiner Kosteneffektivität und der flexiblen Lebenszyklus-Funktionen. Wenn die primäre Protokollierungsoption Amazon CloudWatch Logs sind, sollten Sie erwägen, weniger häufig benötigte Protokolle in Amazon S3 zu archivieren.
- Wählen Sie den Protokollspeicher aus: Die Wahl des Protokollspeichers hängt generell vom verwendeten Abfragetool, den Aufbewahrungsfunktionen, der Vertrautheit damit und den Kosten ab. Die wichtigsten Optionen für die Protokollspeicherung sind ein Amazon S3-Bucket oder eine CloudWatch-Protokollgruppe.

Ein Amazon S3-Bucket bietet kosteneffektiven und dauerhaften Speicher mit optionaler Lebenszyklusrichtlinie. In Amazon S3-Buckets gespeicherte Protokolle können mit Services wie Amazon Athena abgefragt werden.

Eine CloudWatch-Protokollgruppe bietet dauerhaften Speicher und eine integrierte Abfragemöglichkeit über CloudWatch Logs Insights.

- Legen Sie die benötigte Aufbewahrungszeit für Protokolle fest: Wenn Sie einen Amazon S3-Bucket oder eine CloudWatch-Protokollgruppe für die Speicherung von Protokollen verwenden, müssen Sie adäquate Lebenszyklen für jede Protokollquelle einrichten, um Speicher- und Abrufkosten zu optimieren. Normalerweise haben Kunden Protokolle zwischen drei Monaten bis einem Jahr für Abfragen verfügbar, bei einer Gesamtaufbewahrungszeit von bis zu sieben Jahren. Die Wahl von Verfügbarkeit und Aufbewahrungszeit sollte sich nach Ihren Sicherheitsanforderungen und einer Kombination aus gesetzlichen, regulatorischen und unternehmensinternen Vorschriften richten.
- Aktivieren Sie die Protokollierung für jede(n) AWS-Service und -Anwendung mit korrekten Aufbewahrungs- und Lebenszyklusrichtlinien: Suchen Sie für alle AWS-Services oder -Anwendungen in Ihrer Organisation nach den entsprechenden Anleitungen zur Protokollkonfiguration:
 - [Konfigurieren eines AWS CloudTrail-Trails](#)

- [Konfigurieren von VPC-Flow-Protokollen](#)
 - [Konfigurieren des Amazon GuardDuty-Erkenntnisexports](#)
 - [Konfigurieren der AWS Config-Aufzeichnung](#)
 - [Konfigurieren des AWS WAF-Datenverkehrs von WAF](#)
 - [Konfigurieren der Netzwerkdatenverkehrsprotokolle von AWS Network Firewall](#)
 - [Konfigurieren der Zugriffsprotokolle von Elastic Load Balancing](#)
 - [Konfigurieren von Resolver-Abfrageprotokollen von Amazon Route 53](#)
 - [Konfigurieren von Amazon RDS-Protokollen](#)
 - [Konfigurieren von Amazon EKS-Steuerebenenprotokollen](#)
 - [Konfigurieren eines Amazon CloudWatch-Agenten für Amazon EC2-Instances und On-Premises-Server](#)
- Wählen und implementieren Sie Abfragemechanismen für Ihre Protokolle: Für Protokollabfragen können Sie [CloudWatch Logs Insights](#) für in CloudWatch-Protokollgruppen gespeicherte Daten sowie [Amazon Athena](#) und [Amazon OpenSearch Service](#) für in Amazon S3 gespeicherte Daten verwenden. Sie können auch Abfragetools von Drittanbietern wie etwa den SIEM (Security Information and Event Management)-Service verwenden.

Bei der Auswahl eines Tools zur Protokollabfrage sollten Sie die Personen, die Prozesse und die Technologieaspekte Ihrer Sicherheitsoperationen berücksichtigen. Wählen Sie ein Tool, das betriebliche, geschäftliche und sicherheitsrelevante Aspekte berücksichtigt und langfristig sowohl zugänglich als auch wartbar ist. Denken Sie daran, dass Tools zur Protokollabfrage optimal funktionieren, wenn die Anzahl der zu durchsuchenden Protokolle im Rahmen der Limits des jeweiligen Tools liegt. Es ist nicht ungewöhnlich, aus Kostengründen oder aufgrund technischer Einschränkungen mehrere Abfragetools zu verwenden.

Beispielsweise können Sie ein SIEM-Tool eines Drittanbieters für Abfragen der letzten 90 Datentage, aber aufgrund der Protokollerfassungskosten für SIEM Athena für Abfragen verwenden, die darüber hinaus gehen. Prüfen Sie unabhängig von der Implementierung, ob Ihr Konzept die Anzahl der für die Maximierung der operationalen Effizienz erforderlichen Tools minimiert, besonders für Untersuchungen von Sicherheitsvorfällen.

- Verwenden Sie Protokolle für Benachrichtigungen: AWS bietet verschiedene Benachrichtigungsmöglichkeiten über mehrere Sicherheitsservices:
 - [AWS Config](#) überwacht und zeichnet Ihre AWS-Ressourcenkonfigurationen auf. Darüber hinaus ermöglicht es Ihnen, die Auswertung und Korrektur der gewünschten Konfigurationen zu automatisieren.

- [Amazon GuardDuty](#) ist ein Bedrohungserkennungsservice, der kontinuierlich nach schädlichen Aktivitäten und nicht autorisierten Verhaltensweisen sucht, um Ihre AWS-Konten und Ihre Workloads zu schützen. GuardDuty erfasst, aggregiert und analysiert Informationen aus Quellen wie AWS CloudTrail-Verwaltungs- und Datenereignissen, DNS-Protokollen, VPC-Flow-Protokollen und Amazon EKS-Prüfprotokollen. GuardDuty ruft unabhängige Datenströme direkt von CloudTrail, VPC-Flow-Protokollen, DNS-Abfrageprotokollen und Amazon EKS ab. Sie müssen keine Amazon S3-Bucket-Richtlinien verwalten oder die Art und Weise der Erfassung und Speicherung von Protokollen verändern. Es wird jedoch empfohlen, diese Protokolle für Ihre eigenen Untersuchungs- und Compliance-Zwecke aufzubewahren.
- [AWS Security Hub CSPM](#) bietet einen zentralen Ort, an dem Ihre Sicherheitswarnungen oder Ergebnisse von mehreren AWS-Services und optionalen Produkten von Drittanbietern aggregiert, organisiert und priorisiert werden. So erhalten Sie einen umfassenden Überblick über Sicherheitswarnungen und den Compliance-Status.

Sie können auch benutzerdefinierte Alarm-Engines für Sicherheitsalarme verwenden, die von diesen Services nicht abgedeckt werden, bzw. für bestimmte Alarme, die für Ihre Umgebung relevante sind. Informationen zur Erstellung dieser Alarm- und Erkennungsmechanismen finden Sie unter [„Detection“ im AWS Security Incident Response Guide](#).

Ressourcen

Zugehörige bewährte Methoden:

- [SEC04-BP02 Erfassen von Protokollen, Erkenntnissen und Metriken an standardisierten Orten](#)
- [SEC07-BP04 Definieren eines skalierbaren Datenlebenszyklusmanagements](#)
- [SEC10-BP06 Vorabbereitstellen von Tools](#)

Zugehörige Dokumente:

- [AWS Security Incident Response Guide](#)
- [Erste Schritte mit Amazon Security Lake](#)
- [Getting started: Amazon CloudWatch Logs](#)

Zugehörige Videos:

- [AWS re:Invent 2.022 - Introducing Amazon Security Lake](#)

Zugehörige Beispiele:

- [Assisted Log Enabler for AWS](#)
- [AWS Security Hub CSPM Findings Historical Export](#)

SEC04-BP02 Erfassen von Protokollen, Erkenntnissen und Metriken an standardisierten Orten

Sicherheitsteams stützen sich auf Protokolle und Erkenntnisse, um Ereignisse zu analysieren, die auf unbefugte Aktivitäten oder unbeabsichtigte Änderungen hindeuten könnten. Um diese Analyse zu rationalisieren, sollten Sie Sicherheitsprotokolle und Ergebnisse an standardisierten Orten erfassen.

Dies macht Datenpunkte von Interesse für die Korrelation verfügbar und kann die Integration von Tools vereinfachen.

Gewünschtes Ergebnis: Sie verfügen über einen standardisierten Ansatz zum Sammeln, Analysieren und Visualisieren von Protokolldaten, Erkenntnissen und Metriken. Sicherheitsteams können Sicherheitsdaten über verschiedene Systeme hinweg effizient korrelieren, analysieren und visualisieren, um potenzielle Sicherheitsereignisse zu erkennen und Anomalien zu identifizieren. Systeme für Sicherheitsinformation und Ereignisverwaltung (Security Information and Event Management, SIEM) oder andere Mechanismen sind integriert, um Protokolldaten abzufragen und zu analysieren, damit Sie zeitnah auf Sicherheitsereignisse reagieren, diese verfolgen und eskalieren können.

Typische Anti-Muster:

- Teams besitzen und verwalten eigenständig Protokolle und Metriksammlungen, die nicht mit der Protokollierungsstrategie der Organisation übereinstimmen.
- Teams verfügen nicht über angemessene Zugriffskontrollen, um die Sichtbarkeit und Veränderung der erfassten Daten einzuschränken.
- Teams regeln ihre Sicherheitsprotokolle, Erkenntnisse und Metriken nicht als Teil ihrer Richtlinie zur Datenklassifizierung.
- Teams vernachlässigen bei der Konfiguration von Datensammlungen die Anforderungen an die Datenhoheit und die Lokalisierung.

Vorteile der Nutzung dieser bewährten Methode: Eine standardisierte Protokollierungslösung zur Erfassung und Abfrage von Protokolldaten und -ereignissen verbessert die aus den darin enthaltenen Informationen gewonnenen Erkenntnisse. Die Konfiguration eines automatisierten Lebenszyklus für die gesammelten Protokolldaten kann die durch die Speicherung von Protokollen entstehenden

Kosten reduzieren. Sie können eine fein abgestufte Zugriffskontrolle für die gesammelten Protokollinformationen einrichten, je nachdem, wie sensibel die Daten sind und welche Zugriffsmuster Ihre Teams benötigen. Sie können Tools integrieren, um die Daten zu korrelieren, zu visualisieren und Erkenntnisse daraus abzuleiten.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Die zunehmende AWS-Nutzung innerhalb einer Organisation führt zu einer wachsenden Anzahl von verteilten Workloads und Umgebungen. Jeder dieser Workloads und jede dieser Umgebungen generiert Daten über die darin stattfindenden Aktivitäten. Die Erfassung und lokale Speicherung dieser Daten stellt eine Herausforderung für den Sicherheitsbetrieb dar. Sicherheitsteams verwenden Tools wie Sicherheitsinformations- und Ereignisverwaltungssysteme (SIEM), um Daten aus verteilten Quellen zu sammeln und Korrelations-, Analyse- und Reaktionsabläufe durchzuführen. Dies erfordert die Verwaltung einer komplexen Reihe von Berechtigungen für den Zugriff auf die verschiedenen Datenquellen und einen zusätzlichen Aufwand beim Betrieb der Extract, Transform, Load (ETL)-Prozesse.

Um diese Herausforderungen zu bewältigen, sollten Sie alle relevanten Quellen von Sicherheitsprotokolldaten in einem Protokollarchiv-Konto zusammenfassen, wie in [Organisieren Ihrer AWS-Umgebung mittels mehrerer Konten](#) beschrieben. Dazu gehören alle sicherheitsrelevanten Daten aus Ihrem Workload und Protokolle, die AWS-Services erzeugen, wie [AWS CloudTrail](#), [AWS WAF](#), [Elastic Load Balancing](#) und [Amazon Route 53](#). Es hat mehrere Vorteile, diese Daten an standardisierten Orten in einem separaten AWS-Konto mit entsprechenden kontoübergreifenden Berechtigungen zu erfassen. Diese Vorgehensweise hilft, die Manipulation von Protokollen in gefährdeten Workloads und Umgebungen zu verhindern, bietet einen einzigen Integrationspunkt für zusätzliche Tools und bietet ein einfacheres Modell für die Konfiguration der Datenaufbewahrung und des Lebenszyklus. Bewerten Sie die Auswirkungen der Datenhoheit, der Compliance-Bereiche und anderer Vorschriften, um festzustellen, ob mehrere Speicherorte für Sicherheitsdaten und Aufbewahrungsfristen erforderlich sind.

Um die Erfassung und Standardisierung von Protokollen und Erkenntnissen zu erleichtern, bewerten Sie [Amazon Security Lake](#) in Ihrem Protokollarchiv-Konto. Sie können Security Lake so konfigurieren, dass Daten aus gängigen Quellen wie CloudTrail, Route 53, [Amazon EKS](#) und [VPC Flow Logs](#) automatisch aufgenommen werden. Außerdem können Sie AWS Security Hub CSPM auch als Datenquelle in Security Lake konfigurieren, sodass Sie Erkenntnisse aus anderen AWS-Services wie [Amazon GuardDuty](#) und [Amazon Inspector](#) mit Ihren Protokolldaten korrelieren können. Ferner haben Sie die Möglichkeit, Datenquellen von Drittanbietern zu integrieren oder eigene Datenquellen

zu konfigurieren. Alle Integrationen standardisieren Ihre Daten in das [Open Cybersecurity Schema Framework](#) (OCSF)-Format und werden in [Amazon S3](#)-Buckets als Parquet-Dateien gespeichert, sodass keine ETL-Verarbeitung erforderlich ist.

Die Speicherung von Sicherheitsdaten an standardisierten Orten bietet erweiterte Analysemöglichkeiten. AWS empfiehlt Ihnen die Bereitstellung von Tools für Sicherheitsanalysen, die in einer AWS-Umgebung arbeiten, in einem [Security-Tooling](#)-Konto, das von Ihrem Protokollarchiv-Konto getrennt ist. Dieser Ansatz ermöglicht es Ihnen, Kontrollen in der Tiefe zu implementieren, um die Integrität und Verfügbarkeit der Protokolle und des Protokollverwaltungsprozesses zu schützen, und zwar unabhängig von den Tools, die auf sie zugreifen. Erwägen Sie die Nutzung von Services wie [Amazon Athena](#), um On-Demand-Abfragen durchzuführen, die mehrere Datenquellen miteinander in Beziehung setzen. Sie können auch Visualisierungstools wie [QuickSight](#) integrieren. KI-gestützte Lösungen werden zunehmend verfügbar und können Funktionen wie die Übersetzung von Erkenntnissen in für Menschen lesbare Zusammenfassungen und Interaktion in natürlicher Sprache übernehmen. Diese Lösungen lassen sich oft leichter integrieren, wenn ein standardisierter Datenspeicher für Abfragen zur Verfügung steht.

Implementierungsschritte

1. Erstellen Sie die Konten „Protokollarchiv“ und „Security Tooling“
 - a. [Erstellen Sie mit AWS Organizations Organizations die Konten „Protokollarchiv“ und „Security Tooling“](#) unter einer Sicherheitsorganisationseinheit. Wenn Sie AWS Control Tower zur Verwaltung Ihrer Organisation verwenden, werden die Konten für Protokollarchiv und Security Tooling automatisch für Sie erstellt. Konfigurieren Sie bei Bedarf Rollen und Berechtigungen für den Zugriff auf diese Konten und deren Verwaltung.
2. Konfigurieren Sie Ihre standardisierten Speicherorte für Sicherheitsdaten
 - a. Legen Sie Ihre Strategie für die Erstellung standardisierter Sicherheitsdatenorte fest. Sie können dies durch Optionen wie allgemeine Data-Lake-Architekturansätze, Datenprodukte von Drittanbietern oder [Amazon Security Lake](#) erreichen. AWS empfiehlt, dass Sie Sicherheitsdaten von AWS-Regionen-Regionen erfassen, die [für Ihre Konten aktiviert](#) sind, auch wenn sie nicht aktiv genutzt werden.
3. Konfigurieren Sie die Veröffentlichung von Datenquellen an Ihren standardisierten Standorten
 - a. Identifizieren Sie die Quellen für Ihre Sicherheitsdaten und konfigurieren Sie sie so, dass sie an Ihren standardisierten Standorten veröffentlicht werden. Evaluieren Sie Optionen für den automatischen Export von Daten in das gewünschte Format im Gegensatz zu solchen, bei denen ETL-Prozesse entwickelt werden müssen. Mit Amazon Security Lake können Sie Daten aus unterstützten AWS-Quellen und integrierten Drittsystemen [sammeln](#).

4. Konfigurieren Sie Tools für den Zugriff auf Ihre standardisierten Speicherorte

- a. Konfigurieren Sie Tools wie Amazon Athena, QuickSight oder Lösungen von Drittanbietern, um den erforderlichen Zugriff auf Ihre standardisierten Standorte zu erhalten. Konfigurieren Sie diese Tools so, dass sie über das Security Tooling-Konto mit kontoübergreifendem Zugriff auf das Protokollarchiv-Konto arbeiten, sofern zutreffend. [Erstellen Sie Subscriber in Amazon Security Lake](#), um diesen Tools Zugriff auf Ihre Daten zu erteilen.

Ressourcen

Zugehörige bewährte Methoden:

- [SEC01-BP01 Trennen von Workloads mithilfe von Konten](#)
- [SEC07-BP04 Definieren eines skalierbaren Datenlebenszyklusmanagements](#)
- [SEC08-BP04 Durchsetzen der Zugriffskontrolle](#)
- [OPS08-BP02 Analysieren von Workload-Protokollen](#)

Zugehörige Dokumente:

- [AWS Whitepapers: Organizing Your AWS Environment Using Multiple Accounts](#)
- [AWS Prescriptive Guidance: AWS Security Reference Architecture \(AWS SRA\)](#)
- [AWS Prescriptive Guidance: Logging and monitoring guide for application owners](#)

Zugehörige Beispiele:

- [Aggregation, Suche und Visualisierung von Protokolldaten aus verteilten Quellen mit Amazon Athena und QuickSight](#)
- [Visualisieren von Ergebnissen von Amazon Security Lake mit QuickSight](#)
- [Generate AI powered insights for Amazon Security Lake using Amazon SageMaker AI Studio and Amazon Bedrock](#)
- [Identify cybersecurity anomalies in your Amazon Security Lake data using Amazon SageMaker AI](#)
- [Ingest, transform, and deliver events published by Amazon Security Lake to Amazon OpenSearch Service](#)
- [Simplify AWS CloudTrail log analysis with natural language query generation in CloudTrail Lake](#)

Zugehörige Tools:

- [Amazon Security Lake](#)
- [Amazon Security Lake-Partnerintegrationen](#)
- [Open Cybersecurity Schema Framework \(OCSF\)](#)
- [Amazon Athena](#)
- [Schnell](#)
- [Amazon Bedrock](#)

SEC04-BP03 Korrelieren und Anreichern von Sicherheitswarnmeldungen

Unerwartete Aktivitäten können mehrere Sicherheitswarnmeldungen aus verschiedenen Quellen auslösen, die eine weitere Korrelation und Anreicherung erfordern, um den gesamten Kontext zu verstehen. Implementieren Sie die automatische Korrelation und Anreicherung von Sicherheitswarnmeldungen, um eine genauere Identifizierung von Vorfällen und eine bessere Reaktion darauf zu ermöglichen.

Gewünschtes Ergebnis: Da die Aktivitäten in Ihren Workloads und Umgebungen unterschiedliche Warnmeldungen erzeugen, korrelieren automatische Mechanismen die Daten und reichern sie mit zusätzlichen Informationen an. Diese Vorverarbeitung ermöglicht ein detaillierteres Verständnis des Ereignisses, was Ihren Ermittlern hilft, die Kritikalität des Ereignisses zu bestimmen und festzustellen, ob es sich um einen Vorfall handelt, der eine formelle Reaktion erfordert. Dieses Verfahren entlastet Ihre Überwachungs- und Untersuchungsteams.

Typische Anti-Muster:

- Verschiedene Personengruppen untersuchen Erkenntnisse und Warnmeldungen, die von verschiedenen Systemen generiert werden, sofern nicht durch Anforderungen der Aufgabentrennung etwas anderes vorgeschrieben ist.
- Ihre Organisation leitet alle Sicherheitserkenntnisse und -warnmeldungen an Standardspeicherorte weiter, verlangt aber von den Ermittlern, dass sie diese manuell korrelieren und anreichern.
- Sie verlassen sich ausschließlich auf die Intelligenz von Bedrohungserkennungssystemen, um über Erkenntnisse zu berichten und die Kritikalität zu bestimmen.

Vorteile der Nutzung dieser bewährten Methode: Die automatische Korrelation und Anreicherung von Warnmeldungen trägt dazu bei, die gesamte kognitive Belastung und die manuelle

Datenaufbereitung zu reduzieren, die Ihre Ermittler benötigen. Diese Methode kann die Zeit verkürzen, die benötigt wird, um festzustellen, ob es sich bei dem Ereignis um einen Vorfall handelt, und eine formelle Reaktion einzuleiten. Zusätzlicher Kontext hilft Ihnen auch, den wahren Schweregrad eines Ereignisses genau zu bewerten, da er höher oder niedriger sein kann, als eine einzelne Warnmeldung vermuten lässt.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Niedrig

Implementierungsleitfaden

Sicherheitswarnmeldungen können von vielen verschiedenen Quellen innerhalb von AWS stammen, darunter:

- Services wie [Amazon GuardDuty](#), [AWS Security Hub CSPM](#), [Amazon Macie](#), [Amazon Inspector](#), [AWS Config](#), [AWS Identity and Access Management Access Analyzer](#) und [Network Access Analyzer](#)
- Warnmeldungen aus der automatisierten Analyse von AWS-Service-, Infrastruktur- und Anwendungsprotokollen, z. B. von [Security Analytics for Amazon OpenSearch Service](#).
- Warnungen als Reaktion auf Änderungen in Ihrer Abrechnungsaktivität aus Quellen wie [Amazon CloudWatch](#), [Amazon EventBridge](#) oder [AWS Budgets](#).
- Quellen von Drittanbietern wie Threat Intelligence Feeds und [Security Partner Solutions](#) vom AWS Partner Network
- [Kontakt durch AWS-Vertrauen und -Sicherheit](#) oder andere Quellen, wie Kunden oder interne Mitarbeiter.
- Verwenden Sie den [Threat Technique Catalog von AWS \(TTC\)](#), um das Verhalten von Bedrohungsakteuren anhand von Indicator of Compromise (IoC) zu identifizieren und zu korrelieren. Der TTC ist eine Erweiterung des MITRE ATT&CK-Frameworks und kategorisiert alle bekannten und beobachteten Verhaltensweisen und Techniken von Bedrohungsakteuren, die sich gegen AWS-Ressourcen richten.

In ihrer grundlegendsten Form enthalten Warnmeldungen Informationen darüber, wer (Prinzipal oder Identität) was (die Aktion, die ergriffen wird) im Hinblick worauf (Ressourcen, die betroffen sind) tut. Ermitteln Sie für jede dieser Quellen, ob es Möglichkeiten gibt, Zuordnungen zwischen den Identifikatoren für diese Identitäten, Aktionen und Ressourcen als Grundlage für die Durchführung von Korrelationen zu erstellen. Dies kann in Form einer Integration von Quellen für Warnmeldungen mit einem SIEM-Tool (Security Information and Event Management) erfolgen, das eine automatische

Korrelation für Sie durchführt, oder durch den Aufbau eigener Datenpipelines und -verarbeitung oder durch eine Kombination aus beidem.

Ein Beispiel für einen Dienst, der eine Korrelation für Sie durchführen kann, ist [Amazon Detective](#). Detective nimmt laufend Warnmeldungen aus verschiedenen AWS- und Drittquellen auf und nutzt verschiedene Formen von Informationen, um eine visuelle Grafik ihrer Beziehungen zur Unterstützung von Ermittlungen zusammenzustellen.

Während die anfängliche Kritikalität eines Alarms eine Hilfe für die Priorisierung ist, bestimmt der Kontext, in dem der Alarm auftrat, seine wahre Kritikalität. Zum Beispiel kann [Amazon GuardDuty](#) die Warnmeldung ausgeben, dass eine Amazon EC2-Instance innerhalb Ihres Workloads einen unerwarteten Domain-Namen abfragt. GuardDuty könnte dieser Warnmeldung von sich aus eine niedrige Kritikalität zuweisen. Eine automatische Korrelation mit anderen Aktivitäten zum Zeitpunkt der Warnmeldung könnte jedoch aufdecken, dass mehrere hundert EC2-Instances von derselben Identität bereitgestellt wurden, was die Gesamtbetriebskosten erhöht. In diesem Fall würde dieser korrelierte Ereigniskontext zur Veröffentlichung einer neuen Sicherheitswarnung führen. Die Kritikalität wird möglicherweise als hoch festgelegt, was die weiteren Maßnahmen beschleunigen würde.

Implementierungsschritte

1. Identifizieren Sie Quellen für Informationen zu Sicherheitswarnmeldungen. Verstehen Sie, wie Warnmeldungen aus diesen Systemen Identität, Aktion und Ressourcen darstellen, um festzustellen, wo eine Korrelation möglich ist.
2. Richten Sie einen Mechanismus zur Erfassung von Warnmeldungen aus verschiedenen Quellen ein. Ziehen Sie zu diesem Zweck Services wie Security Hub CSPM, EventBridge und CloudWatch in Betracht.
3. Identifizieren Sie Quellen für die Korrelation und Anreicherung von Daten. Zu den Beispielquellen gehören [AWS CloudTrail](#), [VPC-Flow-Protokolle](#), [Route-53-Resolver-Protokolle](#) sowie [Infrastruktur- und Anwendungsprotokolle](#). Einige oder alle dieser Protokolle können über eine einzige Integration mit [Amazon Security Lake](#) verwendet werden.
4. Integrieren Sie Ihre Warnmeldungen mit Ihren Datenkorrelations- und -anreicherungsquellen, um detailliertere Kontexte für Sicherheitsereignisse zu erstellen und die Kritikalität zu ermitteln.
 - a. Amazon Detective, SIEM-Tools oder andere Lösungen von Drittanbietern können ein gewisses Maß an Erfassung, Korrelation und Anreicherung automatisch durchführen.

- b. Sie können auch AWS-Services nutzen, um Ihre eigenen zu erstellen. Sie können zum Beispiel eine AWS Lambda-Funktion aufrufen, um eine Amazon Athena-Abfrage von AWS CloudTrail oder Amazon Security Lake auszuführen, und die Ergebnisse in EventBridge veröffentlichen.

Ressourcen

Zugehörige bewährte Methoden:

- [SEC10-BP03 Vorbereiten forensischer Funktionen](#)
- [OPS08-BP04 Erstellen umsetzbarer Warnmeldungen](#)
- [REL06-BP03 Senden von Benachrichtigungen \(Verarbeitung und Benachrichtigung in Echtzeit\)](#)

Zugehörige Dokumente:

- [AWS Security Incident Response Guide](#)

Zugehörige Beispiele:

- [How to enrich AWS Security Hub CSPM findings with account metadata](#)

Zugehörige Tools:

- [Amazon Detective](#)
- [Amazon EventBridge](#)
- [AWS Lambda](#)
- [Amazon Athena](#)

SEC04-BP04 Initiieren von Abhilfemaßnahmen für nicht konforme Ressourcen

Ihre detektivischen Kontrollen können Sie auf Ressourcen aufmerksam machen, die nicht mit Ihren Konfigurationsanforderungen übereinstimmen. Sie können programmatisch definierte Abhilfemaßnahmen einleiten, entweder manuell oder automatisch, um diese Ressourcen zu korrigieren und mögliche Auswirkungen zu minimieren. Wenn Sie Abhilfemaßnahmen programmatisch definieren, können Sie sofort und konsequent handeln.

Automatisierung kann zwar den Sicherheitsbetrieb verbessern, aber Sie sollten die Automatisierung sorgfältig implementieren und verwalten. Schaffen Sie geeignete Überwachungs- und Kontrollmechanismen, um zu überprüfen, ob die automatisierten Antworten effektiv und genau sind und mit den Organisationsrichtlinien und der Risikobereitschaft übereinstimmen.

Gewünschtes Ergebnis: Sie definieren Standards für die Ressourcenkonfiguration und die Schritte zur Behebung, wenn festgestellt wird, dass die Ressourcen nicht konform sind. Wo immer möglich, haben Sie Abhilfemaßnahmen programmatisch definiert, sodass sie entweder manuell oder durch Automatisierung eingeleitet werden können. Es gibt Erkennungssysteme, die nicht konforme Ressourcen identifizieren und Warnungen in zentralisierten Tools veröffentlichen, die von Ihrem Sicherheitspersonal überwacht werden. Diese Tools unterstützen die Durchführung Ihrer programmatischen Korrekturen, entweder manuell oder automatisch. Automatische Abhilfemaßnahmen verfügen über angemessene Überwachungs- und Kontrollmechanismen, um ihre Verwendung zu steuern.

Typische Anti-Muster:

- Sie implementieren Automatisierung, versäumen es aber, Abhilfemaßnahmen gründlich zu testen und zu validieren. Dies kann unbeabsichtigte Folgen haben, wie z. B. die Unterbrechung legitimer Geschäftsabläufe oder die Instabilität des Systems.
- Sie verbessern die Reaktionszeiten und Verfahren durch Automatisierung, aber ohne angemessene Überwachung und Mechanismen, die bei Bedarf menschliches Eingreifen und Urteilsvermögen ermöglichen.
- Sie verlassen sich ausschließlich auf Abhilfemaßnahmen, anstatt Abhilfemaßnahmen als Teil eines umfassenderen Programms zur Reaktion auf Vorfälle und zur Wiederherstellung zu nutzen.

Vorteile der Nutzung dieser bewährten Methode: Automatische Abhilfemaßnahmen können schneller auf Fehlkonfigurationen reagieren als manuelle Prozesse. So können Sie potenzielle Auswirkungen auf Ihr Unternehmen minimieren und das Zeitfenster für unbeabsichtigte Nutzungen verringern. Wenn Sie Abhilfemaßnahmen programmatisch definieren, werden sie konsistent angewendet, was das Risiko menschlicher Fehler verringert. Die Automatisierung kann auch eine größere Anzahl von Alarmen gleichzeitig verarbeiten, was besonders in Umgebungen von großem Maßstab wichtig ist.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Wie unter [SEC01-BP03 Identifizieren und Validieren von Kontrollzielen](#) beschrieben, können Services wie [AWS Config](#) und [AWS Security Hub CSPM](#) Ihnen dabei helfen, die Konfiguration der Ressourcen in Ihren Konten auf die Einhaltung Ihrer Anforderungen zu überwachen. Wenn nicht konforme Ressourcen erkannt werden, können Services wie AWS Security Hub CSPM bei der angemessenen Weiterleitung von Warnmeldungen und der Behebung von Problemen helfen. Diese Lösungen bieten einen zentralen Ort für Ihre Sicherheitsbeauftragten, um Probleme zu überwachen und Korrekturmaßnahmen zu ergreifen.

Zusätzlich zu AWS Security Hub CSPM führte AWS [Security Hub Advanced](#) ein. Dieser Service, der auf der re:Invent 2025 angekündigt wurde, verändert die Art und Weise, wie Unternehmen ihre kritischsten Sicherheitsprobleme priorisieren und in großem Umfang reagieren, um ihre Cloud-Umgebungen zu schützen. Der erweiterte Security Hub verwendet jetzt fortschrittliche Analytik, um Sicherheitssignale in Ihrer Cloud-Umgebung automatisch zu korrelieren, anzureichern und zu priorisieren. Security Hub lässt sich nahtlos in [Amazon GuardDuty](#), [Amazon Inspector](#), [Amazon Macie](#) und [AWS Security Hub CSPM](#) integrieren. Korrelierte Erkenntnisse in Security Hub können zu einem völlig neuen Ergebnis führen, das als „Bedrohungserkenntnis“ bezeichnet wird. Dazu gehört ein angenommener Angriffspfad, der auf den in jeder Ressource gefundenen Schwachstellen basiert.

Einige Situationen, in denen Ressourcen nicht konform sind, können zwar einzigartig sein und erfordern menschliches Urteilsvermögen, um Abhilfe zu schaffen. Für andere Fälle gibt es jedoch eine Standardreaktion, die Sie programmatisch definieren können. Eine Standardreaktion auf eine falsch konfigurierte VPC-Sicherheitsgruppe könnte zum Beispiel darin bestehen, die unzulässigen Regeln zu entfernen und den Eigentümer zu benachrichtigen. Antworten können in [AWS Lambda](#)-Funktionen, in [AWS-Systems-Manager-Automation](#)-Dokumenten oder durch andere von Ihnen bevorzugte Code-Umgebungen definiert werden. Vergewissern Sie sich, dass die Umgebung in der Lage ist, sich bei AWS zu authentifizieren, indem Sie eine IAM-Rolle mit der geringsten Berechtigung verwenden, die für die Durchführung von Korrekturmaßnahmen erforderlich ist.

Sobald Sie die gewünschte Abhilfemaßnahme definiert haben, können Sie festlegen, wie Sie diese einleiten möchten. AWS Config kann [Abhilfemaßnahmen für Sie einleiten](#). Wenn Sie Security Hub CSPM verwenden, können Sie dies über [benutzerdefinierte Aktionen](#) tun, wodurch die Erkenntnisinformationen in [Amazon EventBridge](#) veröffentlicht werden. Eine EventBridge-Regel kann dann Ihre Abhilfe einleiten. Sie können die Fehlerbehebungen in Security Hub CSPM für die automatische oder manuelle Ausführung konfigurieren.

Für programmatische Abhilfemaßnahmen empfehlen wir Ihnen, umfassende Protokolle und Audits für die durchgeführten Maßnahmen sowie deren Ergebnisse zu führen. Prüfen und

analysieren Sie diese Protokolle, um die Effektivität der automatisierten Prozesse zu bewerten und Verbesserungsmöglichkeiten zu identifizieren. Erfassen Sie Protokolle in [Amazon CloudWatch Logs](#) und Abhilfeergebnisse als [Erkenntnisse](#) in Security Hub CSPM.

Als Ausgangspunkt sollten Sie [Automatische Sicherheitsreaktion in AWS](#) verwenden, das über vorgefertigte Abhilfemaßnahmen zur Behebung häufiger Sicherheitsfehlkonfigurationen verfügt.

Implementierungsschritte

1. Analysieren und priorisieren Sie Warnmeldungen.
 - a. Konsolidieren Sie Sicherheitswarnungen von verschiedenen AWS-Services in Security Hub CSPM für eine zentrale Übersicht, Priorisierung und Abhilfe.
2. Entwickeln Sie Abhilfemaßnahmen.
 - a. Verwenden Sie Services wie Systems Manager und AWS Lambda, um programmatische Korrekturen durchzuführen.
3. Konfigurieren Sie, wie Abhilfemaßnahmen eingeleitet werden.
 - a. Definieren Sie mithilfe von Systems Manager benutzerdefinierte Aktionen, die Erkenntnisse an EventBridge veröffentlichen. Konfigurieren Sie diese Aktionen so, dass sie manuell oder automatisch ausgelöst werden.
 - b. Sie können auch [Amazon Simple Notification Service \(SNS\)](#) verwenden, um Benachrichtigungen und Warnmeldungen an relevante Beteiligte (wie das Sicherheitsteam oder das Vorfallsreaktionsteam) zu senden, damit diese bei Bedarf manuell eingreifen oder eskalieren können.
4. Prüfen und analysieren Sie die Protokolle der Abhilfemaßnahmen auf Wirksamkeit und Verbesserung.
 - a. Senden Sie die Protokollausgabe an CloudWatch Logs. Erfassen Sie die Ergebnisse als Erkenntnisse in Security Hub CSPM.

Ressourcen

Zugehörige bewährte Methoden:

- [SEC06-BP03 Reduzieren der manuellen Verwaltung und des interaktiven Zugriffs](#)

Zugehörige Dokumente:

- [AWS Security Incident Response Guide - Detection](#)

Zugehörige Beispiele:

- [Automatisierte Sicherheitsreaktion in AWS](#)
- [Monitor EC2 instance key pairs using AWS Config](#)
- [Create AWS Config custom rules by using AWS CloudFormation Guard policies](#)
- [Automatically remediate unencrypted Amazon RDS DB instances and clusters](#)

Zugehörige Tools:

- [AWS Systems Manager Automation](#)
- [Automatisierte Sicherheitsreaktion in AWS](#)

Schutz der Infrastruktur

Fragen

- [SEC 5. Wie werden Ihre Netzwerkressourcen geschützt?](#)
- [SEC 6. Wie werden Ihre Rechenressourcen geschützt?](#)

SEC 5. Wie werden Ihre Netzwerkressourcen geschützt?

Alle Workloads, die über eine Art Netzwerkverbindung verfügen, unabhängig davon, ob es sich um das Internet oder ein privates Netzwerk handelt, erfordern mehrere Abwehrebenen, um Schutz vor externen und internen Netzwerkbedrohungen sicherzustellen.

Best Practices

- [SEC05-BP01 Erstellen von Netzwerkebenen](#)
- [SEC05-BP02 Kontrollieren des Datenverkehrsflusses innerhalb Ihrer Netzwerkebenen](#)
- [SEC05-BP03 Implementieren Sie einen inspektionsbasierten Schutz](#)
- [SEC05-BP04 Automatisieren Sie den Netzwerkschutz](#)

SEC05-BP01 Erstellen von Netzwerkebenen

Segmentieren Sie Ihre Netzwerktopologie in verschiedene Ebenen, die auf logischen Gruppierungen Ihrer Workload-Komponenten entsprechend ihrer Datensensibilität und Zugriffsanforderungen basieren. Unterscheiden Sie zwischen Komponenten, auf die vom Internet aus zugegriffen werden

muss, wie z. B. öffentliche Web-Endpunkte, und solchen, die nur intern erreichbar sein müssen, wie z. B. Datenbanken.

Gewünschtes Ergebnis: Die Ebenen Ihres Netzwerks sind Teil eines ganzheitlichen, tiefgreifenden Sicherheitsansatzes, der die Identitätsauthentifizierungs- und Autorisierungsstrategie Ihrer Workloads ergänzt. Je nach Sensibilität der Daten und den Zugriffsanforderungen werden Ebenen mit entsprechenden Verkehrsfluss- und Kontrollmechanismen eingerichtet.

Typische Anti-Muster:

- Sie erstellen alle Ressourcen in einem einzigen VPC oder Subnetz.
- Sie erstellen Ihre Netzwerkebenen ohne Rücksicht auf die Anforderungen an die Datensensibilität, das Verhalten der Komponenten oder die Funktionalität.
- Sie verwenden VPCs und Subnetze als Standards für alle Aspekte der Netzwerkebenen und berücksichtigen nicht, wie verwaltete AWS-Services Ihre Topologie beeinflussen.

Vorteile der Nutzung dieser bewährten Methode: Die Einrichtung von Netzwerkebenen ist der erste Schritt, um unnötige Pfade durch das Netzwerk einzuschränken, insbesondere solche, die zu kritischen Systemen und Daten führen. Dadurch wird es für Unbefugte schwieriger, sich Zugriff auf Ihr Netzwerk zu verschaffen und zu weiteren Ressourcen darin zu navigieren. Diskrete Netzwerkebenen reduzieren den Umfang der Analyse für Inspektionssysteme, z. B. für die Erkennung von Eindringlingen oder die Verhinderung von Malware, vorteilhaft. Dadurch wird das Potenzial für Fehlalarme und unnötigen Verarbeitungsaufwand reduziert.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Beim Entwurf einer Workload-Architektur ist es üblich, die Komponenten je nach ihrer Verantwortlichkeit in verschiedene Ebenen aufzuteilen. Eine Webanwendung kann zum Beispiel eine Präsentationsebene, eine Anwendungsebene und eine Datenebene haben. Bei der Gestaltung Ihrer Netzwerktopologie können Sie einen ähnlichen Ansatz wählen. Die zugrunde liegenden Netzwerkkontrollen können dazu beitragen, die Anforderungen Ihres Workloads an den Datenzugriff durchzusetzen. In einer dreistufigen Webanwendungsarchitektur können Sie zum Beispiel Ihre statischen Präsentationsebenendateien in [Amazon S3](#) speichern und sie von einem Content Delivery Network (CDN) wie [Amazon CloudFront](#) aus bereitstellen. Die Anwendungsebene kann öffentliche Endpunkte haben, die ein [Application Load Balancer \(ALB\)](#) in einem [Amazon VPC](#)-öffentlichen Subnetz (ähnlich einer demilitarisierten Zone oder DMZ) bedient, während die Backend-Services

in privaten Subnetzen bereitgestellt werden. Die Datenebene, die Ressourcen wie Datenbanken und gemeinsam genutzte Dateisysteme hostet, kann sich in anderen privaten Subnetzen befinden als die Ressourcen Ihrer Anwendungsebene. An jeder dieser Ebenengrenzen (CDN, öffentliches Subnetz, privates Subnetz) können Sie Kontrollen bereitstellen, die es nur autorisiertem Datenverkehr erlauben, diese Grenzen zu überqueren.

Ähnlich wie bei der Modellierung von Netzwerkebenen auf der Grundlage des funktionalen Zwecks der Komponenten Ihres Workloads sollten Sie auch die Sensibilität der verarbeiteten Daten berücksichtigen. Wenn Sie das Beispiel der Webanwendung verwenden, kann es sein, dass alle Ihre Workload-Services innerhalb der Anwendungsebene angesiedelt sind, während verschiedene Services Daten mit unterschiedlichen Sensibilitätsstufen verarbeiten. In diesem Fall kann die Aufteilung der Anwendungsebene durch mehrere private Subnetze, verschiedene VPCs in demselben AWS-Konto oder sogar verschiedene VPCs in verschiedenen AWS-Konten für jede Stufe der Datensensibilität je nach Ihren Kontrollanforderungen angemessen sein.

Eine weitere Überlegung für Netzwerkebenen ist die Verhaltenskonsistenz der Komponenten Ihres Workloads. Um das Beispiel fortzusetzen: In der Anwendungsebene haben Sie möglicherweise Services, die Eingaben von Endbenutzern oder externen Systemintegrationen akzeptieren, die von Natur aus risikoreicher sind als die Eingaben für andere Services. Beispiele sind das Hochladen von Dateien, das Ausführen von Skripten, das Scannen von E-Mails und so weiter. Die Unterbringung dieser Services in einer eigenen Netzwerkebene hilft dabei, eine stärkere Isolationsgrenze um sie herum zu schaffen, und kann verhindern, dass ihr einzigartiges Verhalten falsche positive Alarme in Inspektionssystemen erzeugt.

Berücksichtigen Sie bei Ihrer Planung, wie die Nutzung von AWS verwalteten Services Ihre Netzwerktopologie beeinflusst. Erfahren Sie, wie Services wie [Amazon VPC Lattice](#) die Interoperabilität Ihrer Workload-Komponenten über Netzwerkebenen hinweg erleichtern können. Wenn Sie [AWS Lambda](#) verwenden, sollten Sie die Bereitstellung in Ihren VPC-Subnetzen vornehmen, es sei denn, es gibt besondere Gründe, die dagegen sprechen. Bestimmen Sie, wo VPC-Endpunkte und [AWS PrivateLink](#) die Einhaltung von Sicherheitsrichtlinien, die den Zugriff auf Internet-Gateways beschränken, vereinfachen können.

Implementierungsschritte

1. Überprüfen Sie Ihre Workload-Architektur. Gruppieren Sie Komponenten und Services logisch nach den Funktionen, die sie erfüllen, nach der Sensibilität der verarbeiteten Daten und nach ihrem Verhalten.

2. Für Komponenten, die auf Anfragen aus dem Internet reagieren, sollten Sie Load Balancer oder andere Proxys verwenden, um öffentliche Endpunkte bereitzustellen. Erkunden Sie die Verlagerung der Sicherheitskontrollen durch den Einsatz von verwalteten Services wie CloudFront, [Amazon API Gateway](#), Elastic Load Balancing und [AWS Amplify](#) zum Hosten öffentlicher Endpunkte.
3. Für Komponenten, die in Datenverarbeitungsumgebungen ausgeführt werden, wie Amazon EC2-Instances, [AWS Fargate](#)-Container oder Lambda-Funktionen, stellen Sie diese in privaten Subnetzen bereit, und zwar basierend auf Ihren Gruppen aus dem ersten Schritt.
4. Für vollständig verwaltete AWS-Services, wie [Amazon DynamoDB](#), [Amazon Kinesis](#) oder [Amazon SQS](#), sollten Sie VPC-Endpunkte als Standard für den Zugriff über private IP-Adressen verwenden.

Ressourcen

Zugehörige bewährte Methoden:

- [REL02 Planen der Netzwerktopologie](#)
- [PERF04-BP01 Verstehen der Auswirkungen des Netzwerks auf die Leistung](#)

Zugehörige Videos:

- [AWS re:Invent 2.023 - AWS networking foundations](#)

Zugehörige Beispiele:

- [VPC-Beispiele](#)
- [Greifen Sie privat auf Container-Anwendungen auf Amazon ECS zu, indem Sie AWS Fargate, AWS PrivateLink und einen Network Load Balancer verwenden](#)
- [Serve static content in an Amazon S3 bucket through a VPC by using Amazon CloudFront](#)

SEC05-BP02 Kontrollieren des Datenverkehrsflusses innerhalb Ihrer Netzwerkebenen

Verwenden Sie innerhalb der einzelnen Ebenen Ihres Netzwerks eine weitere Segmentierung, um den Datenverkehr auf die für die einzelnen Workloads erforderlichen Flüsse zu beschränken. Konzentrieren Sie sich zunächst auf die Kontrolle des Datenverkehrs zwischen dem Internet oder anderen externen Systemen eines Workloads und Ihrer Umgebung (Nord-Süd-Verkehr). Betrachten

Sie anschließend die Ströme zwischen verschiedenen Komponenten und Systemen (Ost-West-Verkehr).

Gewünschtes Ergebnis: Sie lassen nur die Netzwerkflüsse zu, die für die Kommunikation der Komponenten Ihrer Workloads untereinander, mit ihren Clients und mit allen anderen Services, von denen sie abhängig sind, erforderlich sind. Ihr Design berücksichtigt Überlegungen wie öffentlichen im Vergleich zu privatem Ingress und Egress, Datenklassifizierung, regionale Vorschriften und Protokollanforderungen. Wo immer es möglich ist, bevorzugen Sie Punkt-zu-Punkt-Flüsse gegenüber Netzwerk-Peering im Rahmen des Prinzips der geringsten Berechtigung.

Typische Anti-Muster:

- Sie verfolgen bei der Netzwerksicherheit einen Perimeter-basierten Ansatz und kontrollieren den Datenverkehr nur an den Grenzen Ihrer Netzwerkebenen.
- Sie gehen davon aus, dass der gesamte Verkehr innerhalb einer Netzwerkebene authentifiziert und autorisiert ist.
- Sie kontrollieren entweder den eingehenden oder den ausgehenden Datenverkehr, aber nicht beide.
- Sie verlassen sich bei der Authentifizierung und Autorisierung des Datenverkehrs ausschließlich auf Ihre Workload-Komponenten und Netzwerkkontrollen.

Vorteile der Nutzung dieser bewährten Methode: Diese Vorgehensweise trägt dazu bei, das Risiko unbefugter Bewegungen innerhalb Ihres Netzwerks zu verringern, und fügt Ihren Workloads eine zusätzliche Autorisierungsebene hinzu. Durch die Kontrolle des Datenverkehrs können Sie den Umfang der Auswirkungen eines Sicherheitsvorfalls begrenzen und die Erkennung und Reaktion beschleunigen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Netzwerkebenen helfen zwar bei der Abgrenzung von Komponenten Ihres Workloads, die eine ähnliche Funktion, eine ähnliche Datensensibilität und ein ähnliches Verhalten aufweisen. Sie können jedoch eine wesentlich feinere Ebene der Datenverkehrskontrolle schaffen, indem Sie Techniken zur weiteren Segmentierung von Komponenten innerhalb dieser Ebenen einsetzen, die dem Prinzip der geringsten Berechtigung folgen. Innerhalb von AWS werden Netzwerkebenen in erster Linie über Subnetze entsprechend den IP-Adressbereichen innerhalb eines Amazon VPC definiert. Ebenen

können auch über verschiedene VPCs definiert werden, z. B. für die Gruppierung von Microservice-Umgebungen nach Business Domain. Wenn Sie mehrere VPCs verwenden, vermitteln Sie das Routing mit einem [AWS Transit Gateway](#). Dies ermöglicht zwar die Kontrolle des Datenverkehrs auf Layer-4-Ebene (IP-Adressen- und Portbereiche) mithilfe von Sicherheitsgruppen und Routing-Tabellen, aber Sie können mit zusätzlichen Services, wie [AWS PrivateLink](#), [Amazon Route 53-Resolver-DNS-Firewall](#), [AWS Network Firewall](#) und [AWS WAF](#) weitere Kontrolle erlangen.

Verstehen und inventarisieren Sie den Datenfluss und die Kommunikationsanforderungen Ihrer Workloads in Bezug auf verbindungsauslösende Parteien, Ports, Protokolle und Netzwerkebenen. Prüfen Sie die verfügbaren Protokolle für den Verbindungsaufbau und die Datenübertragung, um diejenigen auszuwählen, die Ihre Schutzanforderungen erfüllen (z. B. HTTPS statt HTTP). Erfassen Sie diese Anforderungen sowohl an den Grenzen Ihrer Netzwerke als auch innerhalb jeder Ebene. Sobald diese Anforderungen identifiziert sind, prüfen Sie die Möglichkeiten, um nur den erforderlichen Datenverkehr an jedem Verbindungspunkt zuzulassen. Ein guter Ausgangspunkt ist die Verwendung von Sicherheitsgruppen innerhalb Ihrer VPC, da sie an Ressourcen angehängt werden können, die eine Elastic-Netzwerk-Schnittstelle (ENI) verwenden, wie Amazon EC2-Instances, Amazon ECS-Aufgaben, Amazon EKS-Pods oder Amazon RDS-Datenbanken. Im Gegensatz zu einer Layer-4-Firewall kann eine Sicherheitsgruppe eine Regel haben, die den Datenverkehr einer anderen Sicherheitsgruppe anhand ihrer Kennung zulässt, wodurch Aktualisierungen minimiert werden, wenn sich die Ressourcen innerhalb der Gruppe im Laufe der Zeit ändern. Sie können den Datenverkehr auch mithilfe von Sicherheitsgruppen nach eingehenden und ausgehenden Regeln filtern.

Wenn sich der Datenverkehr zwischen VPCs bewegt, ist es üblich, VPC-Peering für einfaches Routing oder AWS Transit Gateway für komplexes Routing zu verwenden. Mit diesen Ansätzen erleichtern Sie den Datenverkehrsfluss zwischen dem Bereich der IP-Adressen des Quell- und des Zielnetzwerks. Wenn Ihr Workload jedoch nur Datenverkehrsflüsse zwischen bestimmten Komponenten in verschiedenen VPCs erfordert, sollten Sie eine Punkt-zu-Punkt-Verbindung mit [AWS PrivateLink](#) verwenden. Bestimmen Sie dazu, welcher Service als Produzent und welcher als Verbraucher fungieren soll. Stellen Sie einen kompatiblen Load Balancer für den Produzenten bereit, schalten Sie PrivateLink entsprechend ein und akzeptieren Sie dann eine Verbindungsanfrage des Verbrauchers. Dem Produzenten-Service wird dann eine private IP-Adresse aus der VPC des Verbrauchers zugewiesen, die der Verbraucher für nachfolgende Anfragen verwenden kann. Dieser Ansatz reduziert die Notwendigkeit, die Netzwerke zu peeren. Beziehen Sie die Kosten für die Datenverarbeitung und den Load Balancer in die Bewertung von PrivateLink mit ein.

Sicherheitsgruppen und PrivateLink tragen zwar dazu bei, den Fluss zwischen den Komponenten Ihrer Workloads zu kontrollieren. Eine weitere wichtige Überlegung ist jedoch, wie Sie kontrollieren können, auf welche DNS-Domains Ihre Ressourcen zugreifen dürfen (falls überhaupt). Abhängig

von der DHCP-Konfiguration Ihrer VPCs können Sie zwei verschiedene AWS-Services für diesen Zweck in Betracht ziehen. Die meisten Kunden verwenden den standardmäßigen Route 53-Resolver DNS-Service (auch Amazon-DNS-Server oder AmazonProvidedDNS genannt), der für VPCs unter der +2-Adresse ihres CIDR-Bereichs verfügbar ist. Mit diesem Ansatz können Sie DNS-Firewall-Regeln erstellen und diese mit Ihrer VPC verknüpfen, die festlegen, welche Aktionen für die von Ihnen bereitgestellten Domain-Listen durchgeführt werden sollen.

Wenn Sie nicht den Route 53-Resolver verwenden, oder wenn Sie den Resolver mit tieferen Prüf- und Flusskontrollfunktionen als der Domain-Filterung ergänzen wollen, sollten Sie die Bereitstellung eines AWS Network Firewall erwägen. Dieser Service prüft einzelne Pakete anhand von zustandslosen oder zustandsbehafteten Regeln, um zu entscheiden, ob der Datenverkehr verweigert oder zugelassen werden soll. Einen ähnlichen Ansatz können Sie für die Filterung des eingehenden Internetdatenverkehrs zu Ihren öffentlichen Endpunkten mit AWS WAF verfolgen. Weitere Hinweise zu diesen Services finden Sie unter [SEC05-BP03 Implementieren eines prüfungsbasierten Schutzes](#).

Implementierungsschritte

1. Identifizieren Sie die erforderlichen Datenflüsse zwischen den Komponenten Ihrer Workloads.
2. Wenden Sie mehrere Kontrollen mit einem Ansatz der Tiefenverteidigung sowohl für den eingehenden als auch für den ausgehenden Datenverkehr an, einschließlich der Verwendung von Sicherheitsgruppen und Routing-Tabellen.
3. Verwenden Sie Firewalls, um eine feinkörnige Kontrolle über den Netzwerkverkehr in, aus und zwischen Ihren VPCs zu definieren, wie z. B. die Route 53 Resolver DNS Firewall, AWS Network Firewall und AWS WAF. Erwägen Sie den Einsatz von [AWS Firewall Manager](#) für die zentrale Konfiguration und Verwaltung Ihrer Firewall-Regeln in Ihrer Organisation.

Ressourcen

Zugehörige bewährte Methoden:

- [REL03-BP01 Segmentierung Ihres Workloads](#)
- [SEC09-BP02 Erzwingen einer Verschlüsselung bei der Übertragung](#)

Zugehörige Dokumente:

- [Security best practices for your VPC](#)

- [AWS Network Optimization Tips](#)
- [Guidance for Network Security on AWS](#)
- [Secure your VPC's outbound network traffic in the AWS Cloud](#)

Zugehörige Tools:

- [AWS Firewall Manager](#)

Zugehörige Videos:

- [AWS Transit Gateway reference architectures for many VPCs](#)
- [Application Acceleration and Protection with Amazon CloudFront, AWS WAF, and AWS Shield](#)
- [AWS re:Inforce 2023: Firewalls and where to put them](#)

SEC05-BP03 Implementieren Sie einen inspektionsbasierten Schutz

Richten Sie Kontrollpunkte für den Datenverkehr zwischen Ihren Netzwerkebenen ein, um sicherzustellen, dass die Daten während der Übertragung den erwarteten Kategorien und Mustern entsprechen. Analysieren Sie Datenverkehrsströme, Metadaten und Muster, um Ereignisse effektiver zu identifizieren, zu erkennen und darauf zu reagieren.

Gewünschtes Ergebnis: Der Datenverkehr, der zwischen Ihren Netzwerkebenen verläuft, wird geprüft und autorisiert. Entscheidungen über das Zulassen oder Verweigern von Zugriffen beruhen auf expliziten Regeln, Informationen über Bedrohungen und Abweichungen vom Grundverhalten. Der Schutz wird strenger, je näher der Datenverkehr an sensible Daten heranrückt.

Typische Anti-Muster:

- Ausschließlich auf Firewall-Regeln vertrauen, die auf Ports und Protokollen basieren, und Vorteile intelligenter Systeme außer Acht lassen
- Firewall-Regeln auf der Grundlage bestimmter aktueller Bedrohungsmuster erstellen, die sich ändern können
- Überprüfung des Datenverkehrs auf den Übergang von privaten zu öffentlichen Subnetzen oder von öffentlichen Subnetzen zum Internet beschränken
- Keine Basisansicht Ihres Netzwerkdatenverkehrs haben, die Sie auf Verhaltensanomalien hin überprüfen können

Vorteile der Nutzung dieser bewährten Methode: Prüfungssysteme ermöglichen es Ihnen, intelligente Regeln zu erstellen, z. B. den Datenverkehr nur dann zuzulassen oder zu verweigern, wenn bestimmte Bedingungen in den Datenverkehrsdaten vorliegen. Profitieren Sie von verwalteten Regelsätzen von AWS und Partnern, die auf den neuesten Bedrohungsinformationen basieren, da sich die Bedrohungslandschaft im Laufe der Zeit ändert. Dadurch verringert sich der Aufwand für die Pflege von Regeln und die Suche nach Indikatoren für eine Gefährdung, wodurch das Potenzial für Fehlalarme reduziert wird.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

[Verschaffen Sie sich mithilfe anderer Firewalls und Intrusion Prevention-Systeme \(IPS\) AWS Network Firewall, die Sie hinter einem Gateway Load Balancer \(\) einsetzen können, eine genaue Kontrolle über Ihren AWS Marketplace statusbehafteten und statusfreien Netzwerkverkehr. GWLB](#)
AWS Network Firewall unterstützt [Suricata-kompatible](#) Open-Source-Spezifikationen, um Ihre Workloads zu schützen. IPS

AWS Network Firewall Sowohl die Lösungen als auch die von Anbietern, die A verwenden, GWLB unterstützen unterschiedliche Bereitstellungsmodelle für Inline-Inspektionen. Sie können beispielsweise Inspektionen auf VPC Einzelbasis durchführen, sie in Form einer zentralen Inspektion durchführen oder sie in einem Hybridmodell einsetzen VPC, bei dem der Ost-West-Verkehr durch eine Inspektion fließt VPC und der Interneteingang einzeln geprüft wird. VPC Eine weitere Überlegung ist, ob die Lösung das Entpacken von Transport Layer Security (TLS) unterstützt, wodurch eine Deep Packet Inspection für Datenflüsse, die in beide Richtungen initiiert werden, ermöglicht wird. Weitere Informationen und ausführliche Details zu diesen Konfigurationen finden Sie im Leitfaden für [AWS Network Firewall Best Practices](#).

[Wenn Sie Lösungen verwenden, die out-of-band Inspektionen durchführen, z. B. die PCAP-Analyse von Paketdaten von Netzwerkschnittstellen, die im Promiscuous-Modus arbeiten, können Sie die Verkehrsspiegelung konfigurieren. VPC](#) Gespiegelter Datenverkehr wird auf die verfügbare Bandbreite Ihrer Schnittstellen angerechnet und unterliegt denselben Datenübertragungsgebühren wie nicht gespiegelter Datenverkehr. Sie können sehen, ob virtuelle Versionen dieser Appliances auf dem verfügbar sind [AWS Marketplace](#), was möglicherweise die Inline-Bereitstellung hinter einem unterstützt. GWLB

Schützen Sie Ihre Anwendung bei Komponenten, die über HTTP basierte Protokolle abgewickelt werden, mit einer Webanwendungs-Firewall (WAF) vor häufigen Bedrohungen. [AWS WAF](#) ist eine Firewall für Webanwendungen, mit der Sie Anfragen, die Ihren konfigurierbaren Regeln entsprechen,

überwachen und blockieren HTTP können, bevor sie an Amazon API Gateway CloudFront, Amazon AWS AppSync oder einen Application Load Balancer gesendet werden. Ziehen Sie Deep Packet Inspection in Betracht, wenn Sie den Einsatz Ihrer Webanwendungs-Firewall evaluieren, da Sie bei einigen Anwendungen den Vorgang TLS vor der Datenverkehrsinspektion beenden müssen. Zu Beginn können Sie AWS WAF es [Von AWS verwaltete Regeln](#) in Kombination mit Ihren eigenen Integrationen verwenden oder bestehende [Partnerintegrationen](#) verwenden.

Mit können Sie AWS WAF, AWS Shield Advanced AWS Network Firewall, und VPC Amazon-Sicherheitsgruppen in Ihrer gesamten AWS Organisation zentral verwalten [AWS Firewall Manager](#).

Implementierungsschritte

1. Stellen Sie fest, ob Sie die Inspektionsregeln breit fassen können VPC, z. B. durch eine Inspektion, oder ob Sie einen detaillierteren VPC Ansatz benötigen.
2. Für Inline-Prüfungslösungen:
 - a. Falls Sie diese verwenden AWS Network Firewall, erstellen Sie Regeln, Firewall-Richtlinien und die Firewall selbst. Sobald diese konfiguriert sind, können Sie den [Datenverkehr an den Endpunkt der Firewall leiten](#), um die Prüfung zu aktivieren.
 - b. Wenn Sie eine Appliance eines Drittanbieters mit einem Gateway Load Balancer (GWLB) verwenden, stellen Sie Ihre Appliance in einer oder mehreren Availability Zones bereit und konfigurieren Sie sie. Erstellen Sie dann Ihren GWLB Endpunktdienst und konfigurieren Sie das Routing für Ihren Datenverkehr.
3. Für out-of-band Inspektionslösungen:
 1. Aktivieren Sie VPC Traffic Mirroring auf Schnittstellen, an denen eingehender und ausgehender Datenverkehr gespiegelt werden soll. Sie können EventBridge Amazon-Regeln verwenden, um eine AWS Lambda Funktion aufzurufen, mit der die Verkehrsspiegelung auf Schnittstellen aktiviert wird, wenn neue Ressourcen erstellt werden. Richten Sie die Sitzungen zur Datenverkehrsspiegelung auf den Network Load Balancer vor Ihrer Appliance, der den Datenverkehr verarbeitet.
4. Für Lösungen für eingehenden Internetdatenverkehr:
 - a. Um zu konfigurieren AWS WAF, konfigurieren Sie zunächst eine Web-Zugriffskontrollliste (WebACL). Das Web ACL ist eine Sammlung von Regeln mit einer seriell verarbeiteten Standardaktion (ALLOW oder DENY), die definiert, wie Sie mit dem Datenverkehr WAF umgehen. Sie können Ihre eigenen Regeln und Gruppen erstellen oder AWS verwaltete Regelgruppen in Ihrem Web ACL verwenden.

- b. Sobald Ihr Web konfiguriert ACL ist, verknüpfen Sie das Web ACL mit einer AWS Ressource (wie einem Application Load Balancer, einem API Gateway oder einer CloudFront Distribution) RESTAPI, um mit dem Schutz des Webverkehrs zu beginnen.

Ressourcen

Zugehörige Dokumente:

- [What is Traffic Mirroring?](#)
- [Implementing inline traffic inspection using third-party security appliances](#)
- [AWS Network Firewall Beispielarchitekturen mit Routing](#)
- [Zentralisierte Inspektionsarchitektur mit AWS Gateway Load Balancer und AWS Transit Gateway](#)

Zugehörige Beispiele:

- [Best practices for deploying Gateway Load Balancer](#)
- [TLSInspektionskonfiguration für verschlüsselten Ausgangsverkehr und AWS Network Firewall](#)

Zugehörige Tools:

- [AWS Marketplace IDS/IPS](#)

SEC05-BP04 Automatisieren Sie den Netzwerkschutz

Automatisieren Sie die Implementierung Ihrer Netzwerkschutzmaßnahmen mithilfe von DevOps Methoden wie Infrastructure as Code (IaC) und CI/CD-Pipelines. Diese Praktiken können Ihnen helfen, Änderungen an Ihrem Netzwerkschutz über ein Versionskontrollsystem zu verfolgen, den Zeitaufwand für die Bereitstellung von Änderungen zu reduzieren und zu erkennen, wenn Ihr Netzwerkschutz von der gewünschten Konfiguration abweicht.

Gewünschtes Ergebnis: Sie definieren Netzwerkschutzmaßnahmen mit Vorlagen und übertragen diese in ein Versionskontrollsystem. Automatisierte Pipelines werden initiiert, wenn neue Änderungen vorgenommen werden, die ihre Prüfung und Bereitstellung orchestrieren.

Richtlinienprüfungen und andere statische Tests dienen der Validierung von Änderungen vor der Bereitstellung. Sie stellen die Änderungen in einer Staging-Umgebung bereit, um zu überprüfen, ob die Kontrollen wie erwartet funktionieren. Die Bereitstellung in Ihrer Produktionsumgebung erfolgt ebenfalls automatisch, sobald die Kontrollen genehmigt sind.

Typische Anti-Muster:

- Darauf vertrauen, dass die einzelnen Workload-Teams ihren kompletten Netzwerkstack, Schutzmaßnahmen und Automatisierungen selbst definieren Keine zentrale Veröffentlichung von Standardaspekten des Netzwerkstacks und der Schutzmechanismen für Workload-Teams zur Nutzung
- Auf ein zentrales Netzwerkteam vertrauen, das alle Aspekte des Netzwerks, der Schutzmaßnahmen und der Automatisierungen definiert Verzicht auf die Delegation von Workload-spezifischen Aspekten des Netzwerkstacks und der Schutzmaßnahmen an das Team des Workloads
- Beibehalten eines ausgewogenen Verhältnisses zwischen Zentralisierung und Delegation zwischen einem Netzwerkteam und Workload-Teams, aber keine Anwendung konsistenter Test- und Bereitstellungsstandards über Ihre IaC-Vorlagen und CI/CD-Pipelines hinweg Unterlassen der Erfassung erforderlicher Konfigurationen in Tools, die Ihre Vorlagen auf Einhaltung überprüfen

Vorteile der Nutzung dieser bewährten Methode: Durch die Verwendung von Vorlagen zur Definition Ihres Netzwerkschutzes können Sie Änderungen im Laufe der Zeit mit einem Versionskontrollsystem verfolgen und vergleichen. Der Einsatz von Automatisierung zum Testen und Bereitstellen von Änderungen schafft Standardisierung und Vorhersehbarkeit, erhöht die Chancen auf eine erfolgreiche Bereitstellung und reduziert die sich wiederholenden manuellen Konfigurationen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Eine Reihe von Netzwerkschutzmaßnahmen, die in [SEC05-BP02 Steuern Sie den Datenfluss innerhalb Ihrer Netzwerkschichten](#) und [SEC 05-BP03 Implementieren Sie inspektionsbasierten Schutz](#) beschrieben sind, verfügen über verwaltete Regelsysteme, die automatisch auf der Grundlage der neuesten Bedrohungsinformationen aktualisiert werden können. [Beispiele für den Schutz Ihrer Web-Endgeräte sind AWS WAF verwaltete Regeln und automatische Abwehr auf Anwendungsebene.](#) [AWS Shield Advanced DDoS](#) Verwenden Sie [von AWS Network Firewall verwaltete Regelgruppen](#), um auch bei Domain-Listen mit geringer Reputation und Bedrohungssignaturen auf dem Laufenden zu bleiben.

Neben verwalteten Regeln empfehlen wir Ihnen, DevOps Methoden zur Automatisierung der Bereitstellung Ihrer Netzwerkressourcen, Schutzmaßnahmen und der von Ihnen festgelegten Regeln zu verwenden. Sie können diese Definitionen in [AWS CloudFormation](#) oder einem anderen Infrastructure as Code (IaC)-Tool Ihrer Wahl erfassen, sie an ein Versionskontrollsystem übergeben

und sie über CI/CD-Pipelines bereitstellen. Nutzen Sie diesen Ansatz, um die traditionellen Vorteile der DevOps Verwaltung Ihrer Netzwerkkontrollen zu nutzen, z. B. vorhersehbarere Versionen, automatisierte Tests mit Tools wie [AWS CloudFormation Guard](#) und die Erkennung von Abweichungen zwischen Ihrer bereitgestellten Umgebung und der gewünschten Konfiguration.

Basierend auf den Entscheidungen, die Sie im Rahmen von [SEC05-BP01 Create Network Layers getroffen haben, verfolgen Sie möglicherweise einen zentralen Managementansatz für die Erstellung von Netzwerkschichten](#) VPCs, die für Eingangs-, Ausgangs- und Inspektionsabläufe vorgesehen sind. [Wie in der AWS Sicherheitsreferenzarchitektur \(AWS SRA\) beschrieben, können Sie diese VPCs in einem speziellen Netzwerkinfrastrukturkonto definieren.](#) Sie können ähnliche Techniken verwenden, um zentral die von Ihren Workloads in anderen Konten VPCs verwendeten Sicherheitsgruppen, AWS Network Firewall Bereitstellungen, Route 53-Resolver-Regeln und DNS Firewall-Konfigurationen sowie andere Netzwerkressourcen zu definieren. Sie können diese Ressourcen mit Ihren anderen Konten mit [AWS Resource Access Manager](#) teilen. Mit diesem Ansatz können Sie das automatisierte Testen und die Bereitstellung Ihrer Netzwerkkontrollen für das Netzwerkkonto vereinfachen, da Sie nur ein Ziel verwalten müssen. Sie können dies in einem hybriden Modell tun, bei dem Sie bestimmte Kontrollen zentral bereitstellen und gemeinsam nutzen und andere Kontrollen an die einzelnen Workload-Teams und ihre jeweiligen Konten delegieren.

Implementierungsschritte

1. Legen Sie fest, welche Aspekte des Netzwerks und des Schutzes zentral definiert werden und welche Ihre Workload-Teams verwalten können.
2. Erstellen Sie Umgebungen zum Testen und Bereitstellen von Änderungen an Ihrem Netzwerk und dessen Schutzmaßnahmen. Verwenden Sie zum Beispiel ein Netzwerk-Testkonto und ein Netzwerk-Produktionskonto.
3. Legen Sie fest, wie Sie Ihre Vorlagen in einem Versionskontrollsystem speichern und pflegen wollen. Speichern Sie zentrale Vorlagen in einem Repository, das sich von den Workload-Repositories unterscheidet, während Workload-Vorlagen in Repositories gespeichert werden können, die speziell für diesen Workload gelten.
4. Erstellen Sie CI/CD-Pipelines zum Testen und Bereitstellen von Vorlagen. Definieren Sie Tests, um zu prüfen, ob Fehlkonfigurationen vorliegen und ob die Vorlagen den Standards Ihres Unternehmens entsprechen.

Ressourcen

Zugehörige bewährte Methoden:

- [SEC01-BP06 Automatisieren Sie die Implementierung von Standard-Sicherheitskontrollen](#)

Zugehörige Dokumente:

- [AWS Security Reference Architecture - Network account](#)

Zugehörige Beispiele:

- [AWS Deployment Pipeline Reference Architecture](#)
- [NetDevSecOps zur Modernisierung von Netzwerkinstallationen AWS](#)
- [Integration von AWS CloudFormation Sicherheitstests und Berichten AWS Security Hub CSPMAWS CodeBuild](#)

Zugehörige Tools:

- [AWS CloudFormation](#)
- [AWS CloudFormation Guard](#)
- [cfn_nag](#)

SEC 6. Wie werden Ihre Rechenressourcen geschützt?

Die Rechenressourcen in Ihrer Workload erfordern mehrere Verteidigungsebenen, um sie vor externen und internen Bedrohungen zu schützen. Zu den Rechenressourcen gehören u. a. EC2-Instances, Container, AWS Lambda-Funktionen, Datenbankservices und IoT-Geräte.

Best Practices

- [SEC06-BP01 Schwachstellenmanagement](#)
- [SEC06-BP02 Bereitstellen von Rechenleistung aus gehärteten Images](#)
- [SEC06-BP03 Reduzieren der manuellen Verwaltung und des interaktiven Zugriffs](#)
- [SEC06-BP04 Softwareintegrität validieren](#)
- [SEC06-BP05 Automatisieren des Datenverarbeitungsschutzes](#)

SEC06-BP01 Schwachstellenmanagement

Überprüfen und Patchen Sie Ihren Code, Ihre Abhängigkeiten und Ihre Infrastruktur häufig auf Schwachstellen, um sich vor neuen Bedrohungen zu schützen.

Gewünschtes Ergebnis: Sie verfügen über eine Lösung, die Ihren Workload kontinuierlich auf Software-Schwachstellen, potenzielle Fehler und unbeabsichtigte Netzwerkrisiken überprüft. Sie haben Prozesse und Verfahren eingerichtet, um diese Schwachstellen basierend auf Risikobewertungskriterien zu identifizieren, zu priorisieren und zu beheben. Darüber hinaus haben Sie eine automatisierte Patch-Verwaltung für Ihre Datenverarbeitungs-Instances implementiert. Ihr Programm für das Schwachstellenmanagement ist in Ihren Softwareentwicklungszyklus integriert und bietet Lösungen zum Scannen Ihres Quellcodes in der CI/CD-Pipeline.

Typische Anti-Muster:

- Fehlen eines Programms für das Schwachstellenmanagement
- Durchführung von System-Patches ohne Berücksichtigung des Schweregrads oder der Risikovermeidung
- Verwendung von Software nach dem vom Anbieter angegebenen Lebenszyklusenddatum
- Bereitstellung von Code für die Produktion, bevor dieser auf Sicherheitsprobleme untersucht wurde

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Das Schwachstellenmanagement ist ein wichtiger Aspekt bei der Aufrechterhaltung einer sicheren und robusten Cloud-Umgebung. Es umfasst einen umfassenden Prozess, der Sicherheitsscans, die Identifizierung und Priorisierung von Problemen sowie Patch-Operationen zur Behebung der identifizierten Schwachstellen umfasst. Die Automatisierung spielt in diesem Prozess eine zentrale Rolle, da sie das kontinuierliche Scannen von Workloads auf potenzielle Probleme und unbeabsichtigte Netzwerkrisiken sowie die Durchführung von Abhilfemaßnahmen ermöglicht.

Das [AWS-Modell der gemeinsamen Verantwortung](#) ist ein Basiskonzept, das dem Schwachstellenmanagement zugrunde liegt. Gemäß diesem Modell ist AWS für die Sicherung der zugrunde liegenden Infrastruktur verantwortlich, einschließlich Hardware, Software, Netzwerk und der Einrichtungen, in denen AWS-Services ausgeführt werden. Umgekehrt sind Sie für die Sicherung Ihrer Daten, die Sicherheitskonfigurationen und die Verwaltungsaufgaben im Zusammenhang mit Services wie Amazon-EC2-Instances und Amazon-S3-Objekten verantwortlich.

AWS bietet verschiedene Services zur Unterstützung von Programmen für das Schwachstellenmanagement an. [Amazon Inspector](#) scannt AWS-Workloads kontinuierlich auf Software-Schwachstellen und unbeabsichtigte Netzwerkzugriffe, während [AWS Systems Manager Patch Manager](#) die Verwaltung von Patches für Amazon-EC2-Instances unterstützt. Diese Services können mit [AWS Security Hub CSPM](#) integriert werden, einem Service für das Management der Cloud-Sicherheit. Dieser Service automatisiert AWS-Sicherheitsprüfungen, zentralisiert Sicherheitswarnungen und stellt eine umfassende Übersicht über die Sicherheitslage einer Organisation bereit. Darüber hinaus verwendet [Amazon CodeGuru Security](#) Analysen des statischen Codes, um während der Entwicklungsphase potenzielle Probleme in Java- und Python-Anwendungen zu erkennen.

Durch die Integration von Verfahren für das Schwachstellenmanagement in den Software-Entwicklungszyklus können Sie Schwachstellen proaktiv beseitigen, bevor sie in Produktionsumgebungen eingeführt werden. Dies reduziert das Risiko von Sicherheitsvorfällen und die potenziellen Auswirkungen von Schwachstellen.

Implementierungsschritte

1. Machen Sie sich mit dem Modell der geteilten Verantwortung vertraut: Informieren Sie sich über das AWS-Modell der geteilten Verantwortung, um Ihre Verantwortung für die Sicherung Ihrer Workloads und Daten in der Cloud zu verstehen. AWS ist für die Sicherheit der zugrunde liegenden Cloud-Infrastruktur verantwortlich, während Sie für die Sicherheit Ihrer Anwendungen und Daten sowie der genutzten Services zuständig sind.
2. Implementieren Sie Schwachstellenscans: Konfigurieren Sie einen Service für das Scannen von Schwachstellen, z. B. Amazon Inspector, um Ihre Datenverarbeitungs-Instances (z. B. virtuelle Maschinen, Container oder Serverless-Funktionen) automatisch auf Software-Schwachstellen, potenzielle Fehler und unbeabsichtigte Netzwerkrisiken zu scannen.
3. Richten Sie Prozesse für das Schwachstellenmanagement ein: Definieren Sie Prozesse und Verfahren für die Identifizierung, Priorisierung und Behebung von Schwachstellen. Dies kann die Einrichtung von Zeitplänen für das regelmäßige Scannen auf Schwachstellen, die Festlegung von Kriterien für die Risikobewertung und die Definition von Zeitplänen für die Behebung basierend auf dem Schweregrad der Schwachstelle umfassen.
4. Richten Sie eine Patch-Verwaltung ein: Verwenden Sie einen Service für die Verwaltung von Patches, um das Patchen Ihrer Datenverarbeitungs-Instances zu automatisieren, sowohl für Betriebssysteme als auch für Anwendungen. Sie können den Service für das Scannen von Instances auf fehlende Patches und das automatische Installieren von Patches nach Zeitplan

- konfigurieren. Ziehen Sie AWS Systems Manager Patch Manager in Betracht, um diese Funktionalität bereitzustellen.
5. Konfigurieren Sie einen Malware-Schutz: Implementieren Sie Mechanismen für die Erkennung bössartiger Software in Ihrer Umgebung. Sie können beispielsweise Tools wie [Amazon GuardDuty](#) verwenden, um EC2- und EBS-Volumes hinsichtlich Malware zu analysieren, Malware zu erkennen und vor Malware zu warnen. GuardDuty kann auch neu zu Amazon S3 hochgeladene Objekte auf potenzielle Malware oder Viren scannen und Maßnahmen ergreifen, um sie vor der Aufnahme in nachgelagerte Prozesse zu isolieren.
 6. Integrieren Sie Schwachstellen-Scans in CI/CD-Pipelines: Wenn Sie eine CI/CD-Pipeline für Ihre Anwendungsbereitstellung verwenden, sollten Sie Tools zum Scannen auf Schwachstellen in Ihre Pipeline integrieren. Tools wie Amazon CodeGuru Security und Open-Source-Optionen können Quellcode, Abhängigkeiten und Artefakte auf potenzielle Sicherheitsprobleme scannen.
 7. Konfigurieren Sie einen Service für die Überwachung der Sicherheit: Richten Sie einen Service für die Überwachung der Sicherheit ein, z. B. AWS Security Hub CSPM, um einen umfassenden Überblick über Ihren Sicherheitsstatus über verschiedene Cloud-Services hinweg zu erhalten. Der Service sollte Erkenntnisse zur Sicherheit aus verschiedenen Quellen sammeln und in einem standardisierten Format anzeigen, um die Priorisierung und Behebung zu vereinfachen.
 8. Implementieren Sie Penetrationstests für Webanwendungen: Wenn es sich bei Ihrer Anwendung um eine Webanwendung handelt und Ihre Organisation über die erforderlichen Kompetenzen verfügt oder externe Unterstützung erhalten kann, sollten Sie die Implementierung von Penetrationstests für Webanwendungen in Betracht ziehen, um potenzielle Schwachstellen in Ihrer Anwendung zu identifizieren.
 9. Automatisieren Sie mit „Infrastructure as Code“: Verwenden Sie Infrastructure as Code (IAC)-Tools, z. B. [AWS CloudFormation](#), um die Bereitstellung und Konfiguration Ihrer Ressourcen zu automatisieren, einschließlich der zuvor genannten Sicherheitsservices. Dieses Verfahren hilft, eine konsistentere und standardisierte Ressourcenarchitektur für mehrere Konten und Umgebungen zu erstellen.
 10. Führen Sie Überwachungen durch und nehmen Sie kontinuierliche Verbesserungen vor: Überwachen Sie kontinuierlich die Effektivität Ihres Programms für das Schwachstellenmanagement und verbessern Sie es nach Bedarf. Überprüfen Sie die Sicherheitserkenntnisse, bewerten Sie die Effektivität Ihrer Abhilfemaßnahmen und passen Sie Ihre Prozesse und Tools entsprechend an.

Ressourcen

Zugehörige Dokumente:

- [AWS Systems Manager](#)
- [Security Overview of AWS Lambda](#)
- [Amazon CodeGuru](#)
- [Improved, Automated Vulnerability Management for Cloud Workloads with a New Amazon Inspector](#)
- [Automate vulnerability management and remediation in AWS using Amazon Inspector and AWS Systems Manager – Part 1](#)

Zugehörige Videos:

- [Securing Serverless and Container Services](#)
- [Security best practices for the Amazon EC2 instance metadata service](#)

SEC06-BP02 Bereitstellen von Rechenleistung aus gehärteten Images

Bieten Sie weniger Möglichkeiten für einen unbeabsichtigten Zugriff auf Ihre Laufzeitumgebungen, indem Sie sie über gehärtete Images bereitstellen. Beziehen Sie Laufzeit-Abhängigkeiten wie Container-Images und Anwendungsbibliotheken nur von vertrauenswürdigen Registern und überprüfen Sie deren Signaturen. Erstellen Sie Ihre eigenen privaten Register, um vertrauenswürdige Images und Bibliotheken für die Verwendung in Ihren Build- und Bereitstellungsprozessen zu speichern.

Gewünschtes Ergebnis: Ihre Datenverarbeitungsressourcen werden über gehärtete Baseline-Images bereitgestellt. Sie rufen externe Abhängigkeiten, wie Container-Images und Anwendungsbibliotheken, nur aus vertrauenswürdigen Registern ab und überprüfen deren Signaturen. Diese werden in privaten Registern gespeichert, auf die Ihre Build- und Bereitstellungsprozesse verweisen können. Sie überprüfen und aktualisieren Images und Abhängigkeiten regelmäßig, um sich vor neu entdeckten Schwachstellen zu schützen.

Typische Anti-Muster:

- Abrufen von Images und Bibliotheken aus vertrauenswürdigen Registern, ohne deren Signaturen zu überprüfen oder Schwachstellen zu scannen, bevor sie eingesetzt werden

- Härtung von Images, ohne sie regelmäßig auf neue Schwachstellen zu testen oder auf die neueste Version zu aktualisieren
- Installation oder Nichtentfernung von Softwarepaketen, die während des erwarteten Lebenszyklus des Images nicht benötigt werden
- Vertrauen auf Patches als einzige Methode, um Datenverarbeitungsressourcen in der Produktion auf dem neuesten Stand zu halten Die alleinige Verwendung von Patches kann immer noch dazu führen, dass Datenverarbeitungsressourcen im Laufe der Zeit von dem gehärteten Standard abweichen. Patches sind außerdem nicht in der Lage, Malware zu entfernen, die möglicherweise von einem Bedrohungsakteur während eines Sicherheitsvorfalls installiert wurde.

Vorteile der Nutzung dieser bewährten Methode: Das Härten von Images trägt dazu bei, die Anzahl der in Ihrer Laufzeitumgebung verfügbaren Pfade zu reduzieren, die unbeabsichtigten Zugriff auf nicht autorisierte Benutzer oder Services ermöglichen können. Auch das Ausmaß der Auswirkungen eines unbeabsichtigten Zugriffs kann damit verringert werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Um Ihre Systeme abzusichern, sollten Sie mit den neuesten Versionen von Betriebssystemen, Container-Images und Anwendungsbibliotheken beginnen. Wenden Sie Patches auf bekannte Probleme an. Reduzieren Sie das System auf ein Minimum, indem Sie nicht benötigte Anwendungen, Services, Gerätetreiber, Standardbenutzer und andere Anmeldeinformationen entfernen. Ergreifen Sie alle weiteren erforderlichen Maßnahmen, wie z. B. das Deaktivieren von Ports, um eine Umgebung zu schaffen, die nur über die von Ihren Workloads benötigten Ressourcen und Fähigkeiten verfügt. Von dieser Baseline aus können Sie dann Software, Agenten oder andere Prozesse installieren, die Sie für Zwecke wie die Überwachung des Workloads oder die Verwaltung von Schwachstellen benötigen.

Sie können die Belastung durch die Abhärtung von Systemen verringern, indem Sie sich an Anleitungen von vertrauenswürdigen Quellen wie dem [Center for Internet Security \(CIS\) und den Security Technical Implementation Guides](#) () der Defense Information Systems Agency (DISA) halten. STIGs Wir empfehlen Ihnen, mit einem [Amazon Machine Image](#) (AMI) zu beginnen, das von AWS oder einem APN Partner veröffentlicht wurde, und den AWS [EC2Image Builder](#) zu verwenden, um die Konfiguration gemäß einer geeigneten Kombination von CIS und STIG Kontrollen zu automatisieren.

Zwar sind gehärtete Images und EC2 Image Builder Builder-Rezepte verfügbar, die die CIS DISA STIG Oder-Empfehlungen anwenden, aber Sie stellen möglicherweise fest, dass Ihre Software

aufgrund ihrer Konfiguration nicht erfolgreich ausgeführt werden kann. In diesem Fall können Sie von einem nicht gehärteten Basis-Image ausgehen, Ihre Software installieren und dann schrittweise CIS Kontrollen anwenden, um deren Wirkung zu testen. Testen Sie bei allen CIS Kontrollen, die die Ausführung Ihrer Software verhindern, ob Sie stattdessen die detaillierteren Empfehlungen zur Absicherung implementieren können. DISA Behalten Sie den Überblick über die verschiedenen CIS Steuerungen und DISA STIG Konfigurationen, die Sie erfolgreich anwenden können. Verwenden Sie diese, um Ihre Rezepte für die Bildhärtung in EC2 Image Builder entsprechend zu definieren.

[Für containerisierte Workloads sind gehärtete Images von Docker im öffentlichen Repository von Amazon Elastic Container Registry \(\) ECR verfügbar.](#) Sie können EC2 Image Builder verwenden, um Container-Images gleichzeitig AMIs zu härten.

Ähnlich wie bei Betriebssystemen und Container-Images können Sie Codepakete (oder Bibliotheken) mithilfe von Tools wie pip, npm, Maven und aus öffentlichen Repositories abrufen. NuGet Wir empfehlen Ihnen, Code-Pakete zu verwalten, indem Sie private Repositories, wie z. B. innerhalb von [AWS CodeArtifact](#), mit vertrauenswürdigen öffentlichen Repositories verbinden. Diese Integration kann das Abrufen, Speichern und Aufbewahren von Paketen für Sie übernehmen. up-to-date Ihre Anwendungsentwicklungsverfahren können dann die neueste Version dieser Pakete zusammen mit Ihrer Anwendung abrufen und testen. Dabei kommen Techniken wie Software Composition Analysis (SCA), Static Application Security Testing (SAST) und Dynamic Application Security Testing (DAST) zum Einsatz.

Vereinfachen Sie für serverlose Workloads, die verwenden AWS Lambda, die Verwaltung von Paketabhängigkeiten mithilfe von [Lambda-Schichten](#). Verwenden Sie Lambda-Ebenen, um einen Satz von Standardabhängigkeiten, die von verschiedenen Funktionen gemeinsam genutzt werden, in einem eigenständigen Archiv zu konfigurieren. Sie können Ebenen mithilfe eines eigenen Build-Prozesses erstellen und verwalten, sodass Ihre Funktionen auf zentrale Weise erhalten bleiben. up-to-date

Implementierungsschritte

- Härten des Betriebssystems: Verwenden Sie Basis-Images aus vertrauenswürdigen Quellen als Grundlage für den Aufbau Ihres gehärteten SystemsAMIs. Verwenden Sie [EC2Image Builder](#), um die auf Ihren Images installierte Software anzupassen.
- Härten von containerisierten Ressourcen: Konfigurieren Sie containerisierte Ressourcen so, dass sie den bewährten Methoden im Bereich Sicherheit entsprechen. Wenn Sie Container verwenden, implementieren Sie [ECRImage Scanning](#) in Ihrer Build-Pipeline und regelmäßig anhand Ihres Image-Repositorys, um CVEs in Ihren Containern danach zu suchen.

- Wenn Sie die serverlose Implementierung mit verwenden AWS Lambda, verwenden Sie [Lambda-Schichten](#), um Anwendungsfunktionscode und gemeinsam genutzte abhängige Bibliotheken zu trennen. Konfigurieren Sie die [Codesignierung](#) für Lambda, um sicherzustellen, dass nur vertrauenswürdiger Code in Ihren Lambda-Funktionen ausgeführt wird.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS05-BP05 Führen Sie das Patch-Management durch](#)

Zugehörige Videos:

- [Tauchen Sie tief in die Sicherheit ein AWS Lambda](#)

Zugehörige Beispiele:

- [AMIMit EC2 Image Builder schnell STIG baukonform erstellen](#)
- [Building better container images](#)
- [Using Lambda layers to simplify your development process](#)
- [Entwickeln und implementieren Sie AWS Lambda Ebenen mit dem Serverless Framework](#)
- [Aufbau einer end-to-end AWS DevSecOps CI/CD-Pipeline mit Open Source SCA und Tools SAST DAST](#)

SEC06-BP03 Reduzieren der manuellen Verwaltung und des interaktiven Zugriffs

Nutzen Sie Automatisierung für die Bereitstellung, Konfiguration, Wartung und Untersuchung, wo immer dies möglich ist. Erwägen Sie den manuellen Zugriff auf Datenverarbeitungsressourcen in Notfällen oder in sicheren (Sandbox-)Umgebungen, wenn keine Automatisierung möglich ist.

Gewünschtes Ergebnis: Programmatische Skripte und Automatisierungsdokumente (Runbooks) erfassen autorisierte Aktionen in Ihren Datenverarbeitungsressourcen. Diese Runbooks werden entweder automatisch durch Systeme zur Erkennung von Änderungen oder manuell ausgelöst, wenn ein menschliches Urteilsvermögen erforderlich ist. Der direkte Zugriff auf Datenverarbeitungsressourcen wird nur in Notfällen gewährt, wenn keine Automatisierung verfügbar ist. Alle manuellen Aktivitäten werden protokolliert und in einen Überprüfungsprozess einbezogen, um Ihre Automatisierungsmöglichkeiten kontinuierlich zu verbessern.

Typische Anti-Muster:

- Interaktiver Zugriff auf Amazon EC2-Instances mit Protokollen wie SSH oder RDP.
- Verwalten einzelner Benutzeranmeldungen wie `/etc/passwd` oder lokaler Windows-Benutzer.
- Gemeinsame Nutzung eines Passworts oder privaten Schlüssels für den Zugriff auf eine Instance durch mehrere Benutzer.
- Manuelles Installieren von Software und Erstellen oder Aktualisieren von Konfigurationsdateien.
- Manuelles Aktualisieren oder Patchen von Software.
- Einloggen in eine Instance, um Probleme zu beheben.

Vorteile der Nutzung dieser bewährten Methode: Die Durchführung automatisierter Aktionen hilft Ihnen, das betriebliche Risiko unbeabsichtigter Änderungen und Fehlkonfigurationen zu verringern. Durch das Entfernen von Secure Shell (SSH) und Remote Desktop Protocol (RDP) für den interaktiven Zugriff wird der Umfang des Zugriffs auf Ihre Datenverarbeitungsressourcen reduziert. Damit wird ein gängiger Weg für unbefugte Aktionen abgeschnitten. Die Erfassung Ihrer Aufgaben zur Verwaltung von Datenverarbeitungsressourcen in Automatisierungsdokumenten und programmatischen Skripten bietet einen Mechanismus, mit dem Sie den gesamten Umfang der autorisierten Aktivitäten bis ins kleinste Detail definieren und überprüfen können.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Das Protokollieren einer Instance ist eine klassische Methode der Systemverwaltung. Nach der Installation des Server-Betriebssystems würden sich die Benutzer normalerweise manuell anmelden, um das System zu konfigurieren und die gewünschte Software zu installieren. Während der Lebensdauer des Servers melden sich die Benutzer möglicherweise an, um Software-Updates durchzuführen, Patches anzuwenden, Konfigurationen zu ändern und Probleme zu beheben.

Der manuelle Zugriff birgt jedoch eine Reihe von Risiken. Er erfordert einen Server, der auf Anfragen achtet, wie z. B. einen SSH- oder RDP-Service, der einen potenziellen Pfad für unbefugten Zugriff darstellen kann. Außerdem erhöht sich dadurch das Risiko menschlicher Fehler bei der Durchführung manueller Schritte. Diese können zu Störungen des Workloads, zur Beschädigung oder Zerstörung von Daten oder zu anderen Sicherheitsproblemen führen. Der menschliche Zugriff erfordert außerdem Schutzmaßnahmen gegen die Weitergabe von Anmeldeinformationen, was zusätzlichen Verwaltungsaufwand bedeutet.

Um diese Risiken abzuschwächen, können Sie eine agentenbasierte Remotezugriffslösung implementieren, wie z. B. [AWS Systems Manager](#). AWS Systems Manager Systems Manager-Agent (SSM Agent) initiiert einen verschlüsselten Kanal und ist daher nicht darauf angewiesen, auf von außen initiierte Anfragen zu achten. Erwägen Sie, SSM Agent so zu konfigurieren, dass er [diesen Kanal über einen VPC-Endpunkt aufbaut](#).

Systems Manager gibt Ihnen eine fein abgestufte Kontrolle darüber, wie Sie mit Ihren verwalteten Instances interagieren können. Sie legen fest, welche Automatisierungen ausgeführt werden sollen, wer sie ausführen darf und wann sie ausgeführt werden können. Systems Manager ist in der Lage, Patches anzuwenden, Software zu installieren und Konfigurationsänderungen ohne interaktiven Zugriff auf die Instance vorzunehmen. Systems Manager kann außerdem den Zugriff auf eine entfernte Shell ermöglichen und jeden während der Sitzung aufgerufenen Befehl und seine Ausgabe in Protokollen und [Amazon S3](#) protokollieren. [AWS CloudTrail](#) zeichnet Aufrufe von Systems Manager-APIs zur Überprüfung auf.

Implementierungsschritte

1. [Installieren Sie AWS Systems Manager Agent](#) (SSM Agent) auf Ihren Amazon EC2-Instances. Prüfen Sie, ob der SSM-Agent als Teil Ihrer AMI-Basiskonfiguration enthalten ist und automatisch gestartet wird.
2. Überprüfen Sie, ob die IAM-Rollen, die mit Ihren EC2-Instance-Profilen verbunden sind, die [verwaltete IAM-Richtlinie AmazonSSMManagedInstanceCore](#) enthalten.
3. Deaktivieren Sie SSH, RDP und andere Remotezugriffsservices, die auf Ihren Instances ausgeführt werden. Sie können dies tun, indem Sie Skripte ausführen, die im Abschnitt Benutzerdaten Ihrer Startvorlagen konfiguriert sind, oder indem Sie mit Tools wie EC2 Image Builder angepasste AMIs erstellen.
4. Vergewissern Sie sich, dass die für Ihre EC2-Instances geltenden Ingress-Regeln der Sicherheitsgruppe keinen Zugriff auf Port 22/tcp (SSH) oder Port 3389/tcp (RDP) zulassen. Implementieren Sie die Erkennung und Alarmierung bei falsch konfigurierten Sicherheitsgruppen mit Services wie AWS Config.
5. Definieren Sie entsprechende Automatisierungen, Runbooks und Run Commands in Systems Manager. Verwenden Sie IAM-Richtlinien, um festzulegen, wer diese Aktionen durchführen darf und unter welchen Bedingungen sie erlaubt sind. Testen Sie diese Automatisierungen gründlich in einer nicht produktiven Umgebung. Rufen Sie diese Automatisierungen bei Bedarf auf, anstatt interaktiv auf die Instance zuzugreifen.

6. Verwenden Sie [AWS Systems Manager Session Manager](#), um bei Bedarf interaktiven Zugriff auf Instances zu ermöglichen. Aktivieren Sie die Protokollierung der Sitzungsaktivitäten, um einen Audit Trail zu erstellen, in [Amazon CloudWatch Logs](#) oder [Amazon S3](#).

Ressourcen

Zugehörige bewährte Methoden:

- [REL08-BP04 Bereitstellung mit einer unveränderlichen Infrastruktur](#)

Zugehörige Beispiele:

- [Replacing SSH access to reduce management and security overhead with AWS Systems Manager](#)

Zugehörige Tools:

- [AWS Systems Manager](#)

Zugehörige Videos:

- [Controlling User Session Access to Instances in AWS Systems Manager Session Manager](#)

SEC06-BP04 Softwareintegrität validieren

Verwenden Sie die kryptografische Überprüfung, um die Integrität von Software-Artefakten (einschließlich Images) zu überprüfen, die Ihr Workload verwendet. Signieren Sie Ihre Software kryptografisch, um sie vor unbefugten Änderungen in Ihren Computerumgebungen zu schützen.

Gewünschtes Ergebnis: Alle Artefakte werden aus vertrauenswürdigen Quellen bezogen. Die Zertifikate der Website des Anbieters sind validiert. Heruntergeladene Artefakte werden anhand ihrer Signaturen kryptographisch verifiziert. Ihre eigene Software ist kryptografisch signiert und wird von Ihren Computerumgebungen überprüft.

Typische Anti-Muster:

- Vertrauen auf die Websites seriöser Anbieter, um Software-Artefakte zu erhalten, aber Hinweise zum Ablauf von Zertifikaten ignorieren Fortfahren mit dem Herunterladen, ohne zu bestätigen, dass die Zertifikate gültig sind

- Validieren der Zertifikate von Anbieter-Websites, aber keine kryptografische Überprüfung der heruntergeladenen Artefakte von diesen Websites
- Prüfen der Integrität von Software ausschließlich anhand von Digests oder Hashes Hashes stellen sicher, dass Artefakte gegenüber der ursprünglichen Version nicht verändert wurden, aber sie bestätigen nicht ihre Quelle.
- Nicht signieren Ihrer eigene Software, Ihres eigenen Codes oder Ihrer eigenen Bibliotheken, selbst wenn Sie sie nur in Ihren eigenen Bereitstellungen verwenden.

Vorteile der Nutzung dieser bewährten Methode: Die Überprüfung der Integrität von Artefakten, von denen Ihr Workload abhängt, hilft zu verhindern, dass Malware in Ihre Computerumgebungen eindringt. Das Signieren Ihrer Software schützt Sie davor, dass sie von Unbefugten in Ihrer Computerumgebung ausgeführt wird. Sichern Sie Ihre Softwarelieferkette durch Signieren und Verifizieren von Code.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Betriebssystem-Images, Container-Images und Code-Artefakte werden oft mit verfügbaren Integritätsprüfungen verteilt, z. B. durch einen Digest oder Hash. Diese ermöglichen es den Clients, die Integrität zu überprüfen, indem sie ihren eigenen Hash der Nutzdaten berechnen und überprüfen, ob er mit dem veröffentlichten Hash übereinstimmt. Diese Überprüfungen helfen zwar dabei, sicherzustellen, dass die Nutzdaten nicht manipuliert wurden, aber sie bestätigen nicht, dass die Nutzdaten von der ursprünglichen Quelle (ihrer Herkunft) stammen. Zur Überprüfung der Herkunft ist ein Zertifikat erforderlich, das eine vertrauenswürdige Stelle ausstellt, um das Artefakt digital zu signieren.

Wenn Sie in Ihrem Workload eine heruntergeladene Software oder Artefakte verwenden, prüfen Sie, ob der Anbieter einen öffentlichen Schlüssel für die Überprüfung der digitalen Signatur bereitstellt. Hier sind einige Beispiele dafür, wie AWS einen öffentlichen Schlüssel und Verifizierungsanweisungen für die von uns veröffentlichte Software bereitstellt:

- [EC2Image Builder: Überprüfen Sie die Signatur des AWS TOE Installationsdownloads](#)
- [AWS Systems Manager: Überprüfung der Signatur des Agenten SSM](#)
- [Amazon CloudWatch: Überprüfung der Signatur des CloudWatch Agentenpakets](#)

Integrieren Sie die Überprüfung digitaler Signaturen in die Prozesse, die Sie zum Abrufen und Härten von Images verwenden, wie unter [SEC06-BP02 Bereitstellung von Rechenleistung](#) aus gehärteten Images beschrieben.

Sie können [AWS Signer](#) verwenden, um die Überprüfung von Signaturen sowie Ihren eigenen Lebenszyklus der Codesignatur für Ihre eigene Software und Artefakte zu verwalten. Sowohl [AWS Lambda](#) als auch [Amazon Elastic Container Registry](#) bieten Integrationen mit Signer, um die Signaturen Ihres Codes und Ihrer Images zu überprüfen. Mit den Beispielen im Abschnitt Ressourcen können Sie Signer in Ihre Continuous Integration und Delivery (CI/CD) Pipelines einbinden, um die Überprüfung von Signaturen und die Signierung Ihres eigenen Codes und Ihrer Images zu automatisieren.

Ressourcen

Zugehörige Dokumente:

- [Cryptographic Signing for Containers](#)
- [Bewährte Methoden zur Sicherung Ihrer Pipeline für die Erstellung von Container-Images mithilfe von AWS Signer](#)
- [Ankündigung von Container Image Signing with AWS Signer und Amazon EKS](#)
- [Codesignatur konfigurieren für AWS Lambda](#)
- [Best practices and advanced patterns for Lambda code signing](#)
- [Codesignatur mit AWS Certificate Manager privater CA und AWS Key Management Service asymmetrischen Schlüsseln](#)

Zugehörige Beispiele:

- [Automatisieren Sie die Lambda-Code-Signierung mit Amazon CodeCatalyst und AWS Signer](#)
- [Signieren und Validieren von OCI Artefakten mit AWS Signer](#)

Zugehörige Tools:

- [AWS Lambda](#)
- [AWS Signer](#)
- [AWS Certificate Manager](#)
- [AWS Key Management Service](#)

- [AWS CodeArtifact](#)

SEC06-BP05 Automatisieren des Datenverarbeitungsschutzes

Automatisieren Sie den Datenverarbeitungsschutz, um das Erfordernis menschlichen Eingreifens zu reduzieren. Nutzen Sie automatisierte Scans, um potenzielle Probleme in Ihren Datenverarbeitungsressourcen zu erkennen und mit automatisierten programmatischen Reaktionen oder Flottenmanagement-Vorgängen zu beheben. Integrieren Sie die Automatisierung in Ihre CI/CD-Prozesse, um vertrauenswürdige Workloads mit aktuellen Abhängigkeiten bereitzustellen.

Gewünschtes Ergebnis: Automatisierte Systeme führen alle Scans und Patches von Datenverarbeitungsressourcen durch. Sie verwenden die automatische Überprüfung, um sicherzustellen, dass Software-Images und Abhängigkeiten aus vertrauenswürdigen Quellen stammen und nicht manipuliert wurden. Workloads werden automatisch auf aktuelle Abhängigkeiten geprüft und signiert, um die Vertrauenswürdigkeit in AWS-Datenverarbeitungsumgebungen zu gewährleisten. Automatisierte Abhilfemaßnahmen werden eingeleitet, wenn nicht konforme Ressourcen entdeckt werden.

Typische Anti-Muster:

- Verfolgen des Ansatzes einer unveränderlichen Infrastruktur, aber ohne eine Lösung für Notfall-Patches oder den Austausch von Produktionssystemen
- Verwenden von Automatisierung, um falsch konfigurierte Ressourcen zu korrigieren, ohne dass ein manueller Überschreibungsmechanismus vorhanden ist. Es können Situationen entstehen, in denen Sie die Anforderungen anpassen müssen, und es kann sein, dass Sie die Automatisierungen aussetzen müssen, bis Sie diese Änderungen vorgenommen haben.

Vorteile der Nutzung dieser bewährten Methode: Die Automatisierung kann das Risiko des unbefugten Zugriffs und der Nutzung Ihrer Datenverarbeitungsressourcen verringern. Sie hilft zu verhindern, dass Fehlkonfigurationen in Produktionsumgebungen gelangen, und Fehlkonfigurationen zu erkennen und zu beheben, wenn sie auftreten. Die Automatisierung hilft auch bei der Erkennung von unbefugtem Zugriff und der Nutzung von Datenverarbeitungsressourcen, um Ihre Reaktionszeit zu verkürzen. Dies wiederum kann den Gesamtumfang der Auswirkungen des Problems verringern.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Sie können die in den Methoden der Sicherheitssäule beschriebenen Automatisierungen zum Schutz Ihrer Datenverarbeitungsressourcen anwenden. In [SEC06-BP01 Schwachstellenmanagement](#) wird beschrieben, wie Sie [Amazon Inspector](#) sowohl in Ihren CI/CD-Pipelines als auch für die kontinuierliche Überprüfung Ihrer Laufzeitumgebungen auf bekannte CVEs (Common Vulnerabilities and Exposures) einsetzen können. Sie können [AWSSystems Manager](#) verwenden, um Patches anzuwenden oder neue Images über automatisierte Runbooks bereitzustellen, damit Ihre Computerflotte stets mit der neuesten Software und den neuesten Bibliotheken ausgestattet ist. Nutzen Sie diese Techniken, um den Bedarf an manuellen Prozessen und interaktivem Zugriff auf Ihre Datenverarbeitungsressourcen zu reduzieren. Weitere Informationen finden Sie unter [Reduzieren der manuellen Verwaltung und des interaktiven Zugriffs](#).

Die Automatisierung spielt auch eine Rolle bei der Bereitstellung von Workloads, die vertrauenswürdig sind. Dies wird in [SEC06-BP02 Bereitstellen von Datenverarbeitung über gehärtete Images](#) und [SEC06-BP04 Validieren der Softwareintegrität](#) beschrieben. Sie können Services wie [EC2 Image Builder](#), [AWS Signer](#), [AWS CodeArtifact](#) und [Amazon Elastic Container Registry \(ECR\)](#) verwenden, um gehärtete und genehmigte Images und Code-Abhängigkeiten herunterzuladen, zu überprüfen, zu erstellen und zu speichern. Neben Inspector kann jeder von ihnen eine Rolle in Ihrem CI/CD-Prozess spielen, sodass Ihr Workload nur dann in die Produktion geht, wenn sichergestellt ist, dass seine Abhängigkeiten aktuell sind und aus vertrauenswürdigen Quellen stammen. Ihr Workload ist außerdem signiert, damit AWS-Datenverarbeitungsumgebungen wie [AWS Lambda](#) und [Amazon Elastic Kubernetes Service \(EKS\)](#) überprüfen können, dass er nicht manipuliert wurde, bevor sie ihn ausführen.

Über diese präventiven Kontrollen hinaus können Sie die Automatisierung auch bei den detektivischen Kontrollen für Ihre Datenverarbeitungsressourcen einsetzen. Ein Beispiel: [AWS Security Hub CSPM](#) bietet den Standard [NIST 800-53 Rev. 5](#), der Prüfungen wie [\[EC2.8\] EC2 instances should use Instance Metadata Service Version 2 \(IMDSv2\)](#) enthält. IMDSv2 verwendet die Techniken der Sitzungsauthentifizierung, des Blockierens von Anfragen, die einen X-Forwarded-For HTTP-Header enthalten, und eine Netzwerk-TTL von 1, um den von externen Quellen stammenden Datenverkehr zum Abrufen von Informationen über die EC2-Instance zu stoppen. Diese Prüfung in Security Hub CSPM kann erkennen, wenn EC2 Instances IMDSv1 verwenden und eine automatische Abhilfe einleiten. Weitere Informationen zur automatisierten Erkennung und zu Abhilfemaßnahmen finden Sie unter [SEC04-BP04 Initiieren von Abhilfemaßnahmen für nicht konforme Ressourcen](#).

Implementierungsschritte

1. Automatisieren Sie die Erstellung sicherer, konformer und gehärteter AMIs mit [EC2 Image Builder](#). Sie können Images erstellen, die Kontrollen aus den Center for Internet Security (CIS)-Benchmarks oder Security Technical Implementation Guide (STIG)-Standards aus Basis- AWS und APN-Partner-Images enthalten.
2. Automatische Konfigurationsverwaltung. Erzwingen und validieren Sie sichere Konfigurationen in Ihren Datenverarbeitungsressourcen automatisch. Verwenden Sie dazu einen Service oder ein Tool zur Konfigurationsverwaltung.
 - a. Automatisiertes Konfigurationsmanagement mit [AWS Config](#)
 - b. Automatisiertes Sicherheits- und Compliance-Management mit [AWS Security Hub CSPM](#)
3. Automatisieren Sie das Patchen oder Ersetzen von Amazon Elastic Compute Cloud (Amazon EC2)-Instances. AWS Systems Manager Patch Manager automatisiert das Patchen verwalteter Instances mit sicherheitsrelevanten und anderen Arten von Updates. Sie können Patchmanager verwenden, um Patches sowohl für Betriebssysteme als auch für Anwendungen durchzuführen.
 - a. [AWS Systems Manager Patch Manager](#)
4. Automatisieren Sie das Scannen von Datenverarbeitungsressourcen auf häufige Schwachstellen und Gefährdungen (CVEs) und betten Sie Sicherheitsscan-Lösungen in Ihre Build-Pipeline ein.
 - a. [Amazon Inspector](#)
 - b. [ECR Image Scanning](#)
5. Ziehen Sie Amazon GuardDuty für die automatische Erkennung von Malware und Bedrohungen in Betracht, um Datenverarbeitungsressourcen zu schützen. GuardDuty kann außerdem mögliche Probleme identifizieren, wenn eine [AWS Lambda](#)-Funktion in Ihrer AWS-Umgebung aufgerufen wird.
 - a. [Amazon GuardDuty](#)
6. Ziehen Sie AWS-Partnerlösungen in Betracht. AWS -Partner bieten branchenführende Produkte an, die mit vorhandenen Kontrollen in Ihren On-Premises-Umgebungen gleichwertig oder identisch sind oder sich in diese integrieren lassen. Diese Produkte ergänzen die vorhandenen AWS-Services, sodass Sie eine umfassende Sicherheitsarchitektur bereitstellen und eine nahtlosere Erfahrung in Ihren Cloud- und On-Premises-Umgebungen ermöglichen können.
 - a. [Sicherheit der Infrastruktur](#)

Ressourcen

Zugehörige bewährte Methoden:

- [SEC01-BP06 Automatisieren der Bereitstellung von Standard-Sicherheitskontrollen](#)

Zugehörige Dokumente:

- [Get the full benefits of IMDSv2 and disable IMDSv1 across your AWS infrastructure](#)

Zugehörige Videos:

- [Security best practices for the Amazon EC2 instance metadata service](#)

Datenschutz

Fragen

- [SEC 7. Wie klassifizieren Sie Ihre Daten?](#)
- [SEC 8. Wie werden Ihre Daten im Ruhezustand geschützt?](#)
- [SEC 9. Wie werden Ihre Daten während der Übertragung geschützt?](#)

SEC 7. Wie klassifizieren Sie Ihre Daten?

Die Datenklassifizierung bietet eine Möglichkeit, Daten basierend auf Wichtigkeit und Sensibilität zu kategorisieren, um Ihnen dabei zu helfen, angemessene Schutz- und Aufbewahrungskontrollen zu bestimmen.

Best Practices

- [SEC07-BP01 Verstehen Ihres Schemas zur Datenklassifizierung](#)
- [SEC07-BP02 Anwenden von Datenschutzkontrollen basierend auf der Sensibilität der Daten](#)
- [SEC07-BP03 Automatisieren der Identifizierung und Klassifizierung](#)
- [SEC07-BP04 Definieren eines skalierbaren Datenlebenszyklusmanagements](#)

SEC07-BP01 Verstehen Ihres Schemas zur Datenklassifizierung

Machen Sie sich ein Bild von der Klassifizierung der Daten, die Ihr Workload verarbeitet, den Anforderungen an die Verarbeitung, den damit verbundenen Geschäftsprozessen, dem Ort, an dem die Daten gespeichert sind, sowie dem Eigentümer der Daten. Ihr Schema für die Klassifizierung und den Umgang mit Daten sollte die geltenden rechtlichen und Compliance-Anforderungen Ihres

Workloads und die erforderlichen Datenkontrollen berücksichtigen. Das Verständnis der Daten ist der erste Schritt zur Datenklassifizierung.

Gewünschtes Ergebnis: Die in Ihrem Workload vorhandenen Datentypen sind gut verstanden und dokumentiert. Es gibt angemessene Kontrollen zum Schutz sensibler Daten auf der Grundlage ihrer Klassifizierung. Diese Kontrollen regeln z. B., wer auf die Daten zugreifen darf und zu welchem Zweck, wo die Daten gespeichert werden, die Verschlüsselungsrichtlinie für diese Daten und wie Verschlüsselungsschlüssel verwaltet werden, den Lebenszyklus der Daten und die Anforderungen an die Aufbewahrung, angemessene Vernichtungsprozesse, welche Sicherungs- und Wiederherstellungsprozesse vorhanden sind und die Überprüfung des Zugriffs.

Typische Anti-Muster:

- Fehlen einer formalen Richtlinie zur Datenklassifizierung, um die Sensibilitätsebenen und die Anforderungen an die Handhabung von Daten zu definieren
- Mangel an Wissen über die Sensibilitätsebenen der Daten innerhalb Ihres Workloads und fehlende Erfassung dieser Informationen in der Architektur- und Betriebsdokumentation
- Versäumnis, angemessene Kontrollen für Ihre Daten anzuwenden, die auf deren Sensibilität und Anforderungen basieren, wie in Ihrer Richtlinie zur Datenklassifizierung und -verarbeitung festgelegt
- Unterlassen von Feedback über die Anforderungen an die Datenklassifizierung und -verarbeitung an die Eigentümer der Richtlinien

Vorteile der Nutzung dieser bewährten Methode: Diese Vorgehensweise beseitigt Unklarheiten über den angemessenen Umgang mit Daten innerhalb Ihres Workloads. Die Anwendung einer formellen Richtlinie, die die Sensibilitätsebenen der Daten in Ihrer Organisation und die erforderlichen Schutzmaßnahmen definiert, kann Ihnen helfen, gesetzliche Vorschriften und andere Bescheinigungen und Zertifizierungen im Bereich der Cybersicherheit einzuhalten. Besitzer von Workloads können sich darauf verlassen, dass sie wissen, wo sensible Daten gespeichert sind und welche Schutzkontrollen vorhanden sind. Wenn Sie diese in der Dokumentation festhalten, können neue Team-Mitglieder sie besser verstehen und schon früh in ihrer Amtszeit Kontrollen durchführen. Diese Praktiken können auch dazu beitragen, die Kosten zu senken, indem die Kontrollen für jede Art von Daten richtig dimensioniert werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Wenn Sie einen Workload entwerfen, überlegen Sie vielleicht intuitiv, wie Sie sensible Daten schützen können. Bei einer mandantenfähigen Anwendung ist es beispielsweise intuitiv, die Daten jedes Mandanten als sensibel zu betrachten und Schutzmaßnahmen zu ergreifen, damit ein Mandant nicht auf die Daten eines anderen Mandanten zugreifen kann. Ebenso können Sie intuitiv Zugriffskontrollen so gestalten, dass nur Administratoren Daten ändern können, während andere Benutzer nur Lesezugriff oder gar keinen Zugriff haben.

Indem Sie diese Datensensibilitätsebenen zusammen mit den entsprechenden Datenschutzerfordernissen definieren und in Richtlinien festhalten, können Sie formell feststellen, welche Daten sich in Ihrem Workload befinden. Sie können dann feststellen, ob die richtigen Kontrollen vorhanden sind, ob die Kontrollen überprüft werden können und welche Reaktionen angemessen sind, wenn ein falscher Umgang mit Daten festgestellt wird.

Um die Stellen in Ihrem Workload zu identifizieren, an denen sich sensible Daten befinden, sollten Sie die Verwendung eines Datenkatalogs in Betracht ziehen. Ein Datenkatalog ist eine Datenbank, die Daten in Ihrer Organisation, ihren Standort, ihre Vertraulichkeitsstufe und die zum Schutz dieser Daten eingeführten Kontrollen abbildet. Sie sollten außerdem die Verwendung von [Ressourcen-Tags](#) in Betracht ziehen, wenn verfügbar. Sie können zum Beispiel ein Tag mit dem Tag-Schlüssel `Classification` und dem Tag-Wert `PHI` für geschützte Gesundheitsinformationen (Protected Health Information, PHI) und ein weiteres Tag mit dem Tag-Schlüssel `Sensitivity` und dem Tag-Wert `High` verwenden. Mit Services wie [AWS Config](#) können Sie diese Ressourcen auf Änderungen überwachen und eine Warnung ausgeben, wenn sie so verändert werden, dass sie Ihren Schutzanforderungen nicht mehr genügen (z. B. durch Änderung der Verschlüsselungseinstellungen). Sie können die Standarddefinition Ihrer Tag-Schlüssel und zulässigen Werte mit [Tag-Richtlinien](#), einem Feature von AWS Organizations, erfassen. Es wird nicht empfohlen, dass der Tag-Schlüssel oder -Wert private oder sensible Daten enthält.

Implementierungsschritte

1. Verstehen Sie das Datenklassifizierungsschema und die Schutzanforderungen Ihrer Organisation.
2. Identifizieren Sie die Arten von sensiblen Daten, die von Ihren Workloads verarbeitet werden.
3. Erfassen Sie die Daten in einem Datenkatalog, der einen zentralen Überblick über die Stellen, an denen sich Daten in der Organisation befinden, und ihre Vertraulichkeit bietet.
4. Erwägen Sie die Verwendung von Markierungen auf Ressourcen- und Datenebene, sofern verfügbar, um Daten mit ihrer Sensibilitätsstufe und anderen operativen Metadaten zu versehen, die bei der Überwachung und der Reaktion auf Vorfälle helfen können.

- a. AWS Organizations-Tag-Richtlinien können verwendet werden, um Tagging-Standards durchzusetzen.

Ressourcen

Zugehörige bewährte Methoden:

- [SUS04-BP01 Implementieren einer Richtlinie für die Klassifizierung von Daten](#)

Zugehörige Dokumente:

- [Data Classification Whitepaper](#)
- [Best Practices for Tagging AWS Resources](#)

Zugehörige Beispiele:

- [AWS Organizations Tag Policy Syntax and Examples](#)

Zugehörige Tools:

- [AWS Tag-Editor](#)

SEC07-BP02 Anwenden von Datenschutzkontrollen basierend auf der Sensibilität der Daten

Wenden Sie Datenschutzkontrollen an, die ein angemessenes Maß an Kontrolle für jede in Ihrer Klassifizierungsrichtlinie definierte Datenklasse bieten. Auf diese Weise können Sie sensible Daten vor unbefugtem Zugriff und unbefugter Nutzung schützen und gleichzeitig die Verfügbarkeit und Nutzung der Daten aufrechterhalten.

Gewünschtes Ergebnis: Sie verfügen über eine Klassifizierungsrichtlinie, die die verschiedenen Sensibilitätsstufen für Daten in Ihrer Organisation definiert. Für jede dieser Sensibilitätsebenen haben Sie klare Richtlinien für zugelassene Speicher- und Bearbeitungsservices und -orte sowie deren erforderliche Konfiguration veröffentlicht. Sie implementieren die Kontrollen für jede Ebene entsprechend dem erforderlichen Schutzniveau und den damit verbundenen Kosten. Sie verfügen über Überwachungs- und Warnsysteme, um zu erkennen, wenn sich Daten an nicht autorisierten Orten befinden, in nicht autorisierten Umgebungen verarbeitet werden, nicht autorisierte Akteure darauf zugreifen oder die Konfiguration der zugehörigen Services nicht mehr konform ist.

Typische Anti-Muster:

- Anwenden des gleichen Maßes an Schutzkontrollen für alle Daten: Dies kann dazu führen, dass zu viele Sicherheitskontrollen für wenig sensible Daten bereitgestellt werden oder hochsensible Daten nicht ausreichend geschützt werden.
- Unterlassen, die relevanten Stakeholder aus Sicherheits-, Compliance- und Geschäftsteams bei der Definition von Datenschutzkontrollen einzubeziehen
- Vernachlässigen des betrieblichen Aufwands und der Kosten, die mit der Implementierung und Pflege von Datenschutzkontrollen verbunden sind
- Fehlen von regelmäßigen Überprüfungen der Datenschutzkontrollen, um die Übereinstimmung mit den Klassifizierungsrichtlinien zu gewährleisten
- Fehlen einer vollständigen Übersicht darüber, wo sich die Daten im Ruhezustand und während der Übertragung befinden.

Vorteile der Nutzung dieser bewährten Methode: Indem Sie Ihre Kontrollen auf die Klassifizierungsstufe Ihrer Daten abstimmen, kann Ihre Organisation bei Bedarf in höhere Kontrollstufen investieren. Dies kann eine Aufstockung der Ressourcen für die Sicherung, Überwachung, Messung, Behebung und Berichterstattung beinhalten. Wo weniger Kontrollen angebracht sind, können Sie die Zugänglichkeit und Vollständigkeit der Daten für Ihre Mitarbeiter, Kunden oder Wähler verbessern. Dieser Ansatz bietet Ihrer Organisation die größtmögliche Flexibilität bei der Datennutzung, während gleichzeitig die Datenschutzanforderungen eingehalten werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Die Implementierung von Datenschutzkontrollen auf der Grundlage von Datensensibilitätsebenen umfasst mehrere wichtige Schritte. Ermitteln Sie zunächst die verschiedenen Datensensibilitätsebenen innerhalb Ihrer Workload-Architektur (z. B. öffentlich, intern, vertraulich und eingeschränkt) und bewerten Sie, wo Sie diese Daten speichern und verarbeiten. Als Nächstes definieren Sie Isolationsgrenzen um die Daten herum, basierend auf ihrer Sensibilitätsebene. Wir empfehlen Ihnen, Daten in verschiedene AWS-Konten-Konten zu unterteilen und [Service-Kontrollrichtlinien \(SCPs\)](#) zu verwenden, um die für die einzelnen Sensibilitätsebenen zulässigen Services und Aktionen einzuschränken. Auf diese Weise können Sie starke Isolationsgrenzen schaffen und das Prinzip der geringsten Berechtigung durchsetzen.

Nachdem Sie die Isolationsgrenzen definiert haben, implementieren Sie geeignete Schutzkontrollen auf der Grundlage der Sensibilitätsebenen der Daten. Beachten Sie die bewährten Methoden zum [Schutz von Daten im Ruhezustand](#) und zum [Schutz von Daten während der Übertragung](#), um entsprechende Kontrollen wie Verschlüsselung, Zugriffskontrollen und Audits zu implementieren. Ziehen Sie Techniken wie Tokenisierung oder Anonymisierung in Betracht, um die Sensibilität Ihrer Daten zu verringern. Vereinfachen Sie die Anwendung konsistenter Datenrichtlinien in Ihrem Unternehmen mit einem zentralisierten System für Tokenisierung und De-Tokenisierung.

Überwachen und testen Sie fortlaufend die Wirksamkeit der implementierten Kontrollen. Überprüfen und aktualisieren Sie das Datenklassifizierungsschema, die Risikobewertungen und die Schutzkontrollen regelmäßig, wenn sich die Datenlandschaft und die Bedrohungen in Ihrer Organisation weiterentwickeln. Richten Sie die implementierten Datenschutzkontrollen an den einschlägigen Branchenvorschriften, Standards und gesetzlichen Anforderungen aus. Sorgen Sie außerdem für ein Sicherheitsbewusstsein und bieten Sie Schulungen an, damit die Mitarbeiter das Datenklassifizierungsschema und ihre Verantwortung im Umgang mit sensiblen Daten und deren Schutz verstehen.

Implementierungsschritte

1. Identifizieren Sie die Klassifizierungs- und Sensibilitätsstufen der Daten innerhalb Ihres Workloads.
2. Definieren Sie Isolationsgrenzen für jede Ebene und legen Sie eine Durchsetzungsstrategie fest.
3. Bewerten Sie die von Ihnen definierten Kontrollen, die den Zugriff, die Verschlüsselung, die Prüfung, die Aufbewahrung und andere von Ihrer Datenklassifizierungsrichtlinie geforderte Punkte regeln.
4. Prüfen Sie gegebenenfalls Optionen zur Verringerung der Sensibilität der Daten, z. B. durch Tokenisierung oder Anonymisierung.
5. Überprüfen Sie Ihre Kontrollen durch automatische Tests und die Überwachung Ihrer konfigurierten Ressourcen.

Ressourcen

Zugehörige bewährte Methoden:

- [PERF03-BP01 Verwenden eines speziell entwickelten Datenspeichers, der die Datenzugriffs- und Speicheranforderungen am besten unterstützt](#)
- [COST04-BP05 Durchsetzen von Richtlinien zur Datenaufbewahrung](#)

Zugehörige Dokumente:

- [Data Classification Whitepaper](#)
- [Bewährte Methoden für Sicherheit, Identität und Compliance](#)
- [AWS KMS Best Practices](#)
- [Encryption best practices and features for AWS services](#)

Zugehörige Beispiele:

- [Building a serverless tokenization solution to mask sensitive data](#)
- [How to use tokenization to improve data security and reduce audit scope](#)

Zugehörige Tools:

- [AWS Key Management Service \(AWS KMS\)](#)
- [AWS CloudHSM](#)
- [AWS Organizations](#)

SEC07-BP03 Automatisieren der Identifizierung und Klassifizierung

Durch die Automatisierung der Identifizierung und Klassifizierung von Daten können Sie die richtigen Kontrollen implementieren. Der Einsatz von Automatisierung als Ergänzung zur manuellen Ermittlung verringert das Risiko menschlicher Fehler und das Risiko einer Gefährdung.

Gewünschtes Ergebnis: Sie sind in der Lage zu überprüfen, ob die richtigen Kontrollen auf der Grundlage Ihrer Klassifizierungs- und Bearbeitungsrichtlinien vorhanden sind. Automatisierte Tools und Services helfen Ihnen bei der Identifizierung und Klassifizierung der Sensibilitätsebene Ihrer Daten. Die Automatisierung hilft Ihnen auch bei der kontinuierlichen Überwachung Ihrer Umgebungen, um zu erkennen und zu melden, wenn Daten auf unzulässige Weise gespeichert oder verarbeitet werden, sodass schnell Abhilfemaßnahmen ergriffen werden können.

Typische Anti-Muster:

- Vertrauen auf ausschließlich manuelle Prozesse, die fehleranfällig und zeitaufwendig sein können, um Daten zu identifizieren und zu klassifizieren. Dies kann zu einer ineffizienten und inkonsistenten Datenklassifizierung führen, insbesondere wenn das Datenvolumen wächst.

- Fehlen von Mechanismen zur Verfolgung und Verwaltung von Datenbeständen in der gesamten Organisation
- Vernachlässigen der Notwendigkeit einer kontinuierlichen Überwachung und Klassifizierung von Daten, während sie sich innerhalb der Organisation bewegen und weiterentwickeln

Vorteile der Nutzung dieser bewährten Methode: Die Automatisierung der Identifizierung und Klassifizierung von Daten kann zu einer konsistenteren und präziseren Anwendung von Datenschutzkontrollen führen und das Risiko menschlicher Fehler verringern. Die Automatisierung kann auch den Zugriff auf und die Bewegung von sensiblen Daten transparent machen, sodass Sie unautorisierten Umgang mit diesen Daten erkennen und Korrekturmaßnahmen ergreifen können.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Auch wenn die Klassifizierung von Daten in den ersten Entwurfsphasen eines Workloads häufig nach menschlichem Ermessen erfolgt, sollten Sie zur Vorbeugung Systeme einsetzen, die die Identifizierung und Klassifizierung von Testdaten automatisieren. Beispielsweise können Entwickler ein Tool oder einen Dienst erhalten, um repräsentative Daten zu scannen und ihre Sensibilität zu bestimmen. Innerhalb von AWS können Sie Datensätze in [Amazon S3](#) hochladen und sie unter Verwendung von [Amazon Macie](#), [Amazon Comprehend](#) oder [Amazon Comprehend Medical](#) scannen. Ziehen Sie auch in Betracht, Daten im Rahmen von Modultests und Integrationstests zu scannen, um festzustellen, wo sensible Daten nicht erwartet werden. Eine Warnung vor sensiblen Daten in dieser Phase kann vor der Bereitstellung in der Produktion auf Schutzlücken hinweisen. Andere Funktionen wie die Erkennung sensibler Daten in [AWS Glue](#), [Amazon SNS](#) und [Amazon CloudWatch](#) können ebenfalls verwendet werden, um PII zu erkennen und geeignete Abhilfemaßnahmen zu ergreifen. Verstehen Sie bei jedem automatisierten Tool oder Dienst, wie es sensible Daten definiert, und ergänzen Sie es mit anderen menschlichen oder automatisierten Lösungen, um eventuelle Lücken zu schließen.

Nutzen Sie die kontinuierliche Überwachung Ihrer Umgebungen als detektivische Kontrolle, um festzustellen, ob sensible Daten auf nicht konforme Weise gespeichert werden.

Dies kann dazu beitragen, Situationen zu erkennen, in denen sensible Daten ohne ordnungsgemäße De-Identifizierung oder Schwärzung in Protokolldateien ausgegeben oder in eine Datenanalyseumgebung kopiert werden. Daten, die in Amazon S3 gespeichert sind, können mit Amazon Macie kontinuierlich auf sensible Daten überwacht werden.

Implementierungsschritte

1. Überprüfen Sie das in [SEC07-BP01](#) beschriebene Datenklassifizierungsschema in Ihrem Unternehmen.
 - a. Wenn Sie das Datenklassifizierungsschema Ihres Unternehmens kennen, können Sie präzise Prozesse für die automatisierte Identifizierung und Klassifizierung einrichten, die den Richtlinien Ihres Unternehmens entsprechen.
2. Führen Sie einen ersten Scan Ihrer Umgebungen zur automatischen Identifizierung und Klassifizierung durch.
 - a. Ein erster vollständiger Scan Ihrer Daten kann dazu beitragen, ein umfassendes Verständnis darüber zu erlangen, wo sich sensible Daten in Ihren Umgebungen befinden. Wenn ein vollständiger Scan nicht erforderlich ist oder aus Kostengründen nicht im Voraus durchgeführt werden kann, sollten Sie prüfen, ob Stichprobenverfahren geeignet sind, um Ihre Ziele zu erreichen. Zum Beispiel kann Amazon Macie so konfiguriert werden, dass eine umfassende automatische Erkennung sensibler Daten in Ihren S3 Buckets durchgeführt wird. Diese Funktion nutzt Stichprobenverfahren, um kosteneffizient eine Vorabanalyse darüber durchzuführen, wo sensible Daten gespeichert sind. Eine tiefergehende Analyse von S3 Buckets kann dann mit einem Auftrag zur Erkennung sensibler Daten durchgeführt werden. Auch andere Datenspeicher können in S3 exportiert werden, um von Amazon Macie durchsucht zu werden.
 - b. Richten Sie die in [SEC07-BP02](#) definierte Zugriffskontrolle für Ihre beim Scan identifizierten Datenspeicherressourcen ein.
3. Konfigurieren Sie laufende Scans Ihrer Umgebungen.
 - a. Die automatische Erkennungsfunktion für sensible Daten von Amazon Macie kann für laufende Scans Ihrer Umgebungen verwendet werden. Bekannte S3 Buckets, die für die Speicherung sensibler Daten autorisiert sind, können mit einer Zulassen-Liste in Amazon Macie ausgeschlossen werden.
4. Integrieren Sie die Identifizierung und Klassifizierung in Ihre Build- und Testprozesse.
 - a. Identifizieren Sie Tools, mit denen Entwickler Daten auf Sensibilität prüfen können, während Workloads entwickelt werden. Verwenden Sie diese Tools als Teil der Integrationstests, um bei unerwarteten sensiblen Daten Alarm zu schlagen und eine weitere Bereitstellung zu verhindern.
5. Implementieren Sie ein System oder Runbook, um Maßnahmen zu ergreifen, wenn sensible Daten an nicht autorisierten Orten gefunden werden.
 - a. Beschränken Sie den Zugriff auf Daten mithilfe der automatischen Korrektur. Sie können diese Daten beispielsweise in einen S3-Bucket mit eingeschränktem Zugriff verschieben oder das

Objekt markieren, wenn Sie die attributbasierte Zugriffskontrolle (ABAC) verwenden. Erwägen Sie außerdem, die Daten zu maskieren, wenn sie erkannt werden.

- b. Bitten Sie die für den Datenschutz und die Reaktion auf Vorfälle zuständigen Teams, die Ursache des Vorfalls zu untersuchen. Alle gewonnenen Erkenntnisse können dazu beitragen, zukünftige Vorfälle zu verhindern.

Ressourcen

Zugehörige Dokumente:

- [AWS Glue: Detect and process sensitive data](#)
- [Using managed data identifiers in Amazon SNS](#)
- [Amazon CloudWatch Logs: Help protect sensitive log data with masking](#)

Zugehörige Beispiele:

- [Enabling data classification for Amazon RDS database with Macie](#)
- [Detecting sensitive data in DynamoDB with Macie](#)

Zugehörige Tools:

- [Amazon Macie](#)
- [Amazon Comprehend](#)
- [Amazon Comprehend Medical](#)
- [AWS Glue](#)

SEC07-BP04 Definieren eines skalierbaren Datenlebenszyklusmanagements

Machen Sie sich mit den Anforderungen an den Lebenszyklus Ihrer Daten in Bezug auf die verschiedenen Ebenen der Datenklassifizierung und -verarbeitung vertraut. Dazu kann gehören, wie Daten behandelt werden, wenn sie zum ersten Mal in Ihre Umgebung gelangen, wie Daten umgewandelt werden und welche Regeln für ihre Vernichtung gelten. Berücksichtigen Sie Faktoren wie Aufbewahrungsfristen, Zugriff, Prüfung und Nachvollziehbarkeit der Herkunft.

Gewünschtes Ergebnis: Sie klassifizieren die Daten so nah wie möglich an dem Punkt und dem Zeitpunkt der Datenerfassung. Wenn die Klassifizierung von Daten eine Maskierung, Tokenisierung

oder andere Prozesse zur Verringerung der Sensibilitätsebene erfordert, führen Sie diese Aktionen so nah wie möglich am Zeitpunkt der Datenerfassung durch.

Sie löschen Daten in Übereinstimmung mit Ihrer Richtlinie, wenn sie aufgrund ihrer Klassifizierung nicht mehr aufbewahrt werden sollten.

Typische Anti-Muster:

- Implementieren eines Einheitsansatzes für die Verwaltung des Lebenszyklus von Daten, ohne Berücksichtigung unterschiedlicher Sensibilitätsebenen und Zugriffsanforderungen
- Beschränken der Betrachtung des Lebenszyklusmanagements auf entweder nutzbare Daten oder gesicherte Daten, statt auf beide
- Annehmen, dass Daten, die in Ihren Workload eingegeben wurden, gültig sind, ohne ihren Wert oder ihre Herkunft zu ermitteln
- Vertrauen auf die Haltbarkeit von Daten als Ersatz für Datensicherungen und -schutz
- Beibehalten von Daten über ihre Nützlichkeit und die erforderliche Aufbewahrungsfrist hinaus

Vorteile der Nutzung dieser bewährten Methode: Eine gut definierte und skalierbare Strategie für die Verwaltung des Lebenszyklus von Daten hilft bei der Einhaltung gesetzlicher Vorschriften, verbessert die Datensicherheit, optimiert die Speicherkosten und ermöglicht einen effizienten Datenzugriff und -austausch unter Beibehaltung angemessener Kontrollen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Daten innerhalb eines Workloads sind oft dynamisch. Die Form, in der die Daten in Ihre Workload-Umgebung gelangen, kann sich von der Form unterscheiden, in der sie gespeichert oder in der Geschäftslogik, der Berichterstattung, der Analyse oder dem Machine Learning verwendet werden. Außerdem kann sich der Wert der Daten im Laufe der Zeit ändern. Einige Daten sind zeitlich begrenzt und verlieren an Wert, wenn sie älter werden. Überlegen Sie, wie sich diese Änderungen an Ihren Daten auf die Bewertung nach Ihrem Datenklassifizierungsschema und die damit verbundenen Kontrollen auswirken. Verwenden Sie nach Möglichkeit einen automatisierten Lebenszyklus-Mechanismus wie [Amazon S3-Lebenszyklus-Richtlinien](#) und [Amazon Data Lifecycle Manager](#), um Ihre Datenaufbewahrung, Archivierung und Ablaufprozesse zu konfigurieren. Für in DynamoDB gespeicherte Daten können Sie die [Time To Live \(TTL\)](#)-Funktion verwenden, um einen Ablaufzeitstempel pro Element zu definieren.

Unterscheiden Sie zwischen Daten, die zur Verwendung zur Verfügung stehen, und Daten, die als Backup gespeichert sind. Ziehen Sie die Verwendung von [AWS Backup](#) in Betracht, um die Sicherung von Daten über AWS-Services hinweg zu automatisieren. [Amazon EBS-Snapshots](#) bieten eine Möglichkeit, ein EBS-Volume zu kopieren und es unter Verwendung von S3-Features zu speichern, einschließlich Lebenszyklus, Datenschutz und Zugriff auf Schutzmechanismen. Zwei dieser Mechanismen sind [S3 Object Lock](#) und [AWS Backup Vault Lock](#), die Ihnen zusätzliche Sicherheit und Kontrolle über Ihre Backups bieten können. Verwalten Sie eine klare Aufgabentrennung und Zugriffsrechte für Backups. Isolieren Sie Backups auf Kontoebene, um während eines Ereignisses eine Trennung von der betroffenen Umgebung zu gewährleisten.

Ein weiterer Aspekt des Lifecycle-Managements ist die Aufzeichnung des Datenverlaufs, während diese Ihren Workload durchlaufen. Dies wird als Nachverfolgung der Datenherkunft bezeichnet. Dadurch können Sie sicher sein, dass Sie wissen, woher die Daten stammen, welche Transformationen durchgeführt wurden, welcher Eigentümer oder Prozess diese Änderungen vorgenommen hat und wann. Dieser Verlauf hilft bei der Fehlersuche und bei der Untersuchung möglicher Sicherheitsvorfälle. Sie können zum Beispiel Metadaten über Transformationen in einer [Amazon DynamoDB](#)-Tabelle protokollieren. Innerhalb eines Data Lake können Sie Kopien der transformierten Daten in verschiedenen S3-Buckets für jede Stufe der Datenpipeline aufbewahren. Speichern Sie Schema- und Zeitstempelinformationen in einem [AWS Glue Data Catalog](#).

Unabhängig von Ihrer Lösung sollten Sie die Anforderungen Ihrer Endbenutzer berücksichtigen, um die geeigneten Tools für die Berichterstattung über die Herkunft Ihrer Daten zu bestimmen. So können Sie feststellen, wie Sie Ihre Herkunft am besten verfolgen können.

Implementierungsschritte

1. Analysieren Sie die Datentypen, Sensibilitätsebenen und Zugriffsanforderungen des Workloads, um die Daten zu klassifizieren und geeignete Strategien für das Lebenszyklusmanagement zu definieren.
2. Entwerfen und implementieren Sie Richtlinien für die Datenaufbewahrung und automatisierte Vernichtungsprozesse, die mit den rechtlichen, regulatorischen und organisatorischen Anforderungen übereinstimmen.
3. Etablieren Sie Prozesse und Automatisierungen für die kontinuierliche Überwachung, Prüfung und Anpassung von Strategien, Kontrollen und Richtlinien für die Verwaltung des Datenlebenszyklus, wenn sich die Anforderungen an den Workload und die Vorschriften weiterentwickeln.
 - a. Ermitteln Sie mit [AWS Config](#) Ressourcen, für die das automatisierte Lebenszyklusmanagement nicht aktiviert ist.

Ressourcen

Zugehörige bewährte Methoden:

- [COST04-BP05 Durchsetzen von Richtlinien zur Datenaufbewahrung](#)
- [SUS04-BP03 Verwalten des Lebenszyklus von Datensätzen mithilfe von Richtlinien](#)

Zugehörige Dokumente:

- [Data Classification Whitepaper](#)
- [AWS Blueprint for Ransomware Defense](#)
- [DevOps Guidance: Improve traceability with data provenance tracking](#)

Zugehörige Beispiele:

- [How to protect sensitive data for its entire lifecycle in AWS](#)
- [Build data lineage for data lakes using AWS Glue, Amazon Neptune, and Spline](#)

Zugehörige Tools:

- [AWS Backup](#)
- [Amazon Data Lifecycle Manager](#)
- [AWS Identity and Access Management Access Analyzer](#)

SEC 8. Wie werden Ihre Daten im Ruhezustand geschützt?

Schützen Sie Ihre Daten im Ruhezustand, indem Sie mehrere Kontrollen implementieren und so das Risiko eines unbefugten Zugriffs oder einer falschen Handhabung verringern.

Best Practices

- [SEC08-BP01 Implementieren einer sicheren Schlüsselverwaltung](#)
- [SEC08-BP02 Erzwingen der Verschlüsselung im Ruhezustand](#)
- [SEC08-BP03 Automatisieren des Schutzes von Daten im Ruhezustand](#)
- [SEC08-BP04 Erzwingen der Zugriffskontrolle](#)

SEC08-BP01 Implementieren einer sicheren Schlüsselverwaltung

Eine sichere Schlüsselverwaltung umfasst die Speicherung, Rotation, Zugriffskontrolle und Überwachung von Schlüsseldaten, die zum Schutz von Daten im Ruhezustand für Ihre Workloads erforderlich sind.

Gewünschtes Ergebnis: Ein skalierbarer, wiederholbarer und automatisierter Schlüsselverwaltungsmechanismus. Der Mechanismus setzt den Zugriff mit geringster Berechtigung auf Schlüsseldaten durch und stellt das richtige Gleichgewicht zwischen Schlüsselverfügbarkeit, Vertraulichkeit und Integrität her. Sie überwachen den Zugriff auf die Schlüssel. Wenn eine Rotation von Schlüsseldaten erforderlich ist, rotieren Sie diese mithilfe eines automatisierten Prozesses. Sie lassen keinen Zugriff auf Schlüsseldaten durch menschliche Bediener zu.

Typische Anti-Muster:

- Personen haben Zugriff auf unverschlüsselte Schlüsseldaten.
- Es werden benutzerdefinierte kryptografische Algorithmen erstellt.
- Die Berechtigungen für den Zugriff auf Schlüsseldaten sind zu weit gefasst.

Vorteile der Nutzung dieser bewährten Methode: Indem Sie einen sicheren Mechanismus für die Schlüsselverwaltung für Ihre Workload einrichten, können Sie dazu beitragen, Ihre Inhalte vor unbefugtem Zugriff zu schützen. Darüber hinaus müssen möglicherweise regulatorische Anforderungen hinsichtlich der Verschlüsselung Ihrer Daten erfüllt werden. Eine effektive Schlüsselverwaltungslösung kann technische Mechanismen bereitstellen, die diesen Vorschriften zum Schutz von Schlüsseldaten entsprechen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Die Verschlüsselung von Daten im Ruhezustand ist eine Basis-Sicherheitskontrolle. Um diese Kontrolle zu implementieren, benötigt Ihr Workload einen Mechanismus, um die Schlüsseldaten sicher zu speichern und zu verwalten, die für die Verschlüsselung Ihrer Daten im Ruhezustand verwendet werden.

AWS bietet den AWS Key Management Service (AWS KMS), um eine dauerhafte, sichere und redundante Speicherung von AWS KMS-Schlüsseln bereitzustellen. [Viele AWS-Services lassen sich in AWS KMS integrieren](#), um die Verschlüsselung Ihrer Daten zu unterstützen. AWS KMS verwendet

FIPS 140-3 Level 3-validierte Hardware-Sicherheitsmodule zum Schutz Ihrer Schlüssel. Es gibt keinen Mechanismus zum Exportieren von AWS KMS-Schlüsseln als Klartext.

Bei der Bereitstellung von Workloads mit einer Multi-Konten-Strategie sollten Sie AWS KMS-Schlüssel in dem Konto aufbewahren, in dem sich auch der Workload befindet, der sie verwendet. [Bei diesem verteilten Modell](#) liegt die Verantwortung für die Verwaltung der AWS KMS-Schlüssel bei Ihrem Team. In anderen Anwendungsfällen kann sich Ihre Organisation dafür entscheiden, AWS KMS-Schlüssel in einem zentralen Konto zu speichern. Diese zentralisierte Struktur erfordert zusätzliche Richtlinien, um den kontoübergreifenden Zugriff zu ermöglichen, der benötigt wird, damit das Workload-Konto auf Schlüssel zugreifen kann, die im zentralen Konto gespeichert sind. Dieses Verfahren kann jedoch in Anwendungsfällen, in denen ein einzelner Schlüssel von mehreren AWS-Konten gemeinsam genutzt wird, besser geeignet sein.

Unabhängig davon, wo die Schlüsseldaten gespeichert werden, sollte der Zugriff auf den Schlüssel durch [Schlüsselrichtlinien](#) und IAM-Richtlinien eng kontrolliert werden. Schlüsselrichtlinien sind die primäre Methode für die Kontrolle des Zugriffs auf einen AWS KMS-Schlüssel. Darüber hinaus können AWS KMS-Schlüsselzuweisungen den Zugriff auf AWS-Services ermöglichen, um Daten in Ihrem Namen zu verschlüsseln und zu entschlüsseln. Machen Sie sich mit dem [Leitfaden für die Steuerung des Zugriffs auf Ihre AWS KMS-Schlüssel](#) vertraut.

Sie sollten die Verwendung von Verschlüsselungsschlüsseln auf ungewöhnliche Zugriffsmuster überwachen. Vorgänge, die mit von AWS verwalteten Schlüsseln und kundenseitig verwalteten Schlüsseln ausgeführt werden, die in AWS KMS gespeichert sind, können in AWS CloudTrail protokolliert werden. Sie sollten regelmäßig überprüft werden. Achten Sie besonders auf die Überwachung von Schlüsselzerstörungsereignissen. Um die versehentliche oder böswillige Zerstörung von Schlüsseldaten zu verhindern, werden Schlüsseldaten bei Schlüsselzerstörungsereignissen nicht sofort gelöscht. Versuche, Schlüssel in AWS KMS zu löschen, unterliegen einer [Wartezeit](#), die standardmäßig 30 Tage beträgt. So haben Administratoren Zeit, diese Aktionen zu überprüfen und die Anforderung rückgängig zu machen, wenn notwendig.

Die meisten AWS-Services verwenden AWS KMS auf eine Weise, die für Sie transparent ist. Sie müssen lediglich entscheiden, ob Sie einen in AWS verwalteten oder einen kundenseitig verwalteten Schlüssel verwenden möchten. Wenn Ihr Workload die direkte Verwendung von AWS KMS zum Verschlüsseln oder Entschlüsseln von Daten erfordert, sollten Sie eine [Umschlagverschlüsselung](#) verwenden, um Ihre Daten zu schützen. Das [AWS-Verschlüsselungs-SDK](#) kann clientseitige Verschlüsselungsprimitive für Ihre Anwendungen bereitstellen, um die Umschlagverschlüsselung zu implementieren und eine Integration in AWS KMS zu ermöglichen.

Implementierungsschritte

1. Ermitteln Sie die geeigneten [Schlüsselverwaltungsoptionen](#) (von AWS verwaltet oder kundenseitig verwaltet) für den Schlüssel.
 - a. Aus Gründen der Benutzerfreundlichkeit bietet AWS für die meisten Services AWS-eigene und von AWS verwaltete Schlüssel. Diese stellen eine Funktion für die Verschlüsselung von Daten im Ruhezustand bereit, ohne dass Schlüsseldaten oder -richtlinien verwaltet werden müssen.
 - b. Wenn Sie kundenseitig verwaltete Schlüssel verwenden, sollten Sie den Standard-Schlüsselspeicher in Betracht ziehen, um das beste Gleichgewicht zwischen Agilität, Sicherheit, Datenhoheit und Verfügbarkeit zu erzielen. Andere Anwendungsfälle erfordern möglicherweise die Verwendung von benutzerdefinierten Schlüsselspeichern mit [AWS CloudHSM](#) oder mit dem [externen Schlüsselspeicher](#).
2. Gehen Sie die Liste der Services durch, die Sie für Ihre Workload verwenden, um zu verstehen, wie AWS KMS in den Service integriert wird. EC2-Instances können beispielsweise verschlüsselte EBS-Volumes verwenden, um zu überprüfen, dass die von diesen Volumes erstellten Amazon-EBS-Snapshots auch mit einem kundenseitig verwalteten Schlüssel verschlüsselt werden. So wird die versehentliche Offenlegung unverschlüsselter Snapshot-Daten verhindert.
 - a. [Verwendung von AWS KMS durch AWS-Services](#)
 - b. Ausführliche Informationen zu den Verschlüsselungsoptionen, die ein AWS-Service bietet, finden Sie im Benutzerhandbuch oder im Entwicklerhandbuch für den Service unter dem Thema „Verschlüsselung im Ruhezustand“.
3. Implementieren Sie AWS KMS: AWS KMS erleichtert Ihnen das Erstellen und Verwalten von Schlüsseln sowie die Steuerung der Verschlüsselung in einer Vielzahl von AWS-Services und in Ihren Anwendungen.
 - a. [Erste Schritte: AWS Key Management Service \(AWS KMS\)](#)
 - b. Machen Sie sich mit den [bewährten Methoden für die Steuerung des Zugriffs auf Ihre AWS KMS-Schlüssel](#) vertraut.
4. Ziehen Sie das AWS Encryption SDK in Betracht: Verwenden Sie das AWS Encryption SDK mit AWS KMS-Integration, wenn Ihre Anwendung Daten clientseitig verschlüsseln muss.
 - a. [AWS Encryption SDK](#)
5. Aktivieren Sie [IAM Access Analyzer](#), um automatisch zu überprüfen und benachrichtigt zu werden, wenn zu weit gefasste AWS KMS-Schlüsselrichtlinien vorhanden sind.

- a. Ziehen Sie die Verwendung [benutzerdefinierter Richtlinienprüfungen](#) in Betracht, um sicherzustellen, dass die Aktualisierung der Ressourcenrichtlinie keinen öffentlichen Zugriff auf KMS-Schlüssel gewährt.
6. Aktivieren Sie [Security Hub CSPM](#), um Benachrichtigungen zu erhalten, wenn falsch konfigurierte Schlüsselrichtlinien, Schlüssel mit geplanter Löschung oder Schlüssel ohne aktivierte automatische Rotation vorhanden sind.
7. Ermitteln Sie die für Ihre AWS KMS-Schlüssel geeignete Protokollierungsstufe. Da Aufrufe von AWS KMS, einschließlich schreibgeschützter Ereignisse, protokolliert werden, können die CloudTrail-Protokolle für AWS KMS sehr umfangreich werden.
 - a. Einige Organisationen ziehen es vor, die AWS KMS-Protokollierungsaktivitäten in einem eigenen Pfad zu separieren. Weitere Informationen finden Sie im AWS KMS-Entwicklerhandbuch im Abschnitt [Protokollieren von AWS KMS-API-Aufrufen mit CloudTrail](#).

Ressourcen

Zugehörige Dokumente:

- [AWS Key Management Service](#)
- [AWS Kryptografische -Services und -Tools](#)
- [Schützen von Amazon-S3-Daten durch Verschlüsselung](#)
- [Umschlagverschlüsselung](#)
- [Das Versprechen zu digitaler Souveränität](#)
- [Das Geheimnis von AWS KMS-Schlüsselvorgängen, Bring Your Own Key, benutzerdefiniertem Schlüsselspeicher und Portabilität von Geheimtext](#)
- [AWS Key Management Service Kryptografische Details von](#)

Zugehörige Videos:

- [So funktioniert die Verschlüsselung in AWS](#)
- [Schützen Ihres Blockspeichers in AWS](#)
- [AWS Datenschutz in : Verwenden von Schlössern, Schlüsseln, Signaturen und Zertifikaten](#)

Zugehörige Beispiele:

- [Implementieren erweiterter Zugriffskontrollmechanismen mit AWS KMS](#)

SEC08-BP02 Erzwingen der Verschlüsselung im Ruhezustand

Verschlüsseln Sie private Daten im Ruhezustand, um die Vertraulichkeit zu wahren und eine zusätzliche Schutzebene hinsichtlich der unbeabsichtigten Offenlegung oder Exfiltration der Daten bereitzustellen. Die Verschlüsselung schützt die Daten, sodass sie ohne vorherige Entschlüsselung weder gelesen noch verwendet werden können. Inventarisieren und kontrollieren Sie unverschlüsselte Daten, um die mit einer Offenlegung der Daten verbundenen Risiken zu minimieren.

Gewünschtes Ergebnis: Sie verfügen über Mechanismen, die private Daten im Ruhezustand standardmäßig verschlüsseln. Diese Mechanismen helfen, die Vertraulichkeit der Daten zu wahren und bieten eine zusätzliche Schutzebene hinsichtlich der unbeabsichtigten Offenlegung oder Exfiltration der Daten. Sie führen ein Inventar unverschlüsselter Daten und kennen die Kontrollmechanismen, die zum Schutz dieser Daten eingerichtet wurden.

Typische Anti-Muster:

- keine Verwendung von Konfigurationen mit standardmäßiger Verschlüsselung
- Bereitstellung von Zugriffsmöglichkeiten mit zu vielen Berechtigungen für Entschlüsselungsschlüssel
- fehlende Überwachung der Ver- und Entschlüsselungsschlüssel
- Speichern von Daten ohne Verschlüsselung
- Verwendung desselben Verschlüsselungsschlüssels für alle Daten, ohne Berücksichtigung von Datennutzung, Typen und Klassifizierung

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Ordnen Sie Datenklassifizierungen in Ihren Workloads Verschlüsselungsschlüssel zu. Dieser Ansatz schützt vor Zugriff mit zu vielen Berechtigungen, wenn Sie entweder einen einzelnen Zugriffsschlüssel oder eine sehr kleine Anzahl von Verschlüsselungsschlüsseln für Ihre Daten verwenden (siehe [SEC07-BP01 Verstehen Ihres Schemas zur Datenklassifizierung](#)).

AWS Key Management Service (AWS KMS) kann in viele AWS-Services integriert werden, um die Verschlüsselung Ihrer Daten im Ruhezustand zu vereinfachen. In Amazon Elastic Compute Cloud (Amazon EC2) können Sie beispielsweise die [Standardverschlüsselung](#) von Konten

festlegen, sodass neue EBS-Volumes automatisch verschlüsselt werden. Überlegen Sie sich bei Verwendung von AWS KMS, wie stark die Daten eingeschränkt werden müssen. Standardmäßige und servicegesteuerte AWS KMS-Schlüssel werden von AWS für Sie verwaltet und verwendet. Ziehen Sie für sensible Daten, die einen differenzierten Zugriff auf den zugrunde liegenden Verschlüsselungsschlüssel erfordern, kundenseitig verwaltete Schlüssel (CMKs) in Betracht. Sie haben die vollständige Kontrolle über CMKs, einschließlich Rotation und Zugriffsverwaltung mithilfe von Schlüsselrichtlinien.

Darüber hinaus verschlüsseln Dienste wie Amazon Simple Storage Service ([Amazon S3](#)) jetzt standardmäßig alle neuen Objekte. Diese Implementierung bietet eine höhere Sicherheit, ohne die Leistung zu beeinträchtigen.

Andere Services, wie [Amazon Elastic Compute Cloud](#) (Amazon EC2) oder [Amazon Elastic File System](#) (Amazon EFS), unterstützen Einstellungen für die Standardverschlüsselung. Mit [AWS-Config-Regeln](#) können Sie automatisch überprüfen, ob Sie eine Verschlüsselung für [Volumes in Amazon Elastic Block Store \(Amazon EBS\)](#), [Instances in Amazon Relational Database Service \(Amazon RDS\)](#), [Amazon-S3-Buckets](#) und andere Services in Ihrer Organisation verwenden.

AWS bietet auch Optionen für die clientseitige Verschlüsselung, mit der Sie Daten vor dem Laden in die Cloud verschlüsseln können. Das AWS Encryption SDK ermöglicht die Verschlüsselung Ihrer Daten per [Umschlagverschlüsselung](#). Sie stellen den Wrapping-Schlüssel bereit und das AWS Encryption SDK generiert einen eindeutigen Datenschlüssel für jedes verschlüsselte Datenobjekt. Ziehen Sie die Verwendung von AWS CloudHSM in Betracht, wenn Sie ein verwaltetes Single-Tenant-Hardware-Sicherheitsmodul (HSM) benötigen. Mit AWS CloudHSM können Sie kryptographische Schlüssel auf einem nach FIPS 140-2 Level 3 validierten HSM generieren, importieren und verwalten. Einige Anwendungsfälle von AWS CloudHSM umfassen den Schutz privater Schlüssel für die Ausgabe einer Zertifizierungsstelle (Certificate Authority, CA) und die Aktivierung der transparenten Datenverschlüsselung (Transparent Data Encryption, TDE) für Oracle-Datenbanken. Das AWS CloudHSM-Client-SDK bietet Software, die die clientseitige Verschlüsselung von Daten mit innerhalb von AWS CloudHSM gespeicherten Schlüsseln ermöglicht, bevor die Daten in AWS geladen werden. Der Amazon DynamoDB Encryption Client ermöglicht darüber hinaus das Verschlüsseln und Signieren von Elementen vor dem Laden in eine DynamoDB-Tabelle.

Implementierungsschritte

- Konfigurieren der [Standardverschlüsselung für neue Amazon-EBS-Volumes](#): Geben Sie an, dass alle neu erstellten Amazon-EBS-Volumes verschlüsselt erstellt werden sollen. Dabei können Sie den von AWS bereitgestellten Standardschlüssel oder einen von Ihnen erstellten Schlüssel verwenden.

- Konfigurieren von verschlüsselten Amazon Machine Images (AMIs): Beim Kopieren eines vorhandenen AMI mit konfigurierter Verschlüsselung werden Stamm-Volumes und Snapshots automatisch verschlüsselt.
- Konfigurieren der [Amazon-RDS-Verschlüsselung](#): Konfigurieren Sie die Verschlüsselung für Ihre Amazon-RDS-Datenbank-Cluster DB-Cluster und Snapshots mithilfe der Verschlüsselungsoption.
- Erstellen und Konfigurieren von AWS KMS-Schlüsseln mit Richtlinien, die den Zugriff auf die entsprechenden Prinzipale für die jeweilige Datenklassifizierung einschränken: Erstellen Sie beispielsweise einen AWS KMS-Schlüssel für die Verschlüsselung von Produktionsdaten und einen anderen Schlüssel für die Verschlüsselung von Entwicklungs- oder Testdaten. Sie können auch anderen AWS-Konten Schlüsselzugriff gewähren. Ziehen Sie die Nutzung verschiedener Konten für Ihre Entwicklungs- und Produktionsumgebungen in Betracht. Wenn Ihre Produktionsumgebung Artefakte im Entwicklungskonto entschlüsseln muss, können Sie die zur Verschlüsselung der Entwicklungsartefakte verwendete CMK-Richtlinie so bearbeiten, dass das Produktionskonto diese Artefakte entschlüsseln kann. Die Produktionsumgebung kann dann die entschlüsselten Daten zur Verwendung in der Produktion einlesen.
- Konfigurieren der Verschlüsselung in zusätzlichen AWS-Services: Für andere von Ihnen verwendete AWS-Services können Sie in der zugehörigen [Sicherheitsdokumentation](#) die Verschlüsselungsoptionen des jeweiligen Service ermitteln.

Ressourcen

Zugehörige Dokumente:

- [AWS Crypto Tools](#)
- [AWS Encryption SDK](#)
- [AWS KMS Cryptographic Details \(Whitepaper\)](#)
- [AWS Key Management Service](#)
- [Kryptografische AWS-Services und -Tools](#)
- [Amazon-EBS-Verschlüsselung](#)
- [Standardverschlüsselung für Amazon-EBS-Volumes](#)
- [Verschlüsseln von Amazon-RDS-Ressourcen](#)
- [Wie aktiviere ich die Standardverschlüsselung für einen Amazon-S3-Bucket?](#)
- [Schützen von Amazon-S3-Daten durch Verschlüsselung](#)

Zugehörige Videos:

- [So funktioniert die Verschlüsselung in AWS](#)
- [Schützen Ihres Blockspeichers in AWS](#)

SEC08-BP03 Automatisieren des Schutzes von Daten im Ruhezustand

Nutzen Sie Automatisierung, um Daten im Ruhezustand zu validieren und zu kontrollieren. Nutzen Sie automatisierte Scans, um Fehlkonfigurationen Ihrer Datenspeicherlösungen zu erkennen, und führen Sie, wenn möglich, Abhilfemaßnahmen durch automatisierte programmatische Reaktionen durch. Integrieren Sie Automatisierung in Ihre CI/CD-Prozesse, um Fehlkonfigurationen des Datenspeichers zu erkennen, bevor sie in der Produktion bereitgestellt werden.

Gewünschtes Ergebnis: Automatisierte Systeme scannen und überwachen Datenspeicherorte auf falsch konfigurierte Steuerungen, unbefugten Zugriff und unerwartete Nutzung. Bei Erkennung falsch konfigurierter Speicherorte werden automatische Abhilfemaßnahmen initiiert. Automatisierte Prozesse erstellen Daten-Backups und speichern unveränderliche Kopien außerhalb der ursprünglichen Umgebung.

Typische Anti-Muster:

- Keine Berücksichtigung von Optionen zur Aktivierung der Verschlüsselung in den Standardeinstellungen, sofern unterstützt.
- Keine Berücksichtigung von Sicherheitsereignissen neben den betrieblichen Ereignissen bei der Formulierung einer automatisierten Backup- und Wiederherstellungsstrategie.
- Keine Erzwingung der Einstellungen für den öffentlichen Zugriff auf Speicher-Services.
- Keine Überwachung und Prüfung Ihrer Kontrollen zum Schutz von Daten im Ruhezustand.

Vorteile der Nutzung dieser bewährten Methode: Die Automatisierung trägt dazu bei, das Risiko falsch konfigurierter Datenspeicherorte zu vermeiden. Dadurch wird verhindert, dass Fehlkonfigurationen in Ihre Produktionsumgebungen gelangen. Diese bewährte Methode trägt außerdem dazu bei, gegebenenfalls vorhandene Fehlkonfigurationen zu erkennen und zu beheben.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Die Automatisierung zieht sich wie ein roter Faden durch die Praktiken zum Schutz Ihrer Daten im Ruhezustand. In [SEC01-BP06 Automatisieren der Bereitstellung von Standard-Sicherheitskontrollen](#) wird beschrieben, wie Sie die Konfiguration Ihrer Ressourcen mithilfe von IaC-Vorlagen (Infrastructure as Code) erfassen können – etwa mit [AWS CloudFormation](#). Diese Vorlagen werden in ein Versionskontrollsystem übertragen und zur Bereitstellung von Ressourcen in AWS über eine CI/CD-Pipeline verwendet. Diese Techniken gelten auch für die Automatisierung der Konfiguration Ihrer Datenspeicherlösungen (zum Beispiel für Verschlüsselungseinstellungen für Amazon-S3-Buckets).

Sie können die Einstellungen, die Sie in Ihren IaC-Vorlagen definieren, mithilfe von Regeln in [AWS CloudFormation Guard](#) auf Fehlkonfigurationen in Ihren CI/CD-Pipelines überprüfen. Mit [AWS Config](#) können Sie Einstellungen, die in CloudFormation oder anderen IaC-Tools noch nicht verfügbar sind, auf Fehlkonfigurationen überwachen. Für Fehlkonfigurationen generierte Warnungen können automatisch behandelt werden, wie in [SEC04-BP04 Initiieren von Abhilfemaßnahmen für nicht konforme Ressourcen](#) beschrieben.

Der Einsatz von Automatisierung als Teil Ihrer Strategie zur Verwaltung von Berechtigungen ist ebenfalls ein wesentlicher Bestandteil des automatisierten Datenschutzes. Unter [SEC03-BP02 Gewähren des Zugriffs mit den geringsten Berechtigungen](#) und [SEC03-BP04 Kontinuierliche Reduzierung der Berechtigungen](#) erfahren Sie, wie Sie Richtlinien für Zugriff mit den geringsten Rechten konfigurieren, die kontinuierlich von [AWS Identity and Access Management Access Analyzer](#) überwacht werden, um Erkenntnisse zu generieren, wenn die Berechtigung reduziert werden kann. Neben der Automatisierung zur Überwachung von Berechtigungen können Sie [Amazon GuardDuty](#) für die Überwachung auf anomales Datenzugriffsverhalten für Ihre [EBS-Volumes](#) (über eine EC2-Instance) sowie für Ihre [S3-Buckets](#) und für unterstützte [Amazon-Relational-Database-Service-Datenbanken](#) konfigurieren.

Automatisierung spielt auch eine Rolle bei der Erkennung, ob sensible Daten an nicht autorisierten Orten gespeichert sind. Unter [SEC07-BP03 Automatisieren der Identifizierung und Klassifizierung](#) wird beschrieben, wie [Amazon Macie](#) Ihre S3-Buckets auf unerwartete sensible Daten überwachen und Warnungen generieren kann, die eine automatisierte Reaktion initiieren können.

Orientieren Sie sich an den Vorgehensweisen unter [REL09 Sichern von Daten](#), um eine Strategie für eine automatisierte Datensicherung und -wiederherstellung zu entwickeln. Datensicherung und -wiederherstellung sind für die Wiederherstellung nach Sicherheitsereignissen ebenso wichtig wie für betriebliche Ereignisse.

Implementierungsschritte

1. Erfassen Sie die Datenspeicherkonfiguration in IaC-Vorlagen. Verwenden Sie automatische Prüfungen in Ihren CI/CD-Pipelines, um Fehlkonfigurationen zu erkennen.
 - a. Sie können [CloudFormation](#) für Ihre IaC-Vorlagen verwenden und mithilfe von [CloudFormation Guard](#) Vorlagen auf Fehlkonfigurationen überprüfen.
 - b. Verwenden Sie [AWS Config](#), um Regeln in einem proaktiven Auswertungsmodus auszuführen. Verwenden Sie diese Einstellung, um die Konformität einer Ressource vor der Erstellung als Schritt in Ihrer CI/CD-Pipeline zu prüfen.
2. Überwachen Sie Ressourcen auf Fehlkonfigurationen des Datenspeichers.
 - a. Konfigurieren Sie [AWS Config](#) so, dass Datenspeicherressourcen auf Änderungen der Kontrollkonfigurationen überwacht und Warnungen generiert werden, um Abhilfemaßnahmen aufzurufen, wenn eine Fehlkonfiguration erkannt wird.
 - b. Weitere Hinweise zu automatischen Abhilfemaßnahmen finden Sie unter [SEC04-BP04 Initiieren von Abhilfemaßnahmen für nicht konforme Ressourcen](#).
3. Überwachen und reduzieren Sie die Datenzugriffsberechtigungen kontinuierlich durch Automatisierung.
 - a. [IAM Access Analyzer](#) kann kontinuierlich ausgeführt werden, um Warnungen zu generieren, wenn gegebenenfalls eine Reduzierung der Berechtigungen möglich ist.
4. Überwachen Sie anomales Datenzugriffsverhalten und geben Sie entsprechende Warnungen aus.
 - a. [GuardDuty](#) überwacht sowohl bekannte Bedrohungssignaturen als auch Abweichungen vom Baseline-Verhalten beim Zugriff auf Datenspeicherressourcen wie EBS-Volumes, S3-Buckets und RDS-Datenbanken.
5. Überwachen Sie sensible Daten, die an unerwarteten Orten gespeichert sind, und geben Sie entsprechende Warnungen aus.
 - a. Verwenden Sie [Amazon Macie](#), um Ihre S3-Buckets kontinuierlich auf sensible Daten zu überprüfen.
6. Automatisieren Sie sichere und verschlüsselte Backups Ihrer Daten.
 - a. [AWS Backup](#) ist ein verwalteter Service, der verschlüsselte und sichere Backups verschiedener Datenquellen in AWS erstellt. Mit [Elastic Disaster Recovery](#) können Sie vollständige Server-Workloads kopieren und einen kontinuierlichen Datenschutz mit einem in Sekunden gemessenen Recovery Point Objective (RPO) gewährleisten. Sie können beide Services so konfigurieren, dass sie zusammenarbeiten, um die Erstellung von Daten-Backups und das Kopieren der Daten an Failover-Standorte zu automatisieren. Dies kann dazu beitragen, dass

Ihre Daten auch dann verfügbar bleiben, wenn sie durch betriebliche oder sicherheitsrelevante Ereignisse beeinträchtigt werden.

Ressourcen

Zugehörige bewährte Methoden:

- [SEC01-BP06 Automatisieren der Bereitstellung von Standard-Sicherheitskontrollen](#)
- [SEC03-BP02 Gewähren des Zugriffs mit den geringsten Berechtigungen](#)
- [SEC03-BP04 Kontinuierliche Reduzierung der Berechtigungen](#)
- [SEC04-BP04 Initiieren von Abhilfemaßnahmen für nicht konforme Ressourcen](#)
- [SEC07-BP03 Automatisieren der Identifizierung und Klassifizierung](#)
- [REL09-BP02 Schützen und Verschlüsseln von Backups](#)
- [REL09-BP03 Automatische Daten-Backups](#)

Zugehörige Dokumente:

- [AWS Prescriptive Guidance: Automatisches Verschlüsseln vorhandener und neuer Amazon-EBS-Volumes](#)
- [Ransomware-Risikomanagement in AWS unter Verwendung des NIST Cyber Security Framework \(CSF\)](#)

Zugehörige Beispiele:

- [Verwenden von proaktiven AWS Config-Regeln sowie von AWS CloudFormation-Hooks, um die Erstellung nicht richtlinienkonformer Cloud-Ressourcen zu verhindern](#)
- [Automatisieren und zentrales Verwalten des Datenschutzes für Amazon S3 mit AWS Backup](#)
- [AWS re:Invent 2023 – Implementieren von proaktivem Datenschutz mithilfe von Amazon-EBS-Snapshots](#)
- [AWS re:Invent 2022 – Entwickeln und Automatisieren für Ausfallsicherheit mit modernem Datenschutz](#)

Zugehörige Tools:

- [AWS CloudFormation Guard](#)

- [AWS CloudFormation Guard Rules Registry](#)
- [IAM Access Analyzer](#)
- [Amazon Macie](#)
- [AWS Backup](#)
- [Elastic Disaster Recovery](#)

SEC08-BP04 Erzwingen der Zugriffskontrolle

Um Ihre Daten im Ruhezustand zu schützen, sollten Sie Zugriffskontrollen über Mechanismen wie Isolierung und Versionsverwaltung durchsetzen. Wenden Sie das Prinzip der geringsten Berechtigung und bedingte Zugriffskontrollen an. Verhindern Sie das Erteilen von öffentlichem Zugriff auf Ihre Daten.

Gewünschtes Ergebnis: Sie stellen sicher, dass nur autorisierte Benutzer auf Daten zugreifen können, und nur insoweit sie diese für ihre Arbeit benötigen. Sie schützen Ihre Daten mit regelmäßigen Backups und Versionsverwaltung vor beabsichtigten oder unbeabsichtigten Änderungen oder Löschungen. Sie isolieren kritische Daten von anderen Daten, um ihre Vertraulichkeit und Integrität zu schützen.

Typische Anti-Muster:

- gemeinsame Speicherung von Daten mit unterschiedlichen Anforderungen hinsichtlich Vertraulichkeit oder verschiedenen Klassifizierungen
- Verwendung von übermäßigen Berechtigungen für Entschlüsselungsschlüssel
- nicht ordnungsgemäße Klassifizierung von Daten
- keine Aufbewahrung von Backups wichtiger Daten
- Gewährung von dauerhaftem Zugriff auf Produktionsdaten
- keine Prüfung des Datenzugriffs bzw. keine regelmäßige Prüfung der Berechtigungen

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Der Schutz von Daten im Ruhezustand ist wichtig, um die Integrität und Vertraulichkeit der Daten zu wahren und die Einhaltung gesetzlicher Anforderungen sicherzustellen. Um dies zu erreichen, können Sie mehrere Kontrollen implementieren, darunter Zugriffskontrolle, Isolierung, bedingter Zugriff und Versionsverwaltung.

Sie können eine Zugriffskontrolle auf der Basis der geringsten Berechtigung durchsetzen, die Benutzern und Services nur die zur Ausführung ihrer Aufgaben notwendigen Berechtigungen erteilt. Hierzu gehört auch der Zugriff auf Verschlüsselungsschlüssel. Überprüfen Sie Ihre [AWS Key Management Service \(AWS KMS\)-Richtlinien](#), um sicherzustellen, dass die von Ihnen erteilte Zugriffsebene angemessen ist und entsprechende Bedingungen gelten.

Sie können Daten auf der Grundlage verschiedener Klassifizierungsebenen trennen, indem Sie unterschiedliche AWS-Konten für jede Ebene verwenden und diese Konten mithilfe von [AWS Organizations](#) verwalten. Diese Isolierung kann helfen, einen nicht autorisierten Zugriff zu verhindern, und minimiert das Risiko einer Offenlegung von Daten.

Überprüfen Sie regelmäßig die in Amazon-S3-Bucket-Richtlinien gewährte Zugriffsebene. Vermeiden Sie die Verwendung von öffentlich lesbaren oder beschreibbaren Buckets, wenn dies nicht unbedingt erforderlich ist. Ziehen Sie die Verwendung von [AWS Config](#) in Betracht, um öffentlich verfügbare Buckets zu erkennen, und von Amazon CloudFront, um Inhalten aus Amazon S3 bereitzustellen. Vergewissern Sie sich, dass die Buckets, die keinen öffentlichen Zugriff erteilen sollen, entsprechend konfiguriert sind, um dies zu verhindern.

Implementieren Sie Mechanismen für Versionsverwaltung und Objektsperre für kritische Daten, die in Amazon S3 gespeichert sind. Die [Amazon-S3-Versionsverwaltung](#) behält frühere Versionen von Objekten bei, um nach versehentlichem Löschen oder Überschreiben Daten wiederherstellen zu können. [Amazon S3 Object Lock](#) stellt eine obligatorische Zugriffskontrolle für Objekte bereit, die das Löschen oder Überschreiben von Objekten auch durch Root-Benutzer verhindert, bis die Sperre abläuft. Darüber hinaus bietet [Amazon Glacier Vault Lock](#) ein ähnliches Feature für in Amazon Glacier gespeicherte Archive.

Implementierungsschritte

1. Durchsetzen der Zugriffskontrolle nach dem Prinzip der geringsten Berechtigung:
 - Überprüfen Sie die den Benutzern und Services erteilten Zugriffsberechtigungen und stellen Sie sicher, dass diese nur über die Berechtigungen verfügen, die sie für ihre Aufgaben benötigen.
 - Überprüfen Sie den Zugriff auf Verschlüsselungsschlüssel durch die Prüfung der [AWS Key Management Service \(AWS KMS\)-Richtlinien](#).
2. Trennen von Daten anhand verschiedener Klassifizierungsebenen:
 - Verwenden Sie unterschiedliche AWS-Konten für jede Datenklassifizierungsebene.
 - Verwalten Sie diese Konten mit [AWS Organizations](#).
3. Überprüfen der Berechtigungen für Amazon-S3-Buckets und -Objekte:

- Überprüfen Sie regelmäßig die in Amazon-S3-Bucket-Richtlinien erteilte Zugriffsebene.
 - Vermeiden Sie die Verwendung von öffentlich lesbaren oder beschreibbaren Buckets, wenn dies nicht unbedingt erforderlich ist.
 - Ziehen Sie die Verwendung von [AWS Config](#) für die Erkennung öffentlich verfügbarer Buckets in Betracht.
 - Verwenden Sie Amazon CloudFront, um Inhalte aus Amazon S3 bereitzustellen.
 - Vergewissern Sie sich, dass die Buckets, die keinen öffentlichen Zugriff gewähren sollen, entsprechend konfiguriert sind, um dies zu verhindern.
 - Sie können dasselbe Überprüfungsverfahren für Datenbanken und andere Datenquellen anwenden, die die IAM-Authentifizierung verwenden, z. B. SQS oder Datenspeicher von Drittanbietern.
4. Verwenden von AWS IAM Access Analyzer:
- Sie können [AWS IAM Access Analyzer](#) nutzen, um Amazon-S3-Buckets zu analysieren und Erkenntnisse zu generieren, wenn eine S3-Richtlinie Zugriff auf eine externe Entität gewährt.
5. Implementieren von Mechanismen für Versionsverwaltung und Objektsperre:
- Verwenden Sie die [Amazon-S3-Versionsverwaltung](#), um frühere Versionen von Objekten beizubehalten, sodass Daten nach versehentlichem Löschen oder Überschreiben wiederhergestellt werden können.
 - Verwenden Sie [Amazon S3 Object Lock](#), um eine obligatorische Zugriffskontrolle für Objekte bereitzustellen, die das Löschen oder Überschreiben von Objekten auch durch Root-Benutzer verhindert, bis die Sperre abläuft.
 - Verwenden Sie [Amazon Glacier Vault Lock](#) für Archive, die in Amazon Glacier gespeichert sind.
6. Verwenden von Amazon S3 Inventory:
- Sie können [Amazon S3 Inventory](#) verwenden, um den Replikations- und Verschlüsselungsstatus Ihrer S3-Objekte zu prüfen und zu melden.
7. Überprüfen von Berechtigungen für die Amazon-EBS und AMI-Freigabe:
- Überprüfen Sie die Freigabeberechtigungen für [Amazon EBS](#) und [AMI-Freigabe](#), um sicherzustellen, dass Ihre Images und Volumes nicht für AWS-Konten außerhalb Ihres Workloads freigegeben werden.
8. Regelmäßiges Überprüfen der Freigaben über AWS Resource Access Manager:
- Sie können [AWS Resource Access Manager](#) für die Freigabe von Ressourcen innerhalb Ihrer Amazon-VPCs verwenden, z. B. Richtlinien für AWS Network Firewall, Regeln für Amazon Route 53 Resolver und Subnetze.

- Überprüfen Sie die freigegebenen Ressourcen regelmäßig und beenden Sie die Freigabe von Ressourcen, die keine Freigabe mehr erfordern.

Ressourcen

Zugehörige bewährte Methoden:

- [SEC03-BP01 Definieren von Zugriffsanforderungen](#)
- [SEC03-BP02 Gewähren des Zugriffs mit den geringsten Berechtigungen](#)

Zugehörige Dokumente:

- [AWS KMS Cryptographic Details \(Whitepaper\)](#)
- [Einführung in die Verwaltung von Zugriffsberechtigungen für Ihre Amazon-S3-Ressourcen](#)
- [Übersicht über die Verwaltung des Zugriffs auf Ihre AWS KMS-Ressourcen](#)
- [AWS-Config-Regeln](#)
- [Amazon S3 + Amazon CloudFront: Die perfekte Kombination in der Cloud](#)
- [Verwenden der Versionsverwaltung](#)
- [Sperren von Objekten mithilfe von Amazon S3 Object Lock](#)
- [Freigeben eines Amazon-EBS-Snapshots](#)
- [Gemeinsame AMIs](#)
- [Hosten einer Single-Page-Anwendung auf Amazon S3](#)
- [AWS Globale -Bedingungsschlüssel](#)
- [Erstellen eines Datenperimeters in AWS](#)

Zugehörige Videos:

- [Schützen Ihres Blockspeichers in AWS](#)

SEC 9. Wie werden Ihre Daten während der Übertragung geschützt?

Schützen Sie Ihre Daten während der Übertragung, indem Sie mehrere Kontrollen implementieren und so das Risiko eines unbefugten Zugriffs oder eines Datenverlusts verringern.

Best Practices

- [SEC09-BP01 Implementieren einer sicheren Schlüssel- und Zertifikatverwaltung](#)
- [SEC09-BP02 Erzwingen von Verschlüsselung bei der Übertragung](#)
- [SEC09-BP03 Authentifizieren der Netzwerkkommunikation](#)

SEC09-BP01 Implementieren einer sicheren Schlüssel- und Zertifikatverwaltung

Transport Layer Security (TLS)-Zertifikate werden verwendet, um die Netzwerkkommunikation zu schützen und die Identität von Websites, Ressourcen und Workloads über das Internet sowie in privaten Netzwerken zu bestimmen.

Gewünschtes Ergebnis: Ein sicheres Zertifikatverwaltungssystem, das Zertifikate in einer Public-Key-Infrastruktur (PKI) bereitstellen, speichern und verlängern kann. Ein sicherer Schlüssel- und Zertifikatverwaltungsmechanismus verhindert die Offenlegung von Zertifikatsdaten mit privaten Schlüsseln und erneuert das Zertifikat automatisch in regelmäßigen Abständen. Er lässt sich auch in andere Services integrieren, um eine sichere Netzwerkkommunikation und Identität für Computerressourcen innerhalb Ihrer Workload zu gewährleisten. Schlüsseldaten sollten niemals für menschliche Identitäten zugänglich sein.

Typische Anti-Muster:

- Während der Bereitstellung oder Verlängerung von Zertifikaten werden manuelle Schritte ausgeführt.
- Beim Entwerfen einer privaten Zertifizierungsstelle (Certificate Authority, CA) wird die Hierarchie der Zertifizierungsstelle nicht ausreichend beachtet.
- Für öffentliche Ressourcen werden selbstsignierte Zertifikate verwendet.

Vorteile der Nutzung dieser bewährten Methode:

- Die Zertifikatverwaltung wird durch automatisierte Bereitstellung und Verlängerung vereinfacht.
- Die Verschlüsselung von Daten während der Übertragung mit TLS-Zertifikaten wird gefördert.
- Sicherheit und Überprüfbarkeit der von der Zertifizierungsstelle ausgeführten Zertifikataktionen werden gesteigert.
- Verwaltungsaufgaben werden auf verschiedenen Ebenen der CA-Hierarchie strukturiert.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Moderne Workloads nutzen verschlüsselte Netzwerkkommunikation mithilfe von PKI-Protokollen wie TLS in großem Umfang. Die Verwaltung von PKI-Zertifikaten kann komplex sein. Eine automatisierte Bereitstellung und Verlängerung von Zertifikaten kann jedoch zu einer reibungsloseren Zertifikatverwaltung beitragen.

AWS bietet zwei Services zur Verwaltung universeller PKI-Zertifikate: [AWS Certificate Manager](#) und [AWS Private Certificate Authority \(AWS Private CA\)](#). ACM ist der primäre Service, den Kunden für die Bereitstellung und Verwaltung von Zertifikaten sowohl für öffentliche als auch für private AWS-Workloads verwenden. ACM stellt mithilfe von AWS Private CA private Zertifikate aus und kann in viele andere verwaltete AWS-Services [integriert](#) werden, um sichere TLS-Zertifikate für Workloads bereitzustellen. ACM kann auch über [Amazon Trust Services](#) öffentlich vertrauenswürdige Zertifikate ausstellen. Öffentliche AVM-Zertifikate können für öffentlich zugängliche Workloads verwendet werden, da moderne Browser und Betriebssysteme diesen Zertifikaten standardmäßig vertrauen.

AWS Private CA ermöglicht es Ihnen, Ihre eigene Stamm- oder untergeordnete Zertifizierungsstelle einzurichten und TLS-Zertifikate über eine API auszustellen. Sie können diese Art von Zertifikaten in Szenarien verwenden, in denen Sie die Vertrauenskette auf der Clientseite der TLS-Verbindung steuern und verwalten. Zusätzlich zu TLS-Anwendungsfällen kann AWS Private CA für die Ausstellung von Zertifikaten für Kubernetes-Pods, Matter-Geräteproduktbescheinigungen, Codesignaturen und andere Anwendungsfälle verwendet werden, und zwar mit einer [benutzerdefinierten Vorlage](#). Sie können auch [IAM Roles Anywhere](#) verwenden, um temporäre IAM-Anmeldeinformationen für On-Premises-Workloads bereitzustellen, für die von Ihrer privaten CA signierte X.509-Zertifikate ausgestellt wurden.

Zusätzlich zu ACM und AWS Private CA bietet [AWS IoT Core](#) spezielle Unterstützung für die Bereitstellung und Verwaltung von PKI-Zertifikaten für IoT-Geräte. AWS IoT Core bietet spezielle Mechanismen für das [Onboarding von IoT-Geräten](#) in Ihre Public-Key-Infrastruktur in großem Umfang.

Einige AWS-Services, wie [Amazon API Gateway](#) und [Elastic Load Balancing](#), bieten eigene Funktionen für die Verwendung von Zertifikaten zum Schutz von Anwendungsverbindungen. Sowohl API Gateway als auch Application Load Balancer (ALB) unterstützen beispielsweise Mutual TLS (mTLS) über Client-Zertifikate, die Sie über die AWS-Managementkonsole, die CLI oder APIs erstellen und exportieren.

Überlegungen zur Einrichtung einer privaten CA-Hierarchie

Wenn Sie eine private Zertifizierungsstelle einrichten müssen, ist es wichtig, dass Sie besonders darauf achten, die CA-Hierarchie im Voraus richtig zu entwerfen. Es hat sich bewährt, beim Erstellen einer privaten CA-Hierarchie jede Ebene der Hierarchie in separaten AWS-Konten bereitzustellen. Dieser gezielte Schritt reduziert die Oberfläche für jede Ebene in der CA-Hierarchie, wodurch es einfacher wird, Anomalien in CloudTrail-Protokolldaten zu erkennen und den Umfang des Zugriffs oder die Auswirkungen eines unbefugten Zugriffs auf eines der Konten zu reduzieren. Die Stammzertifizierungsstelle sollte sich in einem eigenen separaten Konto befinden und nur zur Ausstellung eines oder mehrerer Zertifikate für eine Zwischenzertifizierungsstelle verwendet werden.

Erstellen Sie dann eine oder mehrere Zwischenzertifizierungsstellen in Konten, die vom Konto der Stammzertifizierungsstelle getrennt sind, um Zertifikate für Endbenutzer, Geräte oder andere Workloads auszustellen. Stellen Sie abschließend Zertifikate von Ihrer Stammzertifizierungsstelle an die Zwischenzertifizierungsstellen aus, die wiederum Zertifikate für die Endbenutzer oder Geräte ausstellen. Weitere Informationen zur Planung Ihrer CA-Bereitstellung und zum Entwerfen einer CA-Hierarchie, einschließlich Planung von Ausfallsicherheit, regionsübergreifender Replikation, gemeinsamer Nutzung von Zertifizierungsstellen in Ihrer Organisation und mehr, finden Sie unter [Planen Ihrer AWS Private CA-Bereitstellung](#).

Implementierungsschritte

1. Ermitteln Sie die relevanten AWS-Services, die für Ihren Anwendungsfall erforderlich sind:

- In vielen Anwendungsfällen kann die bestehende Public-Key-Infrastruktur von AWS mithilfe von [AWS Certificate Manager](#) genutzt werden. ACM kann zur Bereitstellung von TLS-Zertifikaten für Webserver, Load Balancer oder für andere Zwecke für öffentlich vertrauenswürdige Zertifikate verwendet werden.
- Ziehen Sie die Verwendung von [AWS Private CA](#) in Betracht, wenn Sie Ihre eigene private Zertifizierungsstellenhierarchie einrichten müssen oder Zugriff auf exportierbare Zertifikate benötigen. Mit ACM können dann [viele Arten von Endentitätszertifikaten](#) unter Verwendung der AWS Private CA ausgegeben werden.
- Für Anwendungsfälle, in denen in großem Umfang Zertifikate für eingebettete IoT-Geräte (Internet of Things, Internet der Dinge) bereitgestellt werden müssen, empfiehlt sich gegebenenfalls die Verwendung von [AWS IoT Core](#).
- Ziehen Sie die Verwendung nativer mTLS-Funktionen in Services wie [Amazon API Gateway](#) oder [Application Load Balancer](#) in Betracht.

2. Implementieren Sie nach Möglichkeit eine automatisierte Zertifikatverlängerung:

- Verwenden Sie die [von ACM verwaltete Verlängerung](#) für von ACM ausgestellte Zertifikate zusammen mit integrierten verwalteten AWS-Services.

3. Richten Sie Protokollierung und Audit Trails ein:

- Aktivieren Sie [CloudTrail-Protokolle](#), um Zugriffe auf die Konten zu verfolgen, die Zertifizierungsstellen enthalten. Erwägen Sie, die Integritätsprüfung der Protokolldatei in CloudTrail zu konfigurieren, um die Authentizität der Protokolldaten zu überprüfen.
- Generieren und überprüfen Sie regelmäßig [Auditberichte](#), in denen die Zertifikate aufgeführt werden, die Ihre private CA ausgestellt oder widerrufen hat. Diese Berichte können in einen S3-Bucket exportiert werden.
- Wenn Sie eine private CA bereitstellen, müssen Sie auch einen S3-Bucket einrichten, um die CRL (Certificate Revocation List, Zertifikatsperrliste) zu speichern. Anleitungen zur Konfiguration dieses S3-Buckets basierend auf den Anforderungen Ihrer Workload finden Sie unter [Planung einer Zertifikatsperrliste \(CRL\)](#).

Ressourcen

Zugehörige bewährte Methoden:

- [SEC02-BP02 Verwenden von temporären Anmeldeinformationen](#)
- [SEC08-BP01 Implementieren einer sicheren Schlüsselverwaltung](#)
- [SEC09-BP03 Authentifizieren der Netzwerkkommunikation](#)

Zugehörige Dokumente:

- [Hosten und Verwalten einer ganzen privaten Zertifikatinfrastruktur in AWS](#)
- [Sichern einer ACM Private CA-Hierarchie auf Unternehmensebene für die Automobil- und Produktionsbranche](#)
- [Bewährte Private-CA-Methoden](#)
- [So verwenden Sie AWS RAM, um Ihre ACM Private CA kontoübergreifend zu teilen](#)

Zugehörige Videos:

- [Aktivieren von AWS Certificate Manager Private CA \(Workshop\)](#)

Zugehörige Beispiele:

- [Private-CA-Workshop](#)

- [Workshop zur IoT-Geräteverwaltung](#) (einschließlich Gerätebereitstellung)

Zugehörige Tools:

- [Plugin für Kubernetes-Zertifikatmanager für die Verwendung von AWS Private CA](#)

SEC09-BP02 Erzwingen von Verschlüsselung bei der Übertragung

Erzwingen Sie Ihre definierten Verschlüsselungsanforderungen basierend auf den Richtlinien, regulatorischen Verpflichtungen und Standards Ihrer Organisation, damit Sie Ihre Organisations-, Rechts- und Compliance-Anforderungen erfüllen können. Verwenden Sie nur Protokolle mit Verschlüsselung, wenn Sie vertrauliche Daten außerhalb Ihrer Virtual Private Cloud (VPC) übertragen. Verschlüsselung trägt auch dann zur Wahrung der Datenvertraulichkeit bei, wenn die Daten nicht vertrauenswürdige Netzwerke durchqueren.

Gewünschtes Ergebnis: Sie verschlüsseln den Netzwerkdatenverkehr zwischen Ihren Ressourcen und dem Internet, um nicht autorisierten Zugriff auf die Daten zu verhindern. Sie verschlüsseln den Netzwerkverkehr in Ihrer internen AWS-Umgebung entsprechend Ihren Sicherheitsanforderungen. Sie verschlüsseln Daten während der Übertragung mit sicheren TLS-Protokollen und Cipher Suites.

Typische Anti-Muster:

- Verwendung veralteter Versionen von SSL, TLS und Komponenten von Verschlüsselungssammlungen (zum Beispiel SSL v3.0, RSA-Schlüssel mit 1024 Bit und RC4-Verschlüsselung)
- Zulassen von unverschlüsseltem (HTTP-)Datenverkehr zu oder von öffentlich zugänglichen Ressourcen
- keine Überwachung und kein Ersatz von X.509-Zertifikaten, bevor sie ablaufen
- Verwendung selbstsignierter X.509-Zertifikate für TLS

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

AWS-Services bieten HTTPS-Endpunkte, die für die Kommunikation TLS nutzen. Dadurch werden die Daten bei der Kommunikation mit den AWS-APIs während der Übertragung verschlüsselt. Unsichere HTTP-Protokolle können in einer Virtual Private Cloud (VPC) durch die Verwendung

von Sicherheitsgruppen überprüft und blockiert werden. HTTP-Anforderungen können auch [automatisch an HTTPS umgeleitet werden](#) (in Amazon CloudFront oder in einem [Application Load Balancer](#)). Sie können eine [Bucket-Richtlinie von Amazon Simple Storage Service \(Amazon S3\)](#) verwenden, um die Fähigkeit zum Hochladen von Objekten über HTTP einzuschränken und so die Verwendung von HTTPS für Objekt-Uploads in Ihre(n) Bucket(s) effektiv durchzusetzen. Sie haben uneingeschränkte Kontrolle über Ihre Datenverarbeitungsressourcen und können die Verschlüsselung während der Übertragung in allen Ihren Services implementieren. Darüber hinaus können Sie die VPN-Konnektivität mit Ihrer VPC von einem externen Netzwerk oder von [AWS Direct Connect](#) aus verwenden, um die Verschlüsselung des Datenverkehrs zu erleichtern. Vergewissern Sie sich, dass Ihre Clients bei AWS-API-Aufrufen mindestens TLS 1.2 verwenden, [da AWS die Verwendung früherer TLS-Versionen im Februar 2024 eingestellt hat](#). Wir empfehlen Ihnen, TLS 1.3 zu verwenden. Wenn Sie besondere Anforderungen an die Verschlüsselung bei der Übertragung haben, finden Sie verfügbare Lösungen von Drittanbietern im AWS Marketplace.

Implementierungsschritte

- Erzwingen der Verschlüsselung bei der Übertragung: Die definierten Verschlüsselungsanforderungen sollten sich nach den neuesten Standards und bewährten Methoden richten und nur sichere Protokolle zulassen. Konfigurieren Sie beispielsweise eine Sicherheitsgruppe, die nur das HTTPS-Protokoll für einen Application Load Balancer oder eine Amazon-EC2-Instance zulässt.
- Konfigurieren von sicheren Protokollen in Edge-Services: [Konfigurieren Sie HTTPS mit Amazon CloudFront](#) und verwenden Sie ein [für Ihren Sicherheitsstatus und Ihren Anwendungsfall geeignetes Sicherheitsprofil](#).
- Verwenden eines [VPN für externe Konnektivität](#): Verwenden Sie gegebenenfalls ein IPsec-VPN, um Punkt-zu-Punkt- oder Netzwerk-zu-Netzwerk-Verbindungen zu schützen und so Datenschutz und Datenintegrität zu gewährleisten.
- Konfigurieren von sicheren Protokollen bei Load Balancern: Wählen Sie eine Sicherheitsrichtlinie aus, die die stärksten Verschlüsselungssammlungen bereitstellt, die von den Clients unterstützt werden, die eine Verbindung mit dem Listener herstellen. [Erstellen Sie einen HTTPS-Listener für Ihren Application Load Balancer](#).
- Konfigurieren von sicheren Protokollen bei Amazon Redshift: Konfigurieren Sie Ihren Cluster so, dass eine [Verbindung über Secure Socket Layer \(SSL\) oder Transport Layer Security \(TLS\)](#) verwendet werden muss.
- Konfigurieren von sicheren Protokollen: Sehen Sie sich die AWS-Servicedokumentation an, um die Funktionen zur Verschlüsselung während der Übertragung zu bestimmen.

- Konfigurieren von sicherem Zugriff beim Hochladen in Amazon-S3-Buckets: Verwenden Sie die Richtliniensteuerung für Amazon-S3-Buckets, um [sicheren Zugriff auf Daten zu erzwingen](#).
- Erwägen der Verwendung von [AWS Certificate Manager](#): ACM ermöglicht die Bereitstellung und Verwaltung öffentlicher TLS-Zertifikate für die Verwendung mit AWS-Services.
- Erwägen der Verwendung von [AWS Private Certificate Authority](#) für private PKI-Anforderungen: AWS Private CA ermöglicht die Erstellung privater Zertifizierungsstellenhierarchien, um X.509-Endentitätszertifikate auszustellen, die zum Erstellen verschlüsselter TLS-Kanäle verwendet werden können.

Ressourcen

Zugehörige Dokumente:

- [Verwenden von HTTPS mit CloudFront](#)
- [Verbinden Ihrer VPC mit Remote-Netzwerken über AWS Virtual Private Network](#)
- [Erstellen eines HTTPS-Listeners für Ihren Application Load Balancer](#)
- [Tutorial: SSL/TLS unter Amazon Linux 2 konfigurieren](#)
- [Verwenden von SSL/TLS für die Verschlüsselung einer Verbindung zu einer DB-Instance](#)
- [Konfigurieren von Sicherheitsoptionen für Verbindungen](#)

SEC09-BP03 Authentifizieren der Netzwerkkommunikation

Überprüfen Sie die Kommunikationsidentität mithilfe von Protokollen mit Authentifizierungsunterstützung – beispielsweise Transport Layer Security (TLS) oder IPsec.

Gestalten Sie Ihre Workload so, dass bei der Kommunikation zwischen Services, Anwendungen oder Benutzern sichere, authentifizierte Netzwerkprotokolle verwendet werden. Die Verwendung von Netzwerkprotokollen, die Authentifizierung und Autorisierung unterstützen, ermöglicht eine bessere Kontrolle des Netzwerkflusses und reduziert die Auswirkungen von nicht autorisiertem Zugriff.

Gewünschtes Ergebnis: Eine Workload mit klar definierten Datenflüssen auf Daten- und Steuerebene zwischen Services. Die Datenflüsse verwenden authentifizierte und verschlüsselte Netzwerkprotokolle, sofern dies technisch möglich ist.

Typische Anti-Muster:

- unverschlüsselte oder unauthentifizierte Datenflüsse innerhalb Ihrer Workload

- Wiederverwendung von Authentifizierungsdaten für mehrere Benutzer oder Entitäten
- alleinige Verwendung von Netzwerkkontrollen als Zugriffskontrolle
- Erstellung eines benutzerdefinierten Authentifizierungsmechanismus, anstatt sich auf branchenübliche Standard-Authentifizierungsmechanismen zu verlassen
- Übermäßig freizügige Datenflüsse zwischen Service-Komponenten oder anderen Ressourcen in der VPC

Vorteile der Nutzung dieser bewährten Methode:

- Schränkt den Umfang der Auswirkungen eines unberechtigten Zugriffs auf einen Teil des Workloads ein.
- Bietet ein höheres Maß an Sicherheit, dass Aktionen nur von authentifizierten Personen durchgeführt werden können.
- Verbessert die Entkopplung von Services, indem die vorgesehenen Schnittstellen für die Datenübertragung klar definiert und erzwungen werden.
- Verbessert die Überwachung und Protokollierung sowie die Reaktion auf Vorfälle durch Zuordnung von Anforderungen sowie durch klar definierte Kommunikationsschnittstellen.
- Bietet durch die Kombination von Netzwerkkontrollen mit Authentifizierungs- und Autorisierungskontrollen einen umfassenden Schutz für Ihre Workloads.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Niedrig

Implementierungsleitfaden

Die Netzwerkdatenverkehrsmuster Ihrer Workload lassen sich in zwei Kategorien einteilen:

- Der Ost-West-Verkehr steht für Datenflüsse zwischen Services, die eine Workload ausmachen.
- Der Nord-Süd-Verkehr stellt die Datenflüsse zwischen Ihrer Workload und den Consumern dar.

Während es üblich ist, den Nord-Süd-Verkehr zu verschlüsseln, ist der Schutz des Ost-West-Verkehrs mit authentifizierten Protokollen weniger verbreitet. In modernen Sicherheitspraktiken wird darauf hingewiesen, dass das Netzwerkdesign allein noch keine vertrauenswürdige Beziehung zwischen zwei Entitäten gewährleistet. Auch wenn sich zwei Services innerhalb einer gemeinsamen Netzwerkgrenze befinden, ist es immer noch die beste Methode, die Kommunikation zwischen diesen Services zu verschlüsseln, zu authentifizieren und zu autorisieren.

Beispielsweise verwenden AWS-Service-APIs das Signaturprotokoll [AWS Signature Version 4 \(SigV4\)](#), um den Anforderer zu authentifizieren, unabhängig davon, aus welchem Netzwerk die Anforderung stammt. Diese Authentifizierung stellt sicher, dass AWS-APIs die Identität des Anforderers der Aktion überprüfen können. Diese Identität kann dann mit Richtlinien kombiniert werden, um eine Autorisierungsentscheidung zu treffen und zu bestimmen, ob die Aktion zugelassen werden soll.

Mit Services wie [Amazon VPC Lattice](#) und [Amazon API Gateway](#) können Sie das gleiche SigV4-Signaturprotokoll verwenden, um den Ost-West-Verkehr in Ihren eigenen Workloads zu authentifizieren und zu autorisieren. Wenn Ressourcen außerhalb Ihrer AWS-Umgebung mit Services kommunizieren müssen, die eine SigV4-basierte Authentifizierung und Autorisierung erfordern, können Sie [AWS Identity and Access Management \(IAM\) Roles Anywhere](#) für die AWS-fremde Ressource verwenden, um temporäre AWS-Anmeldeinformationen zu erhalten. Diese Anmeldeinformationen können verwendet werden, um Anforderungen für Services zu signieren, die Zugriff mithilfe von SigV4 autorisieren.

Ein weiterer gängiger Mechanismus zur Authentifizierung des Ost-West-Verkehrs ist die gegenseitige TLS-Authentifizierung (mTLS). Viele IoT-Anwendungen (Internet of Things, Internet der Dinge) und Business-to-Business-Anwendungen sowie Microservices verwenden mTLS, um die Identität beider Seiten einer TLS-Kommunikation durch die Verwendung von X.509-Zertifikaten auf Client- und Server-Seite zu validieren. Diese Zertifikate können von AWS Private Certificate Authority (AWS Private CA) ausgestellt werden. Sie können Services wie [Amazon API Gateway](#) verwenden, um die mTLS-Authentifizierung für die Kommunikation zwischen Workloads oder innerhalb eines Workloads bereitzustellen. [Application Load Balancer unterstützt auch mTLS](#) für interne oder externe Workloads. mTLS stellt zwar Authentifizierungsinformationen für beide Seiten einer TLS-Kommunikation bereit, bietet aber keinen Mechanismus zur Autorisierung.

Und zu guter Letzt: OAuth 2.0 und OpenID Connect (OIDC) sind zwei Protokolle, die in der Regel für die Steuerung des Zugriffs von Benutzern auf Services verwendet werden. Inzwischen werden sie jedoch auch für Datenverkehr zwischen Services immer beliebter. API Gateway bietet einen [JSON Web Token \(JWT\) Authorizer](#), der es Workloads ermöglicht, den Zugriff auf API-Routen mithilfe von JWTs zu beschränken, die von OIDC- oder OAuth-2.0-Identitätsanbietern ausgestellt wurden. OAuth2-Bereiche können als Quelle für grundlegende Autorisierungsentscheidungen verwendet werden, aber die Autorisierungsprüfungen müssen immer noch in der Anwendungsschicht implementiert werden. Und OAuth2-Bereiche allein können komplexere Autorisierungsanforderungen nicht unterstützen.

Implementierungsschritte

- Definieren und Dokumentieren der Netzwerkflüsse Ihrer Workload: Der erste Schritt bei der Implementierung einer umfassenden Verteidigungsstrategie ist die Definition der Datenflüsse Ihrer Workload.
 - Erstellen Sie ein Datenflussdiagramm, das klar definiert, wie Daten zwischen den verschiedenen Services, aus denen sich Ihre Workload zusammensetzt, übertragen werden. Dieses Diagramm ist der erste Schritt zur Erzwingung dieser Datenflüsse über authentifizierte Netzwerkkanäle.
 - Nutzen Sie Ihre Workloads in der Entwicklungs- und Testphase, um zu überprüfen, ob das Datenflussdiagramm das Verhalten der Workloads zur Laufzeit korrekt wiedergibt.
 - Ein Datenflussdiagramm kann auch bei der Durchführung einer Bedrohungsmodellierung nützlich sein, wie unter [SEC01-BP07 Identifizieren von Bedrohungen und Priorisieren von Abhilfemaßnahmen unter Verwendung eines Bedrohungsmodells](#) beschrieben.
- Einrichten von Netzwerkkontrollen: Erwägen Sie die Verwendung von AWS-Funktionen, um Netzwerkkontrollen einzurichten, die auf Ihre Datenflüsse abgestimmt sind. Netzwerkgrenzen sollten zwar nicht die einzige Sicherheitskontrolle sein, aber sie stellen eine Schicht der umfassenden Verteidigungsstrategie zum Schutz Ihrer Workload dar.
 - Verwenden Sie [Sicherheitsgruppen](#), um den Datenfluss zwischen Ressourcen zu definieren und einzuschränken.
 - Erwägen Sie die Verwendung von [AWS PrivateLink](#), um sowohl mit AWS als auch mit Drittanbieter-Services zu kommunizieren, die AWS PrivateLink unterstützen. Daten, die über einen AWS PrivateLink-Schnittstellen-Endpunkt gesendet werden, bleiben innerhalb des AWS-Netzwerk-Backbones und durchlaufen nicht das öffentliche Internet.
- Implementieren von Authentifizierung und Autorisierung für alle Services in Ihrer Workload: Wählen Sie die AWS-Services aus, die am besten geeignet sind, um authentifizierte, verschlüsselte Datenflüsse in Ihrer Workload bereitzustellen.
 - Ziehen Sie die Verwendung von [Amazon VPC Lattice](#) in Betracht, um die Kommunikation zwischen Services zu schützen. VPC Lattice kann [SigV4-Authentifizierung in Kombination mit Authentifizierungsrichtlinien](#) verwenden, um den Zugriff zwischen Services zu steuern.
 - Ziehen Sie für die Kommunikation zwischen Services mit mTLS [API Gateway Application Load Balancer](#) in Betracht. [AWS Private CA](#) kann verwendet werden, um eine private CA-Hierarchie einzurichten, die Zertifikate für die Verwendung mit mTLS ausstellen kann.
 - Im Falle einer Integration in Services, die OAuth 2.0 oder OIDC verwenden, sollten Sie die Verwendung von [API Gateway mit dem JWT-Generierer](#) in Betracht ziehen.

- Für die Kommunikation zwischen Ihrer Workload und IoT-Geräten sollten Sie die Verwendung von [AWS IoT Core](#) in Betracht ziehen. Dadurch stehen Ihnen mehrere Optionen für die Verschlüsselung und Authentifizierung des Netzwerkverkehrs zur Verfügung.
- Überwachung auf nicht autorisierten Zugriff: Überwachen Sie kontinuierlich unbeabsichtigte Kommunikationskanäle, nicht autorisierte Prinzipale, die versuchen, auf geschützte Ressourcen zuzugreifen, und andere unzulässige Zugriffsmuster.
- Wenn Sie VPC Lattice zur Verwaltung des Zugriffs auf Ihre Services verwenden, empfiehlt es sich gegebenenfalls, die [Zugriffsprotokolle von VPC Lattice](#) zu aktivieren und zu überwachen. Diese Zugriffsprotokolle enthalten Informationen über die anfordernde Entität, Netzwerkinformationen einschließlich Quell- und Ziel-VPC sowie Metadaten der Anforderung.
- Erwägen Sie die Aktivierung von [VPC-Flow-Protokollen](#), um Metadaten zu Netzwerkflüssen zu erfassen und regelmäßig auf Anomalien zu überprüfen.
- Weitere Hinweise zum Planen, Simulieren und Reagieren auf Sicherheitsvorfälle finden Sie im [AWS Security Incident Response Guide](#) und im Abschnitt [Vorfallreaktion](#) der Säule „Sicherheit“ des AWS-Well-Architected-Framework.

Ressourcen

Zugehörige bewährte Methoden:

- [SEC03-BP07 Analysieren des öffentlichen und kontoübergreifenden Zugriffs](#)
- [SEC02-BP02 Verwenden von temporären Anmeldeinformationen](#)
- [SEC01-BP07 Identifizieren von Bedrohungen und Priorisieren von Abhilfemaßnahmen unter Verwendung eines Bedrohungsmodells](#)

Zugehörige Dokumente:

- [Auswerten von Zugriffskontrollmethoden zum Schutz von Amazon-API-Gateway-APIs](#)
- [Konfigurieren der gegenseitigen TLS-Authentifizierung für eine REST-API](#)
- [Schützen von API-Gateway-HTTP-Endpunkten mit JWT-Genehmiger](#)
- [Autorisieren von direkten Aufrufen von AWS-Services mithilfe des AWS IoT Core-Anmeldeinformationsanbieters](#)
- [Leitfaden zur Reaktion auf Sicherheitsvorfälle in AWS](#)

Zugehörige Videos:

- [AWS re:invent 2022 – Vorstellung von VPC Lattice](#)
- [AWS re:invent 2020 – Serverlose API-Authentifizierung für HTTP-APIs in AWS](#)

Zugehörige Beispiele:

- [Workshop zu Amazon VPC Lattice](#)
- [Zero-Trust, Episode 1: Der Phantom-Service-Perimeter-Workshop](#)

Vorfallreaktion

Frage

- [SEC 10. Wie können Sie Vorfälle voraussagen, darauf reagieren und diese beheben?](#)

SEC 10. Wie können Sie Vorfälle voraussagen, darauf reagieren und diese beheben?

Auch bei ausgereiften präventiven und Erkennungskontrollen, sollte Ihr Unternehmen Verfahren etablieren, um auf Sicherheitsvorfälle reagieren und mögliche Auswirkungen mindern zu können. Ihre Vorbereitung wirkt sich stark auf die Fähigkeit Ihrer Teams aus, während eines Vorfalls effektiv zu arbeiten, Probleme zu isolieren, einzudämmen und forensisch zu untersuchen sowie den Betrieb in einem bekannten guten Zustand wiederherzustellen. Durch die Bereitstellung von Tools und Zugriff vor einem Sicherheitsvorfall und die routinemäßige Reaktion auf Vorfälle im Alltag können Sie sicherstellen, dass Sie eine Wiederherstellung durchführen und die Betriebsunterbrechung minimieren können.

Best Practices

- [SEC10-BP01 Identifizieren wichtiger Mitarbeiter und externer Ressourcen](#)
- [SEC10-BP02 Entwickeln von Vorfallmanagementplänen](#)
- [SEC10-BP03 Vorbereiten forensischer Funktionen](#)
- [SEC10-BP04 Entwickeln und Testen von Playbooks für die Reaktion auf Sicherheitsvorfälle](#)
- [SEC10-BP05 Vorab bereitgestellter Zugriff](#)
- [SEC10-BP06 Vorabbereitstellen von Tools](#)
- [SEC10-BP07 Durchführen von Simulationen](#)

- [SEC10-BP08 Entwickeln eines Frameworks, um aus Vorfällen zu lernen](#)

SEC10-BP01 Identifizieren wichtiger Mitarbeiter und externer Ressourcen

Identifizieren Sie internes und externes Personal, Ressourcen und rechtliche Anforderungen, um Ihre Organisation bei der Reaktion auf einen Vorfall zu unterstützen.

Gewünschtes Ergebnis: Sie verfügen über eine Liste mit den wichtigsten Mitarbeitern, ihren Kontaktinformationen und ihrer Rolle bei der Reaktion auf ein Sicherheitsereignis. Sie überprüfen diese Informationen regelmäßig und aktualisieren sie, um personelle Veränderungen aus Sicht der internen und externen Tools zu berücksichtigen. Bei der Dokumentation dieser Informationen berücksichtigen Sie alle Drittanbieter und Dienstleister, einschließlich Sicherheitspartnern, Cloud-Anbietern und Software as a Service (SaaS)-Anwendungen. Während eines Sicherheitsereignisses stehen Mitarbeiter mit dem entsprechenden Maß an Verantwortung, Kontext und Zugriff zur Verfügung, um zu reagieren und das Ereignis zu bewältigen.

Typische Anti-Muster:

- Fehlen einer aktualisierten Liste der wichtigsten Mitarbeiter mit Kontaktinformationen, ihren Aufgaben und ihren Verantwortlichkeiten bei der Reaktion auf Sicherheitsvorfälle
- Voraussetzen, dass jeder die Personen, die Abhängigkeiten, die Infrastruktur und die Lösungen bei der Reaktion auf ein Ereignis und bei der Bewältigung eines Ereignisses versteht
- Fehlen eines Dokuments oder eines Wissens-Repositorys, das die wichtigsten Infrastruktur- oder Anwendungsdesigns darstellt
- Fehlen von angemessenen Einarbeitungsprozessen für neue Mitarbeiter, um effektiv zur Reaktion auf ein Sicherheitsereignis beizutragen (etwa die Durchführung von Ereignissimulationen)
- Fehlen eines Eskalationspfads für den Fall, dass wichtige Mitarbeiter vorübergehend nicht verfügbar sind oder bei Sicherheitsereignissen nicht reagieren

Vorteile der Nutzung dieser bewährten Methode Diese Praxis reduziert die Triage- und Reaktionszeit, die für die Identifizierung der richtigen Mitarbeiter und ihrer Rollen während eines Ereignisses aufgewendet wird. Minimieren Sie Zeitverluste während eines Ereignisses, indem Sie eine aktualisierte Liste der wichtigsten Mitarbeiter und ihrer Rollen führen, damit Sie die richtigen Personen für die Triage und die Bewältigung eines Ereignisses einsetzen können.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Identifizieren Sie wichtige Personen in Ihrer Organisation: Führen Sie eine Kontaktliste der Personen in Ihrer Organisation, die Sie einbeziehen müssen. Überprüfen und aktualisieren Sie diese Informationen regelmäßig bei personellen Veränderungen wie organisatorischen Änderungen, Beförderungen und Teamwechseln. Dies ist besonders wichtig für Schlüsselpositionen wie Incident Manager, Incident Responder und Communications Lead.

- **Incident Manager:** Incident Managers haben die Gesamtverantwortung für die Reaktion auf das Ereignis.
- **Incident Responder:** Incident Responders sind für Untersuchungen und Abhilfemaßnahmen zuständig. Diese Personen können sich je nach Art des Ereignisses unterscheiden, sind aber in der Regel Entwickler und Betriebs-Teams, die für die betroffene Anwendung verantwortlich sind.
- **Communications Lead:** Communications Leads sind für die interne und externe Kommunikation verantwortlich, insbesondere mit Behörden, Regulierungsbehörden und Kunden.
- **Onboarding-Prozess:** Führen Sie regelmäßige Schulungen und ein Onboarding für neue Mitarbeiter durch, um ihnen die notwendigen Fähigkeiten und Kenntnisse zu vermitteln, damit sie effektiv bei der Reaktion auf Vorfälle mitwirken können. Integrieren Sie Simulationen und praktische Übungen in den Onboarding-Prozess, damit sie besser vorbereitet sind.
- **Fachexperten (Subject Matter Experts, SMEs):** Im Falle von verteilten und autonomen Teams empfehlen wir Ihnen, für geschäftskritische Workloads SMEs zu bestimmen. Sie bieten Einblicke in den Betrieb und die Datenklassifizierung von kritischen Workloads, die an dem Ereignis beteiligt sind.

Beispieltabellenformat:

	Role	Name	Contact Information	Responsibilities
1	---	---	---	---
2	Incident Manager	Jane Doe	jane.doe@example.com	Overall authority during response
3	Incident Responder	John Smith	john.smith@example.com	Investigation and remediation
4	Communications Lead	Emily Johnson	emily.johnson@example.com	Internal and external communications
5	Communications Lead	Michael Brown	michael.brown@example.com	Insights on critical workloads

Erwägen Sie die Verwendung des Features [AWS Systems Manager Incident Manager](#), um wichtige Kontakte zu erfassen, einen Reaktionsplan zu definieren, Bereitschaftspläne zu automatisieren und Eskalationspläne zu erstellen. Automatisieren und rotieren Sie alle Mitarbeiter durch einen Bereitschaftsdienstplan, sodass die Verantwortung für die Workload auf alle zuständigen Personen verteilt wird. Dies fördert gute Praktiken wie die Ausgabe relevanter Metriken und Protokolle sowie die Definition von Alarmschwellen, die für die Workload von Bedeutung sind.

Identifizieren Sie externe Partner: Unternehmen nutzen Tools, die von unabhängigen Softwareanbietern (ISVs), Partnern und Subunternehmern entwickelt wurden, um differenzierte Lösungen für ihre Kunden zu erstellen. Binden Sie wichtige Mitarbeiter dieser Parteien ein, die Ihnen bei der Reaktion auf einen Vorfall und bei dessen Bewältigung helfen können. Wir empfehlen Ihnen, sich für die entsprechende Stufe von Support anzumelden, um über einen Supportfall sofortigen Zugang zu AWS -Fachexperten zu erhalten. Erwägen Sie, ähnliche Vereinbarungen mit allen Anbietern kritischer Lösungen für die Workloads zu schließen. Einige Sicherheitsereignisse machen es erforderlich, dass börsennotierte Unternehmen die zuständigen Behörden und Aufsichtsbehörden über das Ereignis und dessen Auswirkungen informieren. Pflegen und aktualisieren Sie die Kontaktinformationen der relevanten Abteilungen und der zuständigen Personen.

Implementierungsschritte

1. Richten Sie eine Lösung für das Vorfallmanagement ein.
 - a. Erwägen Sie die Bereitstellung von Incident Manager in Ihrem Security-Tooling-Konto.
2. Definieren Sie Kontakte in Ihrer Lösung für das Vorfallmanagement.
 - a. Definieren Sie für jeden Kontakt mindestens zwei Arten von Kontaktkanälen (z. B. SMS, Telefon oder E-Mail), um die Erreichbarkeit während eines Vorfalls sicherzustellen.
3. Definieren Sie einen Reaktionsplan.
 - a. Ermitteln Sie die am besten geeigneten Ansprechpartner für einen Vorfall. Definieren Sie Eskalationspläne, die sich an den Rollen der einzuschaltenden Mitarbeiter orientieren (und nicht an einzelnen Ansprechpartnern). Erwägen Sie die Aufnahme von Kontakten, die gegebenenfalls für die Benachrichtigung externer Stellen zuständig sind, auch wenn diese nicht direkt an der Lösung des Vorfalls beteiligt sind.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS02-BP03 Betriebsaktivitäten haben feste Besitzer, die für ihre Leistung verantwortlich sind](#)

Zugehörige Dokumente:

- [AWS Security Incident Response Guide](#)

Zugehörige Beispiele:

- [AWS Customer Playbook Framework](#)
- [Vorbereiten und Reagieren auf Sicherheitsvorfälle in Ihrer AWS-Umgebung](#)

Zugehörige Tools:

- [AWS Systems Manager Incident Manager](#)

Zugehörige Videos:

- [Der Sicherheitsansatz von Amazon bei der Entwicklung](#)

SEC10-BP02 Entwickeln von Vorfallmanagementplänen

Das erste Dokument, das für die Vorfallreaktion entwickelt werden muss, ist der Vorfallreaktionsplan. Der Vorfallreaktionsplan ist als Grundlage für Ihr Vorfallreaktionsprogramm und Ihre Vorfallreaktionsstrategie konzipiert.

Vorteile der Nutzung dieser bewährten Methode: Die Entwicklung durchdachter und klar definierter Prozesse zur Vorfallreaktion ist der Schlüssel zu einem erfolgreichen und skalierbaren Vorfallreaktionsprogramm. Wenn ein Sicherheitsereignis eintritt, können Ihnen klare Schritte und Workflows dabei helfen, rechtzeitig zu reagieren. Möglicherweise verfügen Sie bereits über Prozesse zur Vorfallreaktion. Unabhängig von Ihrem aktuellen Status ist es wichtig, Ihre Prozesse zur Vorfallreaktion regelmäßig zu aktualisieren, zu wiederholen und zu testen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Ein Vorfallmanagementplan ist von entscheidender Bedeutung, um auf Sicherheitsvorfälle zu reagieren, sie einzudämmen und ihre potenziellen Folgen zu beheben. Ein Vorfallmanagementplan ist ein strukturierter Prozess für die Identifizierung und Behebung von Sicherheitsvorfällen sowie die zeitnahe Reaktion darauf.

In der Cloud gibt es viele der betrieblichen Rollen und Anforderungen, die auch für eine On-Premises-Umgebung typisch sind. Beim Erstellen eines Plans für das Vorfalldmanagement ist es wichtig, Reaktions- und Wiederherstellungsstrategien zu entwickeln, die Ihren Anforderungen an geschäftliche Ergebnisse und Compliance optimal entsprechen. Wenn Sie beispielsweise Workloads in AWS ausführen, die in den USA FedRAMP-konform sind, sollten Sie die Empfehlungen im [NIST SP 800-61 Computer Security Handling Guide](#) befolgen. Ähnlich gilt bei der Ausführung von Workloads, die persönlich identifizierbare Informationen (PII) speichern, dass Sie diese schützen und auf Probleme im Zusammenhang mit Datenresidenz und Verwendung von Daten reagieren müssen.

Wenn Sie einen Vorfalldmanagementplan für Ihre Workloads in AWS erstellen, beginnen Sie mit dem [AWS-Modell der geteilten Verantwortung](#) zum Aufbau eines durchdachten Verteidigungskonzepts für die Vorfalldreaktion. In diesem Modell kümmert sich AWS um die Sicherheit der Cloud und Sie sind für die Sicherheit in der Cloud verantwortlich. Das bedeutet, dass Sie die Kontrolle behalten und für die Sicherheitskontrollen verantwortlich sind, für deren Implementierung Sie sich entscheiden. Der [Leitfaden für AWS Security Incident Response](#) enthält zentrale Konzepte und grundlegende Anleitungen für den Aufbau eines Cloud-basierten Vorfalldmanagementplans.

Ein effektiver Vorfalldmanagementplan muss kontinuierlich durchlaufen und stets an die Ziele Ihrer Cloud-Operationen angepasst werden. Erwägen Sie die Verwendung der nachfolgend erläuterten Implementierungspläne für die Erstellung und Weiterentwicklung Ihres Vorfalldmanagementplans.

Implementierungsschritte

1. Definieren Sie Rollen und Verantwortlichkeiten innerhalb Ihrer Organisation für den Umgang mit Sicherheitsereignissen. Daran sollten Vertreter verschiedener Bereiche beteiligt sein, darunter:
 - Personalabteilung (HR)
 - Führungsteam
 - Rechtsabteilung
 - Besitzer und Entwickler von Anwendungen (fachliche Experten)
2. Beschreiben Sie klar, welche Personen bei einem Vorfall verantwortlich sind, Rechenschaft geben müssen, konsultiert werden müssen und informiert werden müssen (Responsible, Accountable, Consulted, Informed, RACI). Erstellen Sie ein RACI-Diagramm, um eine schnelle und direkte Kommunikation zu unterstützen, und beschreiben Sie klar die Personen, die während der verschiedenen Phasen eines Ereignisses die Leitung haben.
3. Binden Sie während eines Vorfalls Anwendungsbesitzer und Entwickler (fachliche Experten) ein, da sie wertvolle Informationen und Kontext bereitstellen können, um die Auswirkungen messen

zu können. Entwickeln Sie Beziehungen zu diesen fachlichen Experten und üben Sie mit ihnen Szenarien für die Vorfalldreaktion, bevor es zu einem tatsächlichen Vorfall kommt.

4. Binden Sie vertrauenswürdige Partner oder externe Experten in das Untersuchungs- oder Reaktionsverfahren ein, da sie zusätzliche Kenntnisse und Perspektiven bereitstellen können.
5. Passen Sie Ihre Pläne und Rollen für das Vorfalldmanagement an lokale Vorschriften oder Compliance-Anforderungen an, denen Ihre Organisation unterliegt.
6. Üben und testen Sie Ihre Pläne für die Vorfalldreaktion regelmäßig und beziehen Sie alle definierten Rollen und Verantwortlichkeiten ein. Auf diese Weise können Sie den Prozess rationalisieren und sicherstellen, dass Sie koordiniert und effizient auf Sicherheitsvorfälle reagieren.
7. Überprüfen und aktualisieren Sie Rollen, Verantwortlichkeiten und RACI-Diagramm regelmäßig oder bei Änderungen von Organisationsstruktur oder Anforderungen.

Die AWS-Reaktionsteams und der Support

- AWS Support
 - [Support](#) bietet eine Reihe an Plänen, die den Zugriff auf Tools und das Know-how für den Erfolg und den betriebsbereiten Zustand der AWS-Lösungen ermöglichen. Wenn Sie technischen Support und weitere Ressourcen benötigen, um Ihre AWS-Umgebung zu planen, bereitzustellen und zu optimieren, können Sie einen Supportplan auswählen, der am besten zu Ihrem AWS-Anwendungsfall passt.
 - Das [Support-Center](#) in der AWS-Managementkonsole (Anmeldung erforderlich) ist Ihre zentrale Anlaufstelle, um Unterstützung bei Problemen zu erhalten, die sich auf Ihre AWS-Ressourcen auswirken. Der Zugriff auf den Support wird über AWS Identity and Access Management gesteuert. Weitere Informationen zum Zugriff auf Support-Funktionen finden Sie unter [Erste Schritte mit Support](#).
- AWS Kundenvorfalldreaktionsteam (CIRT)
 - Das AWS-Kundenvorfalldreaktionsteam (CIRT) ist ein spezialisiertes globales, rund um die Uhr verfügbares AWS-Team, das Kunden bei aktiven Sicherheitsereignissen auf Kundenseite des [AWS-Modells der geteilten Verantwortung](#) unterstützt.
 - Wenn das AWS-CIRT Sie unterstützt, bietet es Hilfe bei der Fehlererkennung und Wiederherstellung eines aktiven Sicherheitsereignisses in AWS an. Sie können mithilfe von AWS-Service-Protokollen bei der Ursachenanalyse helfen und Ihnen Empfehlungen für die Wiederherstellung geben. Sie können Ihnen auch Sicherheitsempfehlungen und bewährte Methoden an die Hand geben, mit denen Sie Sicherheitsereignisse in Zukunft vermeiden können.

- AWS-Kunden können das AWS-CIRT über einen [Support-Fall](#) einbinden.
- [AWS Security Incident Response](#)
 - AWS Security Incident Response wurde auf der re:Invent 2024 angekündigt, ein verwalteter Service zur Reaktion auf Sicherheitsvorfälle, der sowohl moderne Triage-Technologie als auch HITL nutzt. Der Service nimmt alle Ergebnisse von GuardDuty und alle Ergebnisse Dritter auf, die an AWS Security Hub CSPM zur Triage gesendet wurden, um den Kunden nur über Ergebnisse zu informieren, die einer Untersuchung bedürfen. Der Service bietet auch ein Portal, über das reaktive Fälle eingereicht werden können, falls der Kunde ein Sicherheitsereignis bemerkt, und Unterstützung durch das Advanced Incident Response Team von AWS angefordert werden kann.
- Unterstützung für DDoS-Reaktion
 - AWS bietet [AWS Shield](#), das einen verwalteten Distributed Denial of Service (DDoS)-Schutz-Service bereitstellt, der in AWS ausgeführte Web-Anwendungen schützt. Shield bietet eine ständig aktive Erkennung und automatische Inline-Schutzmaßnahmen, mit denen Ausfallzeiten und Latenz von Anwendungen minimiert werden können. Sie müssen also nicht Support kontaktieren, um vom DDoS-Schutz zu profitieren. Shield umfasst zwei Stufen: AWS Shield Standard und AWS Shield Advanced. Weitere Informationen zu den Unterschieden zwischen diesen beiden Stufen finden Sie in der [Shield-Funktionsdokumentation](#).
- AWS Managed Services (AMS)
 - [AWS Managed Services \(AMS\)](#) stellt eine fortlaufende Verwaltung Ihrer AWS-Infrastruktur bereit, damit Sie sich auf Ihre Anwendungen konzentrieren können. AMS trägt durch eine Implementierung bewährter Methoden zur Verwaltung Ihrer Infrastruktur dazu bei, den Betriebsaufwand zu reduzieren und das Risiko zu senken. Außerdem automatisiert AMS häufige Aktivitäten wie Änderungsanforderungen, Überwachung, Patch-Verwaltung, Sicherheit sowie Backup-Services und bietet während der gesamten Lebensdauer Services zum Bereitstellen, Ausführen und Unterstützen Ihrer Infrastruktur.
 - AMS übernimmt die Verantwortung für die Bereitstellung einer Reihe von Sicherheitskontrollen und bietet rund um die Uhr Erstreaktion auf Warnungen an. Wenn eine Warnung ausgelöst wird, folgt AMS einer Reihe automatisierter und manueller Standard-Playbooks, um eine konsistente Reaktion zu gewährleisten. Diese Playbooks werden den AMS-Kunden während des Onboardings zur Verfügung gestellt, damit sie eine Reaktion entwickeln und mit AMS abstimmen können.

Erstellen des Vorfallreaktionsplans

Der Vorfallreaktionsplan ist als Grundlage für Ihr Vorfallreaktionsprogramm und Ihre Vorfallreaktionsstrategie konzipiert. Er sollte immer formell schriftlich festgehalten werden. Ein Vorfallreaktionsplan enthält in der Regel folgende Abschnitte:

- Überblick über das Vorfallreaktionsteam: Enthält die Ziele und Funktionen des Vorfallreaktionsteams.
- Rollen und Zuständigkeiten: Hier werden die für die Vorfallreaktion zuständigen Stakeholder aufgeführt und ihre Rollen im Falle eines Vorfalls beschrieben.
- Kommunikationsplan: Enthält Kontaktinformationen und gibt an, wie Sie während eines Vorfalls kommunizieren.
- Alternative Kommunikationsmethoden: Es hat sich bewährt, Out-of-Band-Kommunikation als Alternative für die Kommunikation bei Vorfällen zu verwenden. Ein Beispiel für eine Anwendung, die einen sicheren Out-of-Band-Kommunikationskanal bereitstellt, ist AWS Wickr.
- Phasen der Vorfallreaktion und zu ergreifende Maßnahmen: Hier sind die Phasen der Vorfallreaktion aufgeführt (beispielsweise Erkennung, Analyse, Beseitigung, Eindämmung und Wiederherstellung) – einschließlich der in diesen Phasen zu ergreifenden allgemeinen Maßnahmen.
- Definitionen des Schweregrads und der Priorisierung des Vorfalls: Hier wird erläutert, wie der Schweregrad eines Vorfalls klassifiziert wird, wie der Vorfall priorisiert wird und wie sich die Schweregraddefinitionen dann auf die Eskalationsverfahren auswirken.

Diese Abschnitte sind zwar in Unternehmen verschiedener Größen und Branchen üblich, der Vorfallreaktionsplan ist jedoch für jede Organisation individuell. Erstellen Sie einen Vorfallreaktionsplan, der für Ihre Organisation am besten geeignet ist.

Ressourcen

Zugehörige bewährte Methoden:

- [SEC04 Erkennung](#)

Zugehörige Dokumente:

- [Leitfaden zur Reaktion auf Sicherheitsvorfälle in AWS](#)
- [NIST: Computer Security Incident Handling Guide](#)

SEC10-BP03 Vorbereiten forensischer Funktionen

Bevor es zu einem Sicherheitsvorfall kommt, empfiehlt es sich gegebenenfalls, forensische Funktionen zur Unterstützung der Untersuchung von Sicherheitsereignissen zu entwickeln.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Konzepte aus der traditionellen On-Premises-Forensik gelten auch für AWS. Wichtige Informationen für den Einstieg in den Aufbau forensischer Funktionen in der AWS Cloud finden Sie in den [Strategien für forensische Untersuchungsumgebungen in der AWS Cloud](#).

Nachdem Sie Ihre Umgebung und AWS-Konto-Struktur für die Forensik eingerichtet haben, können Sie die Technologien definieren, die für eine effektive Anwendung forensisch fundierter Methoden in den vier Phasen erforderlich sind:

- **Sammlung:** Erfassen Sie relevante AWS-Protokolle wie AWS CloudTrail, AWS Config, VPC Flow Logs und Protokolle auf Host-Ebene. Erfassen Sie Snapshots, Backups und Speicherabbilder der betroffenen AWS-Ressourcen, sofern verfügbar.
- **Prüfung:** Prüfen Sie die erfassten Daten, indem Sie die relevanten Informationen extrahieren und bewerten.
- **Analyse:** Analysieren Sie die erfassten Daten, um den Vorfall zu verstehen und Schlüsse daraus zu ziehen.
- **Berichterstellung:** Präsentieren Sie die Informationen, die sich aus der Analysephase ergeben.

Implementierungsschritte

Vorbereiten Ihrer forensischen Umgebung

[AWS Organizations](#) hilft Ihnen bei der zentralen Verwaltung und Steuerung einer AWS-Umgebung, während Sie AWS-Ressourcen erweitern und skalieren. Eine AWS-Organisation konsolidiert Ihre AWS-Konten, sodass Sie sie als eine einzige Einheit verwalten können. Mithilfe von Organisationseinheiten können Sie Konten gruppieren und als eine Einheit verwalten.

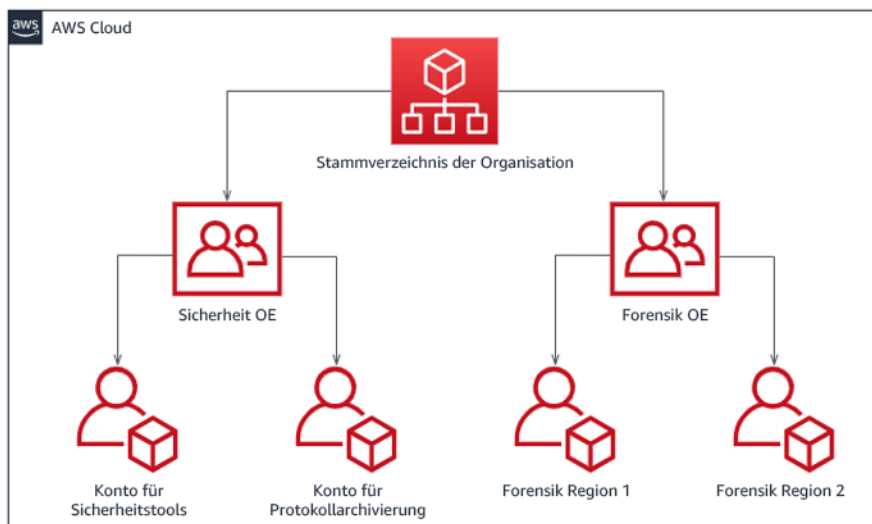
Für die Reaktion auf Vorfälle ist es hilfreich, über eine AWS-Konto-Struktur zu verfügen, die die Funktionen der Vorfallreaktion unterstützt. Dazu gehören eine sicherheitsbezogene Organisationseinheit und eine forensische Organisationseinheit. Innerhalb der sicherheitsbezogenen Organisationseinheit sollten Sie über Konten für Folgendes verfügen:

- **Protokollarchivierung:** Aggregieren Sie Protokolle in einem für die Protokollarchivierung vorgesehenen AWS-Konto mit eingeschränkten Berechtigungen.
- **Sicherheits-Tools:** Zentralisieren Sie Sicherheits-Services in einem AWS-Konto für Sicherheitstools. Dieses Konto fungiert als delegierter Administrator für Sicherheits-Services.

Innerhalb der forensischen Organisationseinheit haben Sie die Möglichkeit, für jede Region, in der Sie tätig sind, eines oder mehrere forensische Konten zu implementieren, je nachdem, was für Ihr Geschäfts- und Betriebsmodell am besten geeignet ist. Wenn Sie ein forensisches Konto pro Region erstellen, können Sie die Erstellung von AWS-Ressourcen außerhalb dieser Region blockieren und so das Risiko verringern, dass Ressourcen in eine unbeabsichtigte Region kopiert werden. Wenn Sie beispielsweise nur in den Regionen „USA Ost (Nord-Virginia)“ (us-east-1) und „USA West (Oregon)“ (us-west-2) aktiv sind, würde die forensische Organisationseinheit zwei Konten umfassen: eins für us-east-1 und eins für us-west-2.

Sie können ein forensisches AWS-Konto für mehrere Regionen erstellen. Achten Sie darauf, dass Sie Ihre Anforderungen an die Datensouveränität einhalten, wenn Sie AWS-Ressourcen in dieses Konto kopieren. Da die Bereitstellung neuer Konten etwas dauert, ist es unerlässlich, die forensischen Konten rechtzeitig vor einem Vorfall einzurichten und zu instrumentieren, damit die Notfallteams vorbereitet sind und sie effektiv nutzen können.

Das folgende Diagramm zeigt eine Beispiel-Kontenstruktur mit einer forensischen Organisationseinheit mit regionsspezifischen forensischen Konten:



Regionsspezifische Kontenstruktur für die Reaktion auf Vorfälle

Erfassen von Backups und Snapshots

Die Einrichtung von Backups wichtiger Systeme und Datenbanken ist für die Wiederherstellung nach einem Sicherheitsvorfall und für forensische Zwecke von entscheidender Bedeutung. Mit vorhandenen Backups können Sie Ihre Systeme wieder in einen vorherigen sicheren Zustand versetzen. In AWS können Sie Snapshots von verschiedenen Ressourcen erstellen. Snapshots bieten Ihnen zeitpunktbezogene Backups dieser Ressourcen. Es gibt viele AWS-Services, die Sie beim Backup und der Wiederherstellung unterstützen können. Einzelheiten zu diesen Services und Ansätzen für Backup und Wiederherstellung finden Sie unter [Präskriptive Leitlinien für Backup und Wiederherstellung](#) sowie unter [Verwendung von Backups zur Wiederherstellung nach Sicherheitsvorfällen](#).

Vor allem, wenn es um Situationen wie Ransomware geht, ist es wichtig, dass Ihre Backups gut geschützt sind. Hinweise zum Schutz Ihrer Backups finden Sie in den [10 besten Sicherheitsmethoden zum Schutz von Backups in AWS](#). Zusätzlich zum Schutz Ihrer Backups sollten Sie Ihre Backup- und Wiederherstellungsprozesse regelmäßig testen, um sicherzustellen, dass die vorhandenen Technologien und Prozesse wie erwartet funktionieren.

Automatisieren der Forensik

Während eines Sicherheitsereignisses muss Ihr Vorfallreaktionsteam in der Lage sein, schnell Nachweise zu sammeln und zu analysieren und gleichzeitig die Genauigkeit für den Zeitraum rund um das Ereignis aufrechtzuerhalten (beispielsweise durch Erfassen von Protokollen zu einem bestimmten Ereignis oder einer bestimmten Ressource oder durch Erfassen von Speicherabbildern einer Amazon-EC2-Instance). Für das Vorfallreaktionsteam ist es sowohl schwierig als auch zeitaufwändig, die relevanten Nachweise manuell zu erfassen, insbesondere bei einer großen Anzahl von Instances und Konten. Darüber hinaus kann die manuelle Erfassung anfällig für menschliche Fehler sein. Daher sollten Sie die Automatisierung für die Forensik so weit wie möglich entwickeln und implementieren.

AWS bietet eine Reihe von Automatisierungsressourcen für die Forensik, die weiter unten im Abschnitt „Ressourcen“ aufgeführt sind. Diese Ressourcen sind Beispiele für forensische Muster, die von entwickelt und von Kunden implementiert wurden. Obwohl sie für den Anfang eine nützliche Referenzarchitektur sein können, sollten Sie erwägen, sie zu ändern oder neue forensische Automatisierungsmuster zu erstellen, die auf Ihrer Umgebung, Ihren Anforderungen, Tools und forensischen Prozessen basieren.

Ressourcen

Zugehörige Dokumente:

- [Leitfaden zur Reaktion auf Sicherheitsvorfälle in AWS: Entwickeln forensischer Funktionen](#)

- [Leitfaden zur Reaktion auf Sicherheitsvorfälle in AWS: Forensische Ressourcen](#)
- [Strategien für forensische Untersuchungsumgebungen in der AWS Cloud](#)
- [Automatisieren der forensischen Datenträgererfassung in AWS](#)
- [Präskriptive AWS-Anleitung: Automatisieren der Vorfallreaktion und Forensik](#)

Zugehörige Videos:

- [Automatisieren der Vorfallreaktion und Forensik](#)

Zugehörige Beispiele:

- [Framework für die automatisierte Reaktion auf Vorfälle und für die Forensik](#)
- [Automatisierter forensischer Orchestrator für Amazon EC2](#)

SEC10-BP04 Entwickeln und Testen von Playbooks für die Reaktion auf Sicherheitsvorfälle

Ein wichtiger Teil der Vorbereitung Ihrer Prozesse zur Vorfallreaktion ist die Entwicklung von Playbooks. Playbooks für die Vorfallreaktion enthalten präskriptive Anleitungen und Schritte, die Sie ausführen sollten, wenn ein Sicherheitsereignis eintritt. Eine klare Struktur und klare Schritte vereinfachen die Reaktion und verringern die Wahrscheinlichkeit menschlicher Fehler.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Playbooks sollten für Vorfallszenarien wie die folgenden erstellt werden:

- Erwartete Vorfälle: Sie sollten Playbooks für zu erwartende Vorfälle erstellen. Dazu gehören Bedrohungen wie Denial of Service (DoS), Ransomware und die Kompromittierung von Anmeldeinformationen.
- Bekannte Sicherheitserkenntnisse oder Warnungen: Sie sollten Playbooks für bekannte Sicherheitserkenntnisse und Warnungen erstellen, z. B. aus Amazon GuardDuty. Wenn Sie eine GuardDuty-Erkennung erhalten, sollte das Playbook klare Schritte beschreiben, um zu verhindern, dass die Warnung falsch behandelt oder ignoriert wird. Weitere Informationen und Anleitungen zur Behebung finden Sie unter [Beheben von Sicherheitsproblemen, die von GuardDuty entdeckt wurden](#).

Playbooks sollten technische Schritte enthalten, die ein Sicherheitsanalyst ausführen muss, um einen potenziellen Sicherheitsvorfall angemessen zu untersuchen und darauf zu reagieren.

Das Customer Incident Response Team (CIRT) von AWS hat ein [GitHub-Repository mit Playbooks zur Reaktion auf Vorfälle](#) veröffentlicht, die nach Bedrohungsszenario, Typ und Ressource geordnet sind. Diese Playbooks können an Ihre bestehenden Verfahren zur Reaktion auf Vorfälle angepasst werden oder als Grundlage für die Entwicklung neuer Verfahren dienen.

Implementierungsschritte

Zu den Elementen, die in ein Playbook aufgenommen werden sollten, gehören:

- **Playbook-Übersicht:** Welches Risiko- oder Vorfallszenario behandelt dieses Playbook? Was ist das Ziel des Playbooks?
- **Voraussetzungen:** Welche Protokolle, Erkennungsmechanismen und automatisierten Tools sind für dieses Vorfallszenario erforderlich? Wie lautet die erwartete Benachrichtigung?
- **Kommunikations- und Eskalationsinformationen:** Wer ist beteiligt und wie lauten die Kontaktinformationen? Welche Aufgaben haben die einzelnen Stakeholder?
- **Reaktionsschritte:** Welche taktischen Maßnahmen sollten in den einzelnen Phasen der Vorfallreaktion ergriffen werden? Welche Abfragen sollte ein Analyst ausführen? Welcher Code sollte ausgeführt werden, um das gewünschte Ergebnis zu erzielen?
 - **Erkennen:** Wie wird der Vorfall erkannt?
 - **Analysieren:** Wie wird der Umfang der Auswirkungen bestimmt?
 - **Eindämmen:** Wie wird der Vorfall isoliert, um den Umfang zu begrenzen?
 - **Beseitigen:** Wie wird die Bedrohung aus der Umgebung entfernt?
 - **Wiederherstellen:** Wie wird das betroffene System oder die betroffene Ressource wieder in der Produktion bereitgestellt?
- **Erwartete Ergebnisse:** Was ist das erwartete Ergebnis des Playbooks, nachdem Abfragen und Code ausgeführt wurden?

Ressourcen

Zugehörige bewährte Methoden für Well-Architected:

- [SEC10-BP02 Entwickeln von Vorfallmanagementplänen](#)

Zugehörige Dokumente:

- [Framework für Playbooks für die Vorfallreaktion](#)
- [Entwickeln eigener Playbooks für die Vorfallreaktion](#)
- [Exemplarische Playbooks für die Vorfallreaktion](#)
- [Entwicklung eines Runbooks für die Vorfallreaktion in AWS mit Jupyter Playbooks und CloudTrail Lake](#)

SEC10-BP05 Vorab bereitgestellter Zugriff

Stellen Sie sicher, dass Notfallteams über den richtigen vorab bereitgestellten Zugriff in AWS verfügen, um die Zeit von der Untersuchung bis zur Wiederherstellung zu verkürzen.

Typische Anti-Muster:

- Verwendung des Root-Kontos für die Reaktion auf Vorfälle
- Veränderung bestehender Benutzerkonten
- Direkte Anpassung von IAM-Berechtigungen bei Just-In-Time-Berechtigungserhöhungen

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

AWS empfiehlt, nach Möglichkeit die Abhängigkeit von langlebigen Anmeldeinformationen zu reduzieren oder ganz zu beseitigen und stattdessen Just-In-Time-Berechtigungseskalationsmechanismen zu verwenden. Langlebige Anmeldeinformationen sind ein potenzielles Sicherheitsrisiko und erhöhen den Verwaltungsaufwand. Für die meisten Verwaltungsaufgaben und für die Reaktion auf Vorfälle empfehlen wir die Implementierung eines [Identitätsverbunds](#) zusammen mit einer [temporären Eskalierung für Administratorzugriff](#). In diesem Modell beantragt ein Benutzer eine höhere Berechtigungsstufe (etwa für eine Vorfallreaktionsrolle). Anschließend wird, sofern der Benutzer grundsätzlich dafür infrage kommt, eine entsprechende Anforderung an einen Genehmiger gesendet. Wenn die Anforderung genehmigt wurde, erhält der Benutzer temporäre [AWS-Anmeldeinformationen](#) für die Durchführung seiner Aufgaben. Nach Ablauf dieser Anmeldeinformationen muss der Benutzer eine neue Erhöhungsanforderung übermitteln.

Wir empfehlen für die meisten Vorfallreaktionsszenarien die Verwendung temporärer Berechtigungseskalierungen. Die korrekte Vorgehensweise ist die Verwendung von [AWS -Security-Token-Service](#) und [Sitzungsrichtlinien](#) zur Festlegung der Zugriffsbereiche.

Es gibt Szenarien, in denen Verbundidentitäten nicht verfügbar sind. Hier ein paar Beispiele:

- Ausfall im Zusammenhang mit einem kompromittierten Identitätsanbieter (IdP)
- Durch fehlerhafte Konfiguration oder menschlichen Fehler beeinträchtigtes Managementsystem für den Verbundzugriff
- Böswillige Aktivität wie etwa ein DDoS-Angriff (Distributed Denial of Service) oder anderweitig verursachte Nichtverfügbarkeit des Systems

Für diese Fälle sollte Notfallzugriff (Break-Glass-Zugriff) konfiguriert werden, um eine Untersuchung und schnelle Behandlung des Vorfalls zu ermöglichen. Wir empfehlen die Verwendung [eines Benutzers, einer Gruppe oder einer Rolle mit ausreichenden Berechtigungen](#) für die Durchführung von Aufgaben und den Zugriff auf AWS-Ressourcen. Verwenden Sie den Root-Benutzer nur für [Aufgaben, die Root-Benutzeranmeldeinformationen erfordern](#). Um zu prüfen, ob die Notfallteams über die korrekte Zugriffsstufe für AWS und andere relevante Systeme verfügen, empfehlen wir die Bereitstellung dedizierter Konten. Die Konten benötigen privilegierten Zugriff und müssen streng kontrolliert und überwacht werden. Die Konten müssen mit den geringstmöglichen Berechtigungen versehen sein, die für die erforderlichen Aufgaben benötigt werden, und die Zugriffsstufe muss auf den Playbooks basieren, die im Rahmen des Vorfallmanagementplans erstellt werden.

Verwenden Sie als bewährte Methode zweckgerichtet erstellte und dedizierte Benutzer und Rollen. Die vorübergehende Eskalierung des Zugriffs eines Benutzers oder einer Rolle über IAM-Richtlinien macht unklar, welchen Zugriff Benutzer während eines Vorfalls hatten, und birgt die Gefahr, dass die eskalierten Berechtigungen später nicht widerrufen werden.

Es ist wichtig, möglichst viele Abhängigkeiten zu entfernen, um sicherzustellen, dass in einem möglichst großen Spektrum von Ausfallszenarien zugegriffen werden kann. Erstellen Sie deshalb ein Playbook, um sicherzustellen, dass Vorfallreaktionsbenutzer als Benutzer in einem dedizierten Sicherheitskonto erstellt und nicht durch einen vorhandenen Verbund oder eine Single Sign-On (SSO)-Lösung verwaltet werden. Alle einzelnen Mitglieder von Notfallteams müssen über ein eigenes benanntes Konto verfügen. Die Kontokonfiguration muss eine [Richtlinie für sichere Passwörter](#) sowie die Multi-Faktor-Authentifizierung (MFA) erzwingen. Wenn die Playbooks zur Vorfallreaktion nur Zugriff auf die AWS-Managementkonsole benötigen, sollten für den Benutzer keine Zugriffsschlüssel konfiguriert werden und er sollte explizit auch keine Zugriffsschlüssel erstellen dürfen. Dies kann mit AWS-Richtlinien oder Service-Kontrollrichtlinien (SCPs) konfiguriert werden, wie in den bewährten -Sicherheitsmethoden für [AWS Organizations SCPs](#) erläutert. Mit Ausnahme der Möglichkeit zur Übernahme von Vorfallreaktionsrollen in anderen Konten sollten die Benutzer über keinerlei Berechtigungen verfügen.

Während eines Vorfalls kann es erforderlich sein, anderen internen oder externen Personen Zugriff zu gewähren, um Untersuchungs-, Korrektur- oder Wiederherstellungsaktivitäten zu unterstützen. Verwenden Sie in diesem Fall den vorher erwähnten Playbook-Mechanismus. Darüber hinaus muss ein Prozess vorhanden sein, um sicherzustellen, dass jeglicher zusätzliche Zugriff sofort nach Abschluss des Vorfalls widerrufen wird.

Um sicherzustellen, dass die Verwendung von Vorfalldaktionsrollen ordnungsgemäß überwacht und geprüft werden kann, ist es wichtig, dass die für diesen Zweck erstellten IAM-Konten nicht von mehreren Personen verwendet werden und dass der Root-Benutzer des AWS-Kontos nur verwendet wird, wenn dies [für eine bestimmte Aufgabe erforderlich](#) ist. Wenn der Root-Benutzer erforderlich ist (zum Beispiel, wenn kein IAM-Zugriff auf ein bestimmtes Konto verfügbar ist), verwenden Sie einen separaten Prozess mit einem verfügbaren Playbook, um die Verfügbarkeit der Anmeldeinformationen und des MFA-Tokens des Root-Benutzers zu prüfen.

Erwägen Sie zur Konfiguration der IAM-Richtlinien für die Vorfalldaktionsrollen die Verwendung von [IAM Access Analyzer](#), um Richtlinien auf der Grundlage von AWS CloudTrail-Protokollen zu erstellen. Gewähren Sie dazu der Vorfalldaktionsrolle in einem Nicht-Produktionskonto Administratorzugriff und durchlaufen Sie Ihre Playbooks. Anschließend kann eine Richtlinie erstellt werden, die nur die ausgeführten Aktionen zulässt. Diese Richtlinie kann dann auf alle Vorfalldaktionsrollen über alle Konten hinweg angewendet werden. Möglicherweise möchten Sie eine separate IAM-Richtlinie für jedes Playbook erstellen, um Management und Auditing zu vereinfachen. Beispiel-Playbooks können Reaktionspläne für Ransomware-Angriffe, Datenschutzverletzungen, Verlust von produktionsrelevantem Zugriff oder andere Szenarien enthalten.

Verwenden Sie die Vorfalldaktionskonten, um dedizierte [IAM-Rollen in anderen AWS-Konten-Konten](#) für die Vorfalldaktion anzunehmen. Diese Rollen müssen so konfiguriert sein, dass sie nur von Benutzern im Sicherheitskonto angenommen werden können, und das Vertrauensverhältnis muss erfordern, dass der aufrufende Prinzipal per MFA authentifiziert wurde. Die Rollen müssen eng gefasste IAM-Richtlinien verwenden, um den Zugriff zu steuern. Stellen Sie sicher, dass alle AssumeRole-Anforderungen für diese Rollen in CloudTrail protokolliert und gemeldet werden und dass alle mit diesen Rollen durchgeführten Aktivitäten protokolliert werden.

Es wird nachdrücklich empfohlen, die IAM-Konten und die IAM-Rollen deutlich zu benennen, damit sie in CloudTrail-Protokollen leicht zu finden sind. Geben Sie also beispielsweise den IAM-Konten den Namen `<USER_ID>-BREAK-GLASS` und den IAM-Rollen den Namen `BREAK-GLASS-ROLE`.

[CloudTrail](#) wird verwendet, um API-Aktivitäten in Ihren AWS-Konten zu protokollieren, und sollte zum [Konfigurieren von Warnungen zur Nutzung der Vorfalldaktionsrollen](#) eingesetzt werden.

Weitere Informationen finden Sie im Blog-Beitrag zur Konfiguration von Warnungen bei Verwendung

von Root-Schlüsseln. Die Anweisungen können geändert werden, um den [Amazon CloudWatch-Metrikfilter](#) so zu konfigurieren, dass nach AssumeRole-Ereignissen gefiltert wird, die mit der IAM-Rolle für die Vorfallreaktion zusammenhängen:

```
{ $.eventName = "AssumeRole" && $.requestParameters.roleArn =  
  "<INCIDENT_RESPONSE_ROLE_ARN>" && $.userIdentity.invokedBy NOT EXISTS && $.eventType !=  
  "AwsServiceEvent" }
```

Da die Vorfallreaktionsrollen sehr wahrscheinlich eine hohe Zugriffsstufe haben, ist es wichtig, dass diese Warnungen an eine weit gefasste Gruppe gesendet werden und dass umgehend auf sie reagiert wird.

Während eines Vorfalls kann es vorkommen, dass ein Mitglied eines Notfallteams Zugriff auf Systeme benötigt, die nicht direkt durch IAM geschützt sind. Dazu können Amazon-Elastic-Compute-Cloud-Instances, Amazon-Relational-Database-Service-Datenbanken oder Software as a Service (SaaS)-Plattformen gehören. Es wird nachdrücklich empfohlen, anstelle nativer Protokolle wie SSH oder RDP [AWS Systems Manager Session Manager](#) für alle administrativen Zugriffe auf Amazon-EC2-Instances zu verwenden. Dieser Zugriff kann mit IAM (sicher und geprüft) gesteuert werden. Gegebenenfalls ist es auch möglich, Teile Ihrer Playbooks mithilfe von [Run-Command-Dokumenten von AWS Systems Manager](#) zu automatisieren, um Benutzerfehler zu reduzieren und die Wiederherstellung zu beschleunigen. Für den Zugriff auf Datenbanken und Tools von Drittanbietern empfehlen wir die Speicherung von Anmeldeinformationen in AWS Secrets Manager und die Gewährung des Zugriffs für die Vorfallreaktionsrollen.

Außerdem sollte die Verwaltung der IAM-Konten für die Vorfallreaktion Ihren [Joiners-, Movers- und Leavers-Prozessen](#) hinzugefügt sowie regelmäßig geprüft und getestet werden, um sicherzustellen, dass nur die beabsichtigten Zugriffsrechte gewährt werden.

Ressourcen

Zugehörige Dokumente:

- [Verwaltung des vorübergehend erhöhten Zugriffs auf Ihre AWS-Umgebung](#)
- [Leitfaden zur Reaktion auf Sicherheitsvorfälle in AWS](#)
- [AWS Elastic Disaster Recovery](#)
- [AWS Systems Manager Incident Manager](#)
- [Einrichten einer Kontopasswortrichtlinie für IAM-Benutzer](#)
- [Verwenden der Multi-Faktor-Authentifizierung \(MFA\) in AWS](#)

- [Konfigurieren des kontoübergreifenden Zugriffs mit MFA](#)
- [Verwenden von IAM Access Analyzer zum Erstellen von IAM-Richtlinien](#)
- [Bewährte Methoden für AWS Organizations-Service-Kontrollrichtlinien in einer Mehrkontenumgebung](#)
- [Empfang von Benachrichtigungen, wenn die Root-Zugriffsschlüssel Ihres AWS-Kontos verwendet werden](#)
- [Erstellen detaillierter Sitzungsberechtigungen mithilfe von IAM-verwalteten Richtlinien](#)
- [„Break Glass“-Zugriff](#)

Zugehörige Videos:

- [Automatisieren der Vorfallreaktion und Forensik in AWS](#)
- [DIY-Leitfaden für Runbooks, Vorfallberichte und Vorfallreaktion](#)
- [Vorbereiten und Reagieren auf Sicherheitsvorfälle in Ihrer AWS-Umgebung](#)

SEC10-BP06 Vorabbereitstellen von Tools

Stellen Sie sicher, dass Sicherheitspersonal über die richtigen Tools verfügt, um die Zeit von der Untersuchung bis zur Wiederherstellung zu verkürzen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Zur Automatisierung von Sicherheitsreaktionen und Betriebsfunktionen können Sie eine umfassende Palette von APIs und Tools von AWS verwenden. Sie können die Identitätsverwaltung, die Netzwerksicherheit, den Datenschutz und Überwachungsfunktionen vollständig automatisieren und mithilfe gängiger Softwareentwicklungsmethoden bereitstellen, die Sie bereits eingerichtet haben. Durch die Sicherheitsautomatisierung kann Ihr System Überwachungs- und Überprüfungsaufgaben übernehmen und eine Reaktion initiieren (im Gegensatz zur manuellen Überwachung der Sicherheitslage und manuellen Reaktion auf Ereignisse).

Wenn Ihre Vorfallreaktionsteams weiterhin auf die gleiche Weise auf Warnungen reagieren, werden Warnungen möglicherweise nicht mehr ernst genommen. Im Laufe der Zeit kann das Team für Warnungen desensibilisiert werden und entweder Fehler bei der Verarbeitung normaler Situationen machen oder außergewöhnliche Warnungen übersehen. Automatisierung hilft, eine Abstumpfung

gegenüber Warnungen zu vermeiden, indem Funktionen verwendet werden, die repetitive und gewöhnliche Warnungen verarbeiten, sodass Mitarbeiter die nötigen freien Kapazitäten haben, um sich um sensible und besondere Vorfälle zu kümmern. Die Integration von Systemen zur Erkennung von Anomalien wie Amazon GuardDuty, AWS CloudTrail Insights und Amazon CloudWatch Anomaly Detection kann den durch allgemeine schwellenwertbasierte Warnungen verursachten Aufwand reduzieren.

Sie können manuelle Prozesse verbessern, indem Sie die Schritte im Prozess programmatisch automatisieren. Nachdem Sie das Korrekturmuster für ein Ereignis definiert haben, können Sie dieses Muster in umsetzbare Logik zerlegen und den Code schreiben, um diese Logik auszuführen. Notfallteams können anschließend diesen Code ausführen, um das Problem zu beheben. Mit der Zeit können Sie immer mehr Schritte automatisieren und schließlich häufige Vorfälle automatisch behandeln.

Bei einer Sicherheitsuntersuchung müssen Sie relevante Protokolle heranziehen können, um alle Aspekte und den Zeitrahmen des Vorfalls zu verstehen. Protokolle werden auch für die Generierung von Warnungen benötigt, die auf bestimmte Ereignisse aufmerksam machen. Es ist sehr wichtig, Abfrage- und Abrufmechanismen auszuwählen, zu aktivieren, zu speichern und einzurichten sowie die Alarmierung einzurichten. Darüber hinaus stellt [Amazon Detective](#) eine effektive Möglichkeit zur Bereitstellung von Tools zum Durchsuchen von Protokolldaten dar.

AWS bietet über 200 Cloud-Services und Tausende von Funktionen. Wir empfehlen Ihnen, die Services zu überprüfen, die Ihre Strategie zur Vorfallreaktion unterstützen und vereinfachen können.

Zusätzlich zur Protokollierung sollten Sie eine [Markierungsstrategie](#) entwickeln und implementieren. Die Markierung kann dabei helfen, einen Kontext im Zusammenhang mit dem Zweck einer AWS-Ressource bereitzustellen. Die Markierung kann auch für die Automatisierung verwendet werden.

Implementierungsschritte

Auswählen und Einrichten von Protokollen für die Analyse und Alarmierung

In der folgenden Dokumentation finden Sie Informationen zur Konfiguration der Protokollierung für die Vorfallreaktion:

- [Protokollierungsstrategien für die Reaktion auf Sicherheitsvorfälle](#)
- [SEC04-BP01 Konfigurieren der Service- und Anwendungsprotokollierung](#)

Aktivieren von Sicherheits-Services zur Unterstützung von Erkennung und Reaktion

AWS bietet native Erkennungs-, Präventions- und Reaktionsfunktionen und es können andere Services für den Aufbau benutzerdefinierter Sicherheitslösungen verwendet werden. Eine Liste der wichtigsten Services für die Reaktion auf Sicherheitsvorfälle finden Sie unter [Definitionen der Cloud-Funktionen](#) und auf der [Security Incident Response-Homepage](#).

Entwickeln und Implementieren einer Markierungsstrategie

Es kann schwierig sein, kontextbezogene Informationen zum geschäftlichen Anwendungsfall und zu relevanten internen Stakeholdern rund um eine AWS-Ressource zu erhalten. Eine Möglichkeit sind Tags, die Ihren AWS-Ressourcen Metadaten zuweisen und aus einem benutzerdefinierten Schlüssel und Wert bestehen. Sie können Tags erstellen, um Ressourcen nach Zweck, Besitzer, Umgebung, Art der verarbeiteten Daten und anderen Kriterien Ihrer Wahl zu kategorisieren.

Eine konsistente Markierungsstrategie kann Reaktionen beschleunigen und den Zeitaufwand für den organisatorischen Kontext minimieren, da Sie Kontextinformationen zu einer AWS-Ressource schnell identifizieren und erkennen können. Tags können auch als Mechanismus zur Initiierung von Reaktionsautomatisierungen dienen. Weitere Informationen über zu markierende Elemente finden Sie unter [Markieren Ihrer AWS-Ressourcen](#). Sie sollten zunächst die Tags definieren, die Sie in Ihrer Organisation implementieren möchten. Anschließend können Sie diese Markierungsstrategie implementieren und erzwingen. Weitere Einzelheiten zur Implementierung und Erzwingung finden Sie unter [Implementieren einer Markierungsstrategie für AWS-Ressourcen mithilfe von AWS-Markierungsrichtlinien und Service-Kontrollrichtlinien \(SCPs\)](#).

Ressourcen

Zugehörige bewährte Methoden für Well-Architected:

- [SEC04-BP01 Konfigurieren der Service- und Anwendungsprotokollierung](#)
- [SEC04-BP02 Erfassen von Protokollen, Erkenntnissen und Metriken an standardisierten Orten](#)

Zugehörige Dokumente:

- [Protokollierungsstrategien für die Reaktion auf Sicherheitsvorfälle](#)
- [Definitionen der Cloud-Funktionen für die Vorfallreaktion](#)

Zugehörige Beispiele:

- [Bedrohungserkennung und -reaktion mit Amazon GuardDuty und Amazon Detective](#)
- [Security-Hub-Workshop](#)

- [Management von Schwachstellen mit Amazon Inspector](#)

SEC10-BP07 Durchführen von Simulationen

Organisationen wachsen und entwickeln sich weiter. Gleiches gilt auch für die Bedrohungslandschaft. Daher ist es wichtig, Ihre Fähigkeiten zur Vorfallreaktion kontinuierlich zu überprüfen. Die Durchführung von Simulationen (auch bekannt als Gamedays) ist eine Methode, mit der diese Bewertung durchgeführt werden kann. Bei Simulationen werden reale Sicherheitsereignisse als Szenarien verwendet, die die Taktiken, Techniken und Verfahren (Tactics, Techniques and Procedures, TTPs) eines Bedrohungsakteurs nachahmen und es einer Organisation ermöglichen, ihre Fähigkeiten zur Vorfallreaktion einzusetzen und zu bewerten, indem sie auf diese simulierten Cyberereignisse so reagieren, wie sie es im Ernstfall tun würden.

Vorteile der Nutzung dieser bewährten Methode: Simulationen haben eine Vielzahl von Vorteilen:

- Validierung der Cybersicherheit und Stärkung des Vertrauens Ihres Vorfallreaktionsteams
- Testen der Genauigkeit und Effizienz von Tools und Workflows
- Optimierung der Kommunikations- und Eskalationsmethoden Ihres Vorfallreaktionsplans
- Möglichkeit, auf weniger verbreitete Vektoren zu reagieren

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Es gibt drei Hauptarten von Simulationen:

- **Tabletop-Übungen:** Beim Tabletop-Ansatz für Simulationen handelt es sich um eine Diskussionsrunde, an der die verschiedenen, mit der Vorfallreaktion betrauten Stakeholder teilnehmen, um Rollen und Verantwortlichkeiten zu üben und etablierte Kommunikationstools und Playbooks zu verwenden. Die Übung kann in der Regel an einem ganzen Tag sowie an einem virtuellen und/oder physischen Ort durchgeführt werden. Da sie auf Diskussionen basiert, konzentriert sich die Tabletop-Übung auf Prozesse, Menschen und Zusammenarbeit. Technologie ist ein integraler Bestandteil der Diskussion, aber der tatsächliche Einsatz von Tools oder Skripten für die Vorfallreaktion ist in der Regel kein Teil der Tabletop-Übung.
- **Übungen des lila Teams:** Übungen des lila Teams verbessern die Zusammenarbeit zwischen dem Vorfallreaktionsteam (blaues Team) und den simulierten Bedrohungsakteuren (rotes Team). Das blaue Team besteht aus Mitgliedern des Security Operations Center (SOC), kann aber auch andere Stakeholder enthalten, die an einem tatsächlichen Cyberereignis beteiligt wären.

Das rote Team besteht aus einem Penetrationstest-Team oder wichtigen Stakeholdern, die in offensiver Sicherheit geschult sind. Das rote Team arbeitet bei der Planung eines Szenarios mit den Übungsleitern zusammen, damit das Szenario korrekt und durchführbar ist. Bei Übungen des lila Teams liegt das Hauptaugenmerk auf den Erkennungsmechanismen, den Tools und den Standard-Betriebsabläufen (Standard Operating Procedures, SOPs), mit denen die Maßnahmen zur Vorfalldreaktion unterstützt werden.

- **Übungen des roten Teams:** Bei einer Übung des roten Teams führt das Offensivteam (rotes Team) eine Simulation durch, um ein bestimmtes Ziel oder eine Reihe von Zielen aus einem vorher festgelegten Bereich zu erreichen. Die Verteidiger (blaues Team) kennen nicht unbedingt den Umfang und die Dauer der Übung, was eine realistischere Einschätzung darüber ermöglicht, wie sie auf einen tatsächlichen Vorfall reagieren würden. Da es sich bei den Übungen des roten Teams um invasive Tests handeln kann, sollten Sie vorsichtig sein und Kontrollen implementieren, um sicherzustellen, dass die Übung Ihrer Umgebung nicht tatsächlich schadet.

Erwägen Sie, in regelmäßigen Abständen Cybersimulationen durchzuführen. Jeder Übungstyp kann den Teilnehmern und der gesamten Organisation einzigartige Vorteile bieten. Sie können also etwa mit weniger komplexen Simulationstypen beginnen (beispielsweise mit Tabletop-Übungen) und dann zu komplexeren Simulationstypen übergehen (Übungen des roten Teams). Wählen Sie auf der Grundlage Ihres Sicherheitsreifegrads, Ihrer Ressourcen und der gewünschten Ergebnisse einen Simulationstyp aus. Einige Kunden entscheiden sich aufgrund der Komplexität und der Kosten möglicherweise gegen Übungen des roten Teams.

Implementierungsschritte

Unabhängig von der Art der gewählten Simulation folgen diese im Allgemeinen den folgenden Implementierungsschritten:

1. **Definieren Sie die wichtigsten Übungselemente:** Definieren Sie das Simulationsszenario und die Ziele der Simulation. Beide sollten von der Führungsebene akzeptiert werden.
2. **Identifizieren Sie die wichtigsten Stakeholder:** Für eine Übung sind mindestens Übungsleiter und Teilnehmer erforderlich. Je nach Szenario können gegebenenfalls weitere Stakeholder einbezogen werden – etwa aus der Rechts- oder Kommunikationsabteilung oder aus der Geschäftsleitung.
3. **Erstellen und testen Sie das Szenario:** Das Szenario muss möglicherweise während der Erstellung neu definiert werden, falls bestimmte Elemente nicht realisierbar sind. Als Ergebnis dieser Phase wird ein fertiges Szenario erwartet.
4. **Führen Sie die Simulation durch:** Die Art der Simulation bestimmt die Durchführung (ein Szenario auf Papier im Vergleich zu einem hochtechnischen, simulierten Szenario). Die Übungsleiter sollten

ihre Taktiken an den Übungsobjekten ausrichten und alle Übungsteilnehmer nach Möglichkeit einbeziehen, um den größtmöglichen Nutzen zu erzielen.

5. Arbeiten Sie den After-Action Report (AAR, Abschlussbericht) aus: Identifizieren Sie Bereiche mit guten Ergebnissen sowie verbesserungswürdige Bereiche und potenzielle Lücken. Der AAR sollte die Effektivität der Simulation sowie die Reaktion des Teams auf das simulierte Ereignis messen, damit der Fortschritt im Laufe der Zeit mit zukünftigen Simulationen verfolgt werden kann.

Ressourcen

Zugehörige Dokumente:

- [Leitfaden zur Reaktion auf Sicherheitsvorfälle in AWS](#)

Zugehörige Videos:

- [AWS GameDay – Sicherheitsausgabe](#)
- [Ausführen effektiver Simulationen von Reaktionen auf Sicherheitsvorfälle](#)

SEC10-BP08 Entwickeln eines Frameworks, um aus Vorfällen zu lernen

Die Implementierung eines Erkenntnis-Frameworks und einer Ursachenanalyse kann nicht nur zur Verbesserung der Reaktion auf Vorfälle, sondern auch zur Verhinderung einer Wiederholung des Vorfalls beitragen. Durch das Lernen aus Vorfällen können Sie verhindern, dass sich die gleichen Fehler, Risiken oder Fehlkonfigurationen wiederholen. Dies verbessert nicht nur Ihre Sicherheitslage, sondern minimiert auch den Zeitverlust durch vermeidbare Situationen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Es ist wichtig, ein Erkenntnis-Framework zu implementieren, das ganz allgemein Folgendes ermittelt und erreicht:

- Wann kommt es zu Erkenntnissen?
- Was beinhaltet der Erkenntnisprozess?
- Wie werden Erkenntnisse gewonnen?
- Wer ist auf welche Weise an dem Prozess beteiligt?
- Wie werden verbesserungswürdige Bereiche identifiziert?

- Wie stellen Sie sicher, dass Verbesserungen effektiv verfolgt und implementiert werden?

Das Framework sollte sich nicht auf Einzelpersonen konzentrieren oder ihnen die Schuld geben, sondern stattdessen den Fokus auf die Verbesserung der Tools und Prozesse legen.

Implementierungsschritte

Abgesehen von den zuvor aufgeführten Ergebnissen auf hoher Ebene ist es wichtig, sicherzustellen, dass Sie die richtigen Fragen stellen, um den größtmöglichen Nutzen (Informationen, die zu umsetzbaren Verbesserungen führen) aus dem Prozess zu ziehen. Berücksichtigen Sie die folgenden Fragen, um Ihre Diskussionen über Erkenntnisse zu fördern:

- Was ist vorgefallen?
- Wann wurde der Vorfall zum ersten Mal identifiziert?
- Wie wurde er identifiziert?
- Von welchen Systemen wurde eine Warnung im Zusammenhang mit der Aktivität ausgegeben?
- Welche Systeme, Services und Daten waren beteiligt?
- Was ist konkret passiert?
- Was hat gut funktioniert?
- Was hat nicht gut funktioniert?
- Welcher Prozess oder welche Verfahren haben versagt oder konnten nicht skaliert werden, um auf den Vorfall zu reagieren?
- Was kann in den folgenden Bereichen verbessert werden:
 - Personen
 - Waren die Mitarbeiter, die kontaktiert werden mussten, tatsächlich verfügbar und war die Kontaktliste auf dem neuesten Stand?
 - Fehlten den Mitarbeitern Schulungen oder Fähigkeiten, die erforderlich waren, um effektiv auf den Vorfall reagieren und ihn untersuchen zu können?
 - Waren die erforderlichen Ressourcen bereit und verfügbar?
 - Prozess
 - Wurden Prozesse und Verfahren eingehalten?
 - Waren Prozesse und Verfahren für diesen Vorfall bzw. für diese Art von Vorfall dokumentiert und verfügbar?
 - Fehlten erforderliche Prozesse und Verfahren?

- Konnten die Notfallteams rechtzeitig auf die erforderlichen Informationen zugreifen, um auf das Problem zu reagieren?
- Technologie
 - Haben die bestehenden Warnsysteme die Aktivität effektiv identifiziert und gemeldet?
 - Wie hätten wir die Zeit bis zur Erkennung um 50 % reduzieren können?
 - Müssen bestehende Warnungen verbessert oder neue Warnungen für diesen Vorfall bzw. für diese Art von Vorfall erstellt werden?
 - War mit den vorhandenen Tools eine effektive Untersuchung (Suche/Analyse) des Vorfalls möglich?
 - Was kann getan werden, um diesen Vorfall bzw. diese Art von Vorfall früher zu erkennen?
 - Was kann getan werden, um zu verhindern, dass sich dieser Vorfall bzw. diese Art von Vorfall wiederholt?
 - Wer ist für den Verbesserungsplan zuständig und wie testen Sie, ob er implementiert wurde?
 - Wie sieht der Zeitplan für die Implementierung und das Testen zusätzlicher Überwachungen oder präventiver Kontrollen und Prozesse aus?

Diese Liste ist nicht vollständig. Sie soll jedoch als Ausgangspunkt dienen, um zu ermitteln, was die Organisations- und Geschäftsanforderungen sind und wie Sie diese analysieren können, um am effektivsten aus Vorfällen zu lernen und Ihre Sicherheitslage kontinuierlich zu verbessern. Am wichtigsten ist, damit zu beginnen und Erkenntnisse standardmäßig in Ihren Prozess zur Vorfallreaktion, in die Dokumentation und in die Erwartungen der Stakeholder zu integrieren.

Ressourcen

Zugehörige Dokumente:

- [AWS-Leitfaden für die Reaktion auf Sicherheitsvorfälle – Entwickeln eines Frameworks, um aus Vorfällen zu lernen](#)
- [NCSC-CAF-Leitfaden – Erkenntnisse](#)

Anwendungssicherheit

Frage

- [SEC 11. Wie beziehen Sie die Sicherheitseigenschaften von Anwendungen während des gesamten Entwurfs-, Entwicklungs- und Bereitstellungslebenszyklus ein und validieren sie?](#)

SEC 11. Wie beziehen Sie die Sicherheitseigenschaften von Anwendungen während des gesamten Entwurfs-, Entwicklungs- und Bereitstellungslebenszyklus ein und validieren sie?

Das Schulen von Personen, das Testen mithilfe von Automatisierung, ein Verständnis der Abhängigkeiten und die Validierung der Sicherheitseigenschaften von Tools und Anwendungen helfen dabei, die Wahrscheinlichkeit eines Sicherheitsproblems bei Produktions-Workloads zu verringern.

Best Practices

- [SEC11-BP01 Schulen für Anwendungssicherheit](#)
- [SEC11-BP02 Automatisieren des Testens während des Entwicklungs- und Veröffentlichungslebenszyklus](#)
- [SEC11-BP03 Durchführen regelmäßiger Penetrationstests](#)
- [SEC11-BP04 Durchführen von Codeüberprüfungen](#)
- [SEC11-BP05 Zentralisieren von Services für Pakete und Abhängigkeiten](#)
- [SEC11-BP06 Programmatisches Bereitstellen von Software](#)
- [SEC11-BP07 Regelmäßiges Bewerten von Sicherheitseigenschaften der Pipelines](#)
- [SEC11-BP08 Entwickeln eines Programms, das Workload-Teams die Verantwortung für die Sicherheit überträgt](#)

SEC11-BP01 Schulen für Anwendungssicherheit

Schulen Sie Ihr Team in Bezug auf Praktiken für die sichere Entwicklung und Ausführung, um die Entwicklung sicherer und qualitativ hochwertiger Software zu fördern. Diese Vorgehensweise hilft Ihrem Team, Sicherheitsprobleme früher im Entwicklungszyklus zu verhindern, zu erkennen und zu beheben. Ziehen Sie Schulungen zu den Themen Bedrohungsmodellierung, sichere Programmierpraktiken und Nutzung von Services für sichere Konfigurationen und Abläufe in Betracht. Bieten Sie Ihrem Team Zugang zu Schulungen über Self-Service-Ressourcen und holen Sie regelmäßig Feedback ein, um kontinuierlich Verbesserungen vorzunehmen.

Gewünschtes Ergebnis: Sie vermitteln Ihrem Team die erforderlichen Kenntnisse und Fähigkeiten, um Software von Anfang an unter dem Aspekt der Sicherheit zu entwerfen und zu entwickeln. Durch Schulungen zu Bedrohungsmodellierung und sicheren Entwicklungspraktiken verfügt Ihr Team über fundierte Kenntnisse potenzieller Sicherheitsrisiken und weiß, wie diese Risiken während des Softwareentwicklungszyklus gemindert werden können. Dieser proaktive Sicherheitsansatz

ist Teil Ihrer Teamkultur und versetzt Sie in die Lage, potenzielle Sicherheitsprobleme frühzeitig zu erkennen und zu beheben. Ihr Team kann somit qualitativ hochwertige, sichere Software und Features effizienter bereitstellen, wodurch die Bereitstellung insgesamt beschleunigt werden kann. In Ihrer Organisation besteht eine auf Zusammenarbeit und Einbeziehung beruhende Sicherheitskultur, wobei alle Entwickler gemeinsam für die Sicherheit verantwortlich sind.

Typische Anti-Muster:

- Sie warten bis zu einer Sicherheitsüberprüfung und berücksichtigen erst dann die Sicherheitseigenschaften eines Systems.
- Sie überlassen alle sicherheitsbezogenen Entscheidungen einem zentralen Sicherheitsteam.
- Sie kommunizieren nicht, wie die im Softwareentwicklungszyklus getroffenen Entscheidungen mit den allgemeinen Sicherheitserwartungen- oder -richtlinien der Organisation im Zusammenhang stehen.
- Sie führen die Sicherheitsüberprüfung zu spät durch.

Vorteile der Nutzung dieser bewährten Methode:

- Bessere Kenntnis der Unternehmensanforderungen hinsichtlich der Sicherheit frühzeitig im Entwicklungszyklus
- Schnellere Auslieferung von Funktionen durch schnelles Identifizieren und Lösen potenzieller Sicherheitsprobleme
- Verbesserte Qualität von Software und Systemen

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Um sichere und qualitativ hochwertige Software zu entwickeln, sollten Sie Ihr Team in Bezug auf gängige Praktiken für die sichere Entwicklung und Ausführung von Anwendungen schulen. Diese Vorgehensweise hilft Ihrem Team, Sicherheitsprobleme früher im Entwicklungszyklus zu verhindern, zu erkennen und zu beheben. So kann eine schnellere Bereitstellung erfolgen.

Um dieses Ziel zu erreichen, sollten Sie Ihr Team mithilfe von AWS-Ressourcen wie dem [Workshop zur Bedrohungsmodellierung](#) in der Bedrohungsmodellierung schulen. Die Bedrohungsmodellierung kann Ihrem Team helfen, potenzielle Sicherheitsrisiken zu verstehen und Systeme von Anfang an mit Blick auf die Sicherheit zu entwerfen. Darüber hinaus können Sie Zugang zu Schulungen von [AWS](#)

[Training and Certification](#), Branchenschulungen oder Schulungen von AWS-Partnern zum Thema „Sichere Entwicklungspraktiken“ bieten. Weitere Informationen zu einem umfassenden Ansatz für Design, Entwicklung, Sicherung und effizienten Betrieb in großem Maßstab finden Sie in der [AWS-DevOps-Anleitung](#).

Definieren und kommunizieren Sie den Prozess der Sicherheitsüberprüfung in Ihrer Organisation klar und deutlich und legen Sie die Verantwortlichkeiten Ihres Teams, des Sicherheitsteams und anderer Stakeholder fest. Veröffentlichen Sie Self-Service-Anleitungen, Codebeispiele und Vorlagen zur Erfüllung Ihrer Sicherheitsanforderungen. Sie können AWS-Services wie [AWS CloudFormation](#), [AWS Cloud Development Kit \(AWS CDK\) \(AWS CDK\)-Constructs](#) und [Service Catalog](#) verwenden, um vorab genehmigte, sichere Konfigurationen bereitzustellen und die Notwendigkeit benutzerdefinierter Einrichtungen zu verringern.

Holen Sie regelmäßig Feedback von Ihrem Team zu seiner Erfahrung mit dem Prozess zur Sicherheitsüberprüfung und den Schulungen ein und verwenden Sie dieses Feedback, um kontinuierlich Verbesserungen zu implementieren. Führen Sie Gamedays oder Bug-Bash-Kampagnen durch, um Sicherheitsprobleme zu identifizieren und zu beheben und gleichzeitig die Fähigkeiten Ihres Teams zu verbessern.

Implementierungsschritte

1. Ermittlung des Schulungsbedarfs: Beurteilen Sie anhand von Umfragen, Codeüberprüfungen oder Diskussionen mit Teammitgliedern den aktuellen Kenntnisstand und die Wissenslücken in Ihrem Team in Bezug auf sichere Entwicklungspraktiken.
2. Planung der Schulung: Erstellen Sie auf der Grundlage des ermittelten Bedarfs einen Schulungsplan, der relevante Themen wie Bedrohungsmodellierung, sichere Programmierungspraktiken, Sicherheitstests und sichere Bereitstellungspraktiken umfasst. Nutzen Sie Ressourcen wie den [Workshop zur Bedrohungsmodellierung](#), Schulungen von [AWS Training and Certification](#), Branchenschulungen oder Schulungsprogramme von AWS-Partnern.
3. Planung und Durchführung von Schulungen: Planen Sie regelmäßige Schulungen oder Workshops für Ihr Team. Hierbei kann es sich je nach Vorlieben des Teams und Verfügbarkeit um Schulungen mit Kursleiter oder um Schulungen zum Selbststudium handeln. Ermutigen Sie zu praktischen Übungen und zur Nutzung von praktischen Beispielen, um das Lernen zu vertiefen.
4. Definition eines Prozesses zur Sicherheitsüberprüfung: Legen Sie in Zusammenarbeit mit Ihrem Sicherheitsteam und anderen Stakeholdern den Prozess zur Sicherheitsüberprüfung für Ihre Anwendungen klar fest. Dokumentieren Sie die Verantwortlichkeiten aller am Prozess beteiligten Teams oder Einzelpersonen, einschließlich Ihres Entwicklungsteams, Ihres Sicherheitsteams und anderer relevanter Stakeholder.

5. Erstellen von Self-Service-Ressourcen: Entwickeln Sie Self-Service-Anleitungen, Codebeispiele und Vorlagen, die zeigen, wie die Sicherheitsanforderungen Ihrer Organisation erfüllt werden können. Ziehen Sie die Verwendung von AWS-Services wie [CloudFormation](#), [AWS CDK-Constructs](#) und [Service Catalog](#) in Betracht, um vorab genehmigte, sichere Konfigurationen bereitzustellen und die Notwendigkeit benutzerdefinierter Einrichtungen zu verringern.
6. Kommunikation und Kontaktpflege: Informieren Sie Ihr Team auf effektive Weise über den Prozess zur Sicherheitsüberprüfung und die verfügbaren Self-Service-Ressourcen. Führen Sie Schulungen oder Workshops durch, um die Teammitglieder mit diesen Ressourcen vertraut zu machen, und vergewissern Sie sich, dass sie über die richtige Handhabung der Ressourcen Bescheid wissen.
7. Einholung von Feedback und Verbesserungen: Holen Sie regelmäßig Feedback von Ihrem Team zu seiner Erfahrung mit dem Prozess zur Sicherheitsüberprüfung und den Schulungen ein. Nutzen Sie dieses Feedback, um Bereiche mit Verbesserungspotenzial zu identifizieren und die Schulungsmaterialien, Self-Service-Ressourcen sowie den Prozess zur Sicherheitsüberprüfung kontinuierlich zu verbessern.
8. Durchführung von Sicherheitsübungen: Organisieren Sie Gamedays oder Bug-Bash-Kampagnen, um Sicherheitsprobleme in Ihren Anwendungen zu identifizieren und zu beheben. Diese Übungen helfen nicht nur dabei, potenzielle Schwachstellen aufzudecken, sondern bieten Ihrem Team auch die Möglichkeit, praktische Erfahrungen zu sammeln und so seine Fähigkeiten in Bezug auf die sichere Entwicklung und Ausführung zu verbessern.
9. Weiteres Lernen und weitere Verbesserungen: Ermutigen Sie Ihr Team, sich über die neuesten Methoden, Tools und Techniken für die sichere Entwicklung auf dem Laufenden zu halten. Überprüfen und aktualisieren Sie Ihre Schulungsmaterialien und Ressourcen regelmäßig, um der sich weiterentwickelnden Sicherheitslandschaft und Änderungen in Bezug auf bewährte Methoden Rechnung zu tragen.

Ressourcen

Zugehörige bewährte Methoden:

- [SEC11-BP08 Entwickeln eines Programms, das Workload-Teams die Verantwortung für die Sicherheit überträgt](#)

Zugehörige Dokumente:

- [AWS Training and Certification](#)
- [Betrachtung der Cloud-Sicherheits-Governance](#)

- [Konzepte für Bedrohungsmodellierung](#)
- [Schulungen beschleunigen – AWS Skills Guild](#)
- [AWS DevOps Sagas](#)

Zugehörige Videos:

- [Proaktive Sicherheit: Überlegungen und Ansätze](#)

Zugehörige Beispiele:

- [Workshop zur Bedrohungsmodellierung](#)
- [Branchenbewusstsein für Entwickler](#)

Zugehörige Services:

- [AWS CloudFormation](#)
- [AWS Cloud Development Kit \(AWS CDK\) \(AWS CDK\) Konstrukte](#)
- [Servicekatalog](#)

SEC11-BP02 Automatisieren des Testens während des Entwicklungs- und Veröffentlichungslebenszyklus

Automatisieren Sie das Testen der Sicherheitseigenschaften während des Entwicklungs- und Veröffentlichungslebenszyklus. Automatisierung vereinfacht die kontinuierliche und wiederholbare Identifizierung potenzieller Probleme. Dadurch wird das Risiko von Sicherheitsproblemen in der bereitgestellten Software verringert.

Gewünschtes Ergebnis: Das Ziel von automatisiertem Testen ist, eine programmatische Möglichkeit zur frühen Erkennung von potenziellen Problemen – häufig im Laufe des Entwicklungslebenszyklus – zu bieten. Wenn Sie Regressionstests automatisieren, können Sie funktionale und nicht funktionale Tests erneut durchführen, um zu überprüfen, ob zuvor getestete Software nach einer Änderung weiterhin wie erwartet funktioniert. Wenn Sie Sicherheitstests für Komponenten definieren, um nach häufigen Fehlkonfigurationen zu suchen, wie einer fehlerhaften oder fehlenden Authentifizierung, können Sie diese Fehler früh im Entwicklungsprozess identifizieren und beheben.

Testautomatisierung verwendet speziell entwickelte Testfälle zur Anwendungsvalidierung auf Basis der Anforderungen und der gewünschten Funktionalität der Anwendung. Das Ergebnis

von automatisiertem Testen basiert auf dem Vergleich zwischen der erstellten Testausgabe und der erwarteten Ausgabe, wodurch der gesamte Lebenszyklus des Testens beschleunigt wird. Testmethoden wie Regressionstests und Modultest-Suites eignen sich am besten zur Automatisierung. Durch die Automatisierung des Testens von Sicherheitseigenschaften können Entwickler automatisiertes Feedback erhalten, ohne auf eine Sicherheitsüberprüfung warten zu müssen. Automatisierte Tests in Form von statischer oder dynamischer Codeanalyse können die Qualität von Code erhöhen und dabei helfen, potenzielle Softwareprobleme früh im Entwicklungslebenszyklus zu erkennen.

Typische Anti-Muster:

- Testfälle und Testergebnisse des automatisierten Testens werden nicht kommuniziert.
- Automatisiertes Testen wird erst unmittelbar vor einer Veröffentlichung durchgeführt.
- Testfälle werden mit sich häufig ändernden Anforderungen automatisiert.
- Es wird keine Anleitung für den Umgang mit den Ergebnissen von Sicherheitstests bereitgestellt.

Vorteile der Nutzung dieser bewährten Methode:

- Verringerte Abhängigkeit von Menschen, die die Sicherheitseigenschaften eines Systems evaluieren
- Konsistente Erkenntnisse bei mehreren Arbeitsabläufen für mehr Konsistenz
- Geringere Wahrscheinlichkeit, dass Sicherheitsprobleme in Produktionssoftware gelangen
- Kürzeres Zeitfenster zwischen der Erkennung und Behebung von Softwareproblemen, da sie früher entdeckt werden
- Erhöhte Sichtbarkeit von systemischem oder wiederholtem Verhalten bei mehreren Arbeitsabläufen, dank derer organisationsweite Verbesserungen vorangetrieben werden können

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Setzen Sie während der Entwicklung Ihrer Software unterschiedliche Mechanismen für das Testen von Software ein, um sicherzustellen, dass Sie Ihre Anwendung sowohl auf funktionale Anforderungen (basierend auf Ihrer Geschäftslogik) als auch auf nicht funktionale Anforderungen testen, die sich auf die Zuverlässigkeit, Leistung und Sicherheit der Anwendung konzentrieren.

Bei statischen Anwendungssicherheitstests (SAST) wird Ihr Quellcode auf ungewöhnliche Sicherheitsmuster untersucht und es werden Hinweise auf fehleranfälligen Code bereitgestellt. SAST nutzt statische Eingaben wie etwa die Dokumentation (Anforderungsspezifikationen, Designdokumentation und Designspezifikationen) und den Anwendungscode, um Tests in Bezug auf eine Reihe von bekannten Sicherheitsproblemen durchzuführen. Statische Code-Analyzer helfen dabei, die Analyse großer Codemengen zu beschleunigen. Die [NIST Quality Group](#) bietet einen Vergleich von [Quellcode-Sicherheitsanalysen](#), der Open-Source-Tools für [Bytecode-Scanner](#) und [Binärcode-Scanner](#) umfasst.

Ergänzen Sie Ihr statisches Testen mit Methoden für dynamische Anwendungssicherheitstests (DAST). Hierbei wird die Anwendung bei ihrer Ausführung getestet, um potenzielles unerwartetes Verhalten zu identifizieren. Dynamisches Testen kann verwendet werden, um potenzielle Probleme zu erkennen, die über die statische Analyse nicht gefunden werden können. Das Testen in der Code-Repository-, Build- und Pipeline-Phase ermöglicht es Ihnen, nach unterschiedlichen Arten potenzieller Fehler in Ihrem Code zu suchen. [Amazon Q Developer](#) bietet Codeempfehlungen, einschließlich Sicherheitsscans, in der IDE des Entwicklers. [Amazon CodeGuru Security](#) kann kritische Fehler, Sicherheitsprobleme und schwer zu findende Bugs während der Anwendungsentwicklung identifizieren und bietet Empfehlungen zur Verbesserung der Codequalität. Auch die Extraktion von Software-Stücklisten (SBOM, Software Bill of Materials) ermöglicht es Ihnen, einen formalen Datensatz zu extrahieren, der die Details und die Beziehungen der verschiedenen Komponenten enthält, die bei der Erstellung Ihrer Software verwendet wurden. Dies ermöglicht Ihnen ein fundiertes Schwachstellenmanagement und die schnelle Identifizierung von Software- oder Komponentenabhängigkeiten sowie Lieferkettenrisiken.

Der Workshop [Security for Developers](#) verwendet AWS-Entwickler-Tools, wie [AWS CodeBuild](#), [AWS CodeCommit](#) und [AWS CodePipeline](#) für die Automatisierung der Veröffentlichungs-Pipeline, die SAST- und DAST-Testmethoden umfasst.

Richten Sie beim Durchlaufen Ihres Softwareentwicklungs-Lebenszyklus einen iterativen Prozess ein, der regelmäßige Anwendungsüberprüfungen mit Ihrem Sicherheitsteam enthält. Aus diesen Sicherheitsüberprüfungen gewonnenes Feedback sollte behandelt und im Rahmen der Bereitschaftsüberprüfung Ihrer Softwareversion validiert werden. Diese Überprüfungen schaffen einen robusten Sicherheitsstatus der Anwendungen und bieten Entwicklern umsetzbares Feedback, um Maßnahmen zum Beheben von Problemen zu ergreifen.

Implementierungsschritte

- Implementieren Sie eine konsistente IDE, eine Codeüberprüfung und CI/CD-Tools mit Sicherheitstests.

- Überlegen Sie, wo im Softwareentwicklungs-Lebenszyklus Pipelines blockiert werden können, anstatt Entwickler darüber zu informieren, dass Probleme behoben werden müssen.
- [Automated Security Helper \(ASH\)](#) ist ein Beispiel für ein Open-Source-Tool für Codesicherheits-Scans.
- Die Durchführung von Tests oder Codeanalysen mithilfe von automatisierten Tools wie [Amazon Q Developer](#), das in Entwickler-IDEs integriert ist, und [Amazon CodeGuru Security](#) für das Scannen von Code beim Commit ermöglicht es Entwicklern, Feedback zur richtigen Zeit zu erhalten.
- Beim Entwickeln mithilfe von AWS Lambda können Sie [Amazon Inspector](#) verwenden, um den Anwendungscode in Ihren Funktionen zu scannen.
- Wenn automatisiertes Testen in CI/CD-Pipelines enthalten ist, sollten Sie ein Ticket-System verwenden, um die Meldung und Behebung von Softwareproblemen nachzuverfolgen.
- Bei Sicherheitstests, die möglicherweise Erkenntnisse liefern, sollten Sie Lösungsanweisungen bereitstellen, damit Entwickler die Codequalität verbessern können.
- Analysieren Sie regelmäßig die Erkenntnisse automatisierter Tools, um die nächste Automatisierung, Entwicklerschulung oder Sensibilisierungskampagne zu planen.
- Für eine SBOM-Extraktion im Rahmen Ihrer CI/CD-Pipelines verwenden Sie [Amazon Inspector SBOM Generator](#), um Software-Stücklisten für Archive, Container-Images, Verzeichnisse, lokale Systeme und kompilierte Go- und Rust-Binärdateien im CycloneDX-SBOM-Format zu erstellen.

Ressourcen

Zugehörige bewährte Methoden:

- [DevOps-Leitfaden: DL.CR.3 Festlegen klarer Abschlusskriterien für Codeaufgaben](#)

Zugehörige Dokumente:

- [Continuous Delivery und Continuous Deployment](#)
- [AWS Kompetenzpartner für DevOps](#)
- [AWS Kompetenzpartner für Sicherheit für Anwendungssicherheit](#)
- [Auswählen eines Well-Architected-CI/CD-Ansatzes](#)
- [Secrets-Erkennung in Amazon CodeGuru Security](#)
- [Erkennungsbibliothek von Amazon CodeGuru Security](#)

- [Beschleunigen von Bereitstellungen in AWS mit effektiver Governance](#)
- [Umgang von AWS mit der Automatisierung sicherer, vollautomatischer Bereitstellungen](#)
- [Erreichen eines effektiven Gleichgewichts zwischen Sicherheit und Geschwindigkeit mithilfe von Amazon CodeGuru Security](#)

Zugehörige Videos:

- [Vollständige Automatisierung: Automatisieren der Pipelines für kontinuierliche Bereitstellung bei Amazon](#)
- [Automatisieren von kontoübergreifenden CI/CD-Pipelines](#)
- [Der Softwareentwicklungsprozess bei Amazon](#)
- [Testen von Software und Systemen bei Amazon](#)

Zugehörige Beispiele:

- [Branchenbewusstsein für Entwickler](#)
- [Automated Security Helper \(ASH\)](#)
- [AWS CodePipeline-Governance – Github](#)

SEC11-BP03 Durchführen regelmäßiger Penetrationstests

Führen Sie regelmäßige Penetrationstests für Ihre Software durch. Dieser Mechanismus hilft bei der Identifizierung potenzieller Softwareprobleme, die bei automatisierten Tests oder einer manuellen Überprüfung des Codes nicht erkannt werden können. Er kann Ihnen außerdem dabei helfen, die Wirksamkeit Ihrer Erkennungskontrollen zu verstehen. Penetrationstests sollen ermitteln, ob es möglich ist, die Software so zu beeinflussen, dass sie sich unerwartet verhält – etwa, indem sie Daten verfügbar macht, die geschützt sein sollten, oder umfassendere Berechtigungen gewährt als erwartet.

Gewünschtes Ergebnis: Penetrationstests werden zur Erkennung und Behandlung sowie zur Validierung der Sicherheitseigenschaften Ihrer Anwendung verwendet. Regelmäßige und geplante Penetrationstests sollten als Teil des Softwareentwicklungs-Lebenszyklus durchgeführt werden. Die aus Penetrationstests gewonnenen Erkenntnisse sollten vor der Veröffentlichung der Software behandelt werden. Analysieren Sie die Erkenntnisse von Penetrationstests, um zu ermitteln, ob es sich um Probleme handelt, die mithilfe von Automatisierung gefunden werden können. Ein

regelmäßiger und wiederholbarer Prozess für Penetrationstests mit einem aktiven Feedback-Mechanismus fließt in die Anweisungen für Entwickler ein und verbessert die Softwarequalität.

Typische Anti-Muster:

- Penetrationstests werden nur für bekannte oder weit verbreitete Sicherheitsprobleme verwendet.
- Penetrationstests werden für Anwendungen ohne abhängige Drittanbieter-Tools und -Bibliotheken durchgeführt.
- Penetrationstests werden nur für Paketsicherheitsprobleme durchgeführt und die implementierte Geschäftslogik wird nicht evaluiert.

Vorteile der Nutzung dieser bewährten Methode:

- Gesteigertes Vertrauen in die Sicherheitseigenschaften der Software vor der Veröffentlichung
- Möglichkeit, bevorzugte Anwendungsmuster zu identifizieren, wodurch die Softwarequalität erhöht wird
- Verbesserte Sicherheitseigenschaften von Software durch eine Feedbackschleife, die früher im Entwicklungszyklus ermittelt, wo Automatisierungen oder zusätzliche Schulungen hilfreich sind

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Penetrationstests sind eine strukturierte Sicherheitstestübung, bei der Szenarien mit geplanten Sicherheitsverstößen zur Erkennung und Behandlung sowie zur Validierung von Sicherheitskontrollen durchgespielt werden. Penetrationstests starten mit einer Erkundung, bei der Daten basierend auf dem aktuellen Design der Anwendung und ihrer Abhängigkeiten gesammelt werden. Eine kuratierte Liste mit sicherheitsspezifischen Testszenarien wird entwickelt und durchlaufen. Der wesentliche Zweck dieser Tests besteht darin, Sicherheitsprobleme in Ihrer Anwendung aufzudecken, die dazu genutzt werden können, unbeabsichtigten Zugriff auf Ihre Umgebung oder unautorisierten Zugriff auf Daten zu erhalten. Sie sollten Penetrationstests durchführen, wenn Sie neue Funktionen einführen oder wenn sich die Funktion oder technische Implementierung Ihrer Anwendung erheblich geändert hat.

Sie sollten in Ihrem Entwicklungslebenszyklus die am besten geeignete Phase bestimmen, um Penetrationstests durchzuführen. Diese Tests sollten so spät stattfinden, dass sich das System nahe am vorgesehenen Veröffentlichungszustand befindet, aber es sollte noch genügend Zeit vorhanden sein, damit Probleme behoben werden können.

Implementierungsschritte

- Implementieren Sie einen strukturierten Prozess für den Umfang der Penetrationstests. Dieser Prozess sollte auf dem [Bedrohungsmodell](#) basieren, um den Kontext zu wahren.
- Bestimmen Sie den geeigneten Zeitpunkt im Entwicklungszyklus zum Durchführen von Penetrationstests. Penetrationstests sollten dann erfolgen, wenn nur noch minimale Anwendungsänderungen zu erwarten sind, aber noch ausreichend Zeit für die Fehlerbehebung übrig ist.
- Schulen Sie Ihre Entwickler hinsichtlich der zu erwartenden Erkenntnisse aus Penetrationstests sowie dahingehend, wie sie Informationen zu deren Behandlung erhalten können.
- Verwenden Sie Tools zur Beschleunigung von Penetrationstests durch Automatisierung gängiger oder wiederholbarer Tests.
- Analysieren Sie Erkenntnisse aus Penetrationstests, um systemische Sicherheitsprobleme zu identifizieren, und verwenden Sie diese Daten, um sie in zusätzliche automatisierte Tests und fortlaufende Entwicklerschulungen einfließen zu lassen.

Ressourcen

Zugehörige bewährte Methoden:

- [SEC11-BP01 Schulen für Anwendungssicherheit](#)
- [SEC11-BP02 Automatisieren des Testens während des Entwicklungs- und Veröffentlichungslebenszyklus](#)

Zugehörige Dokumente:

- [AWS-Penetrationstests](#) enthält ausführliche Informationen zu Penetrationstests in AWS.
- [Beschleunigen von Bereitstellungen in AWS mit effektiver Governance](#)
- [AWS-Kompetenzpartner für Sicherheit](#)
- [Modernisieren Ihrer Penetrationstestarchitektur in AWS Fargate](#)
- [AWS-Fehlerinjektionsservice](#)

Zugehörige Beispiele:

- [Automatisieren von API-Tests mit AWS CodePipeline](#) (GitHub)

- [Automated Security Helper](#) (GitHub)

SEC11-BP04 Durchführen von Codeüberprüfungen

Führen Sie Codeüberprüfungen durch, um die Qualität und Sicherheit der Software während der Entwicklung zu überprüfen. Bei Codeüberprüfungen müssen andere Teammitglieder als der ursprüngliche Verfasser des Codes den Code auf mögliche Probleme und Schwachstellen sowie auf die Einhaltung von Programmierstandards und bewährten Methoden überprüfen. Dieser Prozess hilft, Fehler, Inkonsistenzen und Sicherheitslücken zu erkennen, die der ursprüngliche Entwickler möglicherweise übersehen hat. Verwenden Sie automatisierte Tools zur Unterstützung bei Codeüberprüfungen.

Gewünschtes Ergebnis: Sie beziehen Codeüberprüfungen während der Entwicklung ein, um die Qualität der Software, die gerade geschrieben wird, zu verbessern. Sie nutzen die bei der Codeüberprüfung gewonnenen Erkenntnisse für die Weiterbildung weniger erfahrener Mitglieder des Teams. Sie identifizieren Möglichkeiten zur Automatisierung und unterstützen den Prozess zur Codeüberprüfung mithilfe von automatisierten Tools und Tests.

Typische Anti-Muster:

- Sie überprüfen den Code vor der Bereitstellung nicht.
- Die Person, die den Code geschrieben hat, überprüft den Code selbst.
- Sie verwenden keine Automatisierung und keine Tools zur Unterstützung und Orchestrierung von Codeüberprüfungen.
- Sie schulen die Entwickler vor der Codeüberprüfung nicht in Bezug auf Anwendungssicherheit.

Vorteile der Nutzung dieser bewährten Methode:

- Verbesserte Codequalität
- Erhöhte Konsistenz bei der Codeentwicklung durch erneute Verwendung gängiger Ansätze
- Verringerte Anzahl von Schwierigkeiten, die bei Penetrationstests und in späteren Phasen entdeckt werden
- Verbesserter Wissenstransfer innerhalb des Teams

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Codeüberprüfungen helfen, die Qualität und Sicherheit der Software während der Entwicklung zu überprüfen. Bei manuellen Prüfungen wird der Code von einem anderen Teammitglied als dem ursprünglichen Verfasser des Codes auf mögliche Probleme und Schwachstellen sowie auf die Einhaltung von Programmierstandards und bewährten Methoden überprüft. Dieser Prozess hilft, Fehler, Inkonsistenzen und Sicherheitslücken zu erkennen, die der ursprüngliche Entwickler möglicherweise übersehen hat.

Ziehen Sie die Verwendung von [Amazon CodeGuru Security](#) für automatisierte Codeüberprüfungen in Betracht. CodeGuru Security verwendet Machine Learning und Automated Reasoning, um Ihren Code zu analysieren und potenzielle Sicherheitsschwachstellen und Codierungsprobleme zu identifizieren. Integrieren Sie automatisierte Codeüberprüfungen in Ihre bestehenden Code-Repositorys und CI/CD-Pipelines (Continuous Integration/Continuous Deployment).

Implementierungsschritte

1. Richten Sie einen Prozess zur Codeüberprüfung ein:

- Legen Sie fest, wann Codeüberprüfungen durchgeführt werden sollen, beispielsweise vor der Zusammenführung von Code in den Hauptzweig oder vor der Bereitstellung in der Produktion.
- Legen Sie fest, wer am Prozess zur Codeüberprüfung beteiligt sein sollte, beispielsweise Teammitglieder, erfahrene Entwickler und Sicherheitsexperten.
- Entscheiden Sie sich für die Methode zur Codeüberprüfung, einschließlich des Prozesses und der zu verwendenden Tools.

2. Richten Sie Tools zur Codeüberprüfung ein:

- Evaluieren Sie Tools zur Codeüberprüfung, die den Anforderungen Ihres Teams entsprechen, wie z. B. GitHub Pull Requests oder CodeGuru Security, und wählen Sie geeignete Tools aus.
- Integrieren Sie die ausgewählten Tools in Ihre bestehenden Code-Repositorys und CI/CD-Pipelines.
- Konfigurieren Sie die Tools so, dass die Anforderungen an die Codeüberprüfung (z. B. Mindestanzahl von Prüfern und Genehmigungsregeln) durchgesetzt werden.

3. Definieren Sie eine Checkliste und Richtlinien für die Codeüberprüfung:

- Erstellen Sie eine Checkliste oder Richtlinien für die Codeüberprüfung, aus der/denen hervorgeht, was überprüft werden sollte. Berücksichtigen Sie Faktoren wie die Codequalität, Sicherheitsschwachstellen, die Einhaltung von Programmierstandards und die Leistung.

- Kommunizieren Sie die Checkliste oder die Richtlinien dem Entwicklungsteam und stellen Sie sicher, dass jeder die Erwartungen versteht.
4. Schulen Sie die Entwickler in Bezug auf bewährte Methoden zur Codeüberprüfung:
- Bieten Sie Ihrem Team Schulungen zur Durchführung effektiver Codeüberprüfungen an.
 - Informieren Sie Ihr Team über die Prinzipien der Anwendungssicherheit und allgemeine Schwachstellen, auf die bei Überprüfungen geachtet werden sollte.
 - Fördern Sie den Wissensaustausch und Pair-Programming-Sitzungen zur Weiterbildung von weniger erfahrenen Teammitgliedern.
5. Implementieren Sie den Prozess zur Codeüberprüfung:
- Integrieren Sie den Schritt zur Codeüberprüfung in Ihren Entwicklungsworkflow, beispielsweise das Erstellen einer Pull-Anforderung und das Zuweisen von Prüfern.
 - Verlangen Sie die Überprüfung von Codeänderungen vor der Zusammenführung oder Bereitstellung.
 - Fördern Sie während des Überprüfungsprozesses eine offene Kommunikation und konstruktives Feedback.
6. Überwachen Sie den Prozess und nehmen Sie Verbesserungen vor:
- Überprüfen Sie regelmäßig die Effektivität Ihres Prozesses zur Codeüberprüfung und holen Sie Feedback vom Team ein.
 - Identifizieren Sie Möglichkeiten zur Automatisierung oder zur Verbesserung der Tools, um den Prozess zur Codeüberprüfung zu optimieren.
 - Aktualisieren und optimieren Sie die Checkliste oder die Richtlinien für die Codeüberprüfung kontinuierlich auf der Grundlage von Erkenntnissen und bewährten Methoden der Branche.
7. Fördern Sie eine Kultur der Codeüberprüfung:
- Machen Sie deutlich, wie wichtig Codeüberprüfungen für die Wahrung der Codequalität und -sicherheit sind.
 - Feiern Sie Erfolge und Erkenntnisse aus dem Prozess zur Codeüberprüfung.
 - Fördern Sie ein auf Zusammenarbeit und Unterstützung beruhendes Umfeld, in dem Entwickler ein gutes Gefühl dabei haben, Feedback zu geben und zu empfangen.

Ressourcen

Zugehörige bewährte Methoden:

- [SEC11-BP02 Automatisieren des Testens während des Entwicklungs- und Veröffentlichungslebenszyklus](#)

Zugehörige Dokumente:

- [DevOps-Leitfaden: DL.CR.2 Durchführen von Peer-Reviews bei Codeänderungen](#)
- [Informationen zu Pull-Anforderungen in GitHub](#)

Zugehörige Beispiele:

- [Automatisieren von Codeüberprüfungen mit Amazon CodeGuru Security](#)
- [Automatisieren der Erkennung von Sicherheitsschwachstellen und Bugs in CI/CD-Pipelines mithilfe der CLI von Amazon CodeGuru Security](#)

Zugehörige Videos:

- [Kontinuierliche Verbesserung der Codequalität mit Amazon CodeGuru Security](#)

SEC11-BP05 Zentralisieren von Services für Pakete und Abhängigkeiten

Stellen Sie zentralisierte Services für den Erhalt von Softwarepaketen und anderen Abhängigkeiten für Ihre Teams bereit. Auf diese Weise können Pakete validiert werden, bevor sie in die von Ihnen geschriebene Software integriert werden, und es wird eine Datenquelle für die Analyse der Software bereitgestellt, die in Ihrer Organisation verwendet wird.

Gewünschtes Ergebnis: Sie erstellen Ihren Workload aus externen Softwarepaketen neben dem von Ihnen geschriebenen Code. Dies macht die Implementierung von häufig verwendeten Funktionen wie einem JSON-Parser oder einer Verschlüsselungsbibliothek einfacher. Sie stellen die Quellen für diese Pakete und Abhängigkeiten zentral bereit, sodass Ihr Sicherheitsteam sie vor der Verwendung validieren kann. Sie verwenden diesen Ansatz zusammen mit manuellen und automatisierten Tests, um das Vertrauen in die Qualität der entwickelten Software zu steigern.

Typische Anti-Muster:

- Sie rufen Pakete willkürlich aus Repositorys im Internet ab.
- Sie testen neue Pakete nicht, bevor Sie sie für Entwickler verfügbar machen.

Vorteile der Nutzung dieser bewährten Methode:

- Besseres Verständnis, welche Pakete in der entwickelten Software verwendet werden
- Benachrichtigung von Workload-Teams, wenn ein Paket aktualisiert werden muss (basierend auf dem Verständnis davon, wer was verwendet)
- Geringeres Risiko, dass ein Paket mit Problemen in Ihre Software eingeschlossen wird

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Stellen Sie zentralisierte Services für Pakete und Abhängigkeiten so bereit, dass sie von Entwicklern einfach verwendet werden können. Zentralisierte Services können logisch zentral sein, anstatt als monolithisches System implementiert zu werden. Mit diesem Ansatz können Sie Services anbieten, die die Anforderungen Ihrer Entwickler erfüllen. Sie sollten eine effiziente Methode implementieren, mit der dem Repository im Falle von Updates oder neuen Anforderungen Pakete hinzugefügt werden können. Mithilfe von AWS-Services wie [AWS CodeArtifact](#) oder ähnlichen AWS-Partnerlösungen kann diese Funktion bereitgestellt werden.

Implementierungsschritte

- Implementieren Sie einen logisch zentralisierten Repository-Service, der in allen Umgebungen, in denen die Software entwickelt wird, verfügbar ist.
- Schließen Sie Zugriff auf das Repository als Komponente des AWS-Konto-Vergabeprozesses ein.
- Entwickeln Sie eine Automatisierung zum Testen von Paketen, bevor diese in einem Repository veröffentlicht werden.
- Pflegen Sie Metriken der am häufigsten verwendeten Pakete, Sprachen und Teams mit den häufigsten Änderungen.
- Stellen Sie einen automatisierten Mechanismus für Entwicklerteams bereit, damit sie neue Pakete anfordern und Feedback geben können.
- Scannen Sie regelmäßig Pakete in Ihrem Repository, um die Auswirkungen kürzlich entdeckter Probleme zu identifizieren.

Ressourcen

Zugehörige bewährte Methoden:

- [SEC11-BP02 Automatisieren des Testens während des Entwicklungs- und Veröffentlichungslebenszyklus](#)

Zugehörige Dokumente:

- [DevOps-Anleitung: DL.CS.2 Signieren von Codeartefakten nach jedem Build](#)
- [Ebenen der Lieferkette für Softwareartefakte \(SLSA\)](#)

Zugehörige Beispiele:

- [Beschleunigen von Bereitstellungen in AWS mit effektiver Governance](#)
- [Erhöhen Ihrer Paketsicherheit mit dem Toolkit von CodeArtifact Package Origin Control](#)
- [Pipeline zur Veröffentlichung von Paketen in mehreren Regionen](#) (GitHub)
- [Veröffentlichen von Node.js-Modulen in AWS CodeArtifact mithilfe von AWS CodePipeline](#) (GitHub)
- [Beispiel für eine AWS CDK-Java-CodeArtifact-Pipeline](#) (GitHub)
- [Verteilen privater .NET-NuGet-Pakete mit AWS CodeArtifact](#) (GitHub)

Zugehörige Videos:

- [Proaktive Sicherheit: Überlegungen und Ansätze](#)
- [Die AWS-Philosophie zu Sicherheit \(re:Invent 2017\)](#)
- [Wenn Sicherheit und Dringlichkeit von Bedeutung sind: Umgang mit Log4Shell](#)

SEC11-BP06 Programmatisches Bereitstellen von Software

Führen Sie Bereitstellungen von Software nach Möglichkeit programmatisch durch. Dieser Ansatz verringert die Wahrscheinlichkeit eines Bereitstellungsfehlers oder der Einführung eines unerwarteten Problems aufgrund eines menschlichen Fehlers.

Gewünschtes Ergebnis: Sie testen die Version Ihres Workloads, die Sie auch bereitstellen, und die Bereitstellung erfolgt stets konsistent. Sie externalisieren die Konfiguration Ihres Workloads, wodurch die Bereitstellung in verschiedenen Umgebungen ohne Änderungen leichter möglich wird. Sie nutzen das kryptografische Signieren Ihrer Softwarepakete, um sicherzustellen, dass sich zwischen den Umgebungen nichts ändert.

Typische Anti-Muster:

- Software wird manuell in der Produktion bereitgestellt.
- An Software werden manuell Änderungen vorgenommen, um unterschiedlichen Umgebungen gerecht zu werden.

Vorteile der Nutzung dieser bewährten Methode:

- Gesteigertes Vertrauen in den Prozess der Softwareveröffentlichung
- Verringerter Risiko, dass eine fehlgeschlagene Änderung die Geschäftsfunktionen beeinträchtigt
- Erhöhte Veröffentlichungsfrequenz aufgrund eines geringeren Änderungsrisikos
- Automatische Rollback-Funktion für unerwartete Ereignisse während der Bereitstellung
- Möglichkeit, kryptografisch nachzuweisen, dass es sich bei der getesteten Software um die bereitgestellte Software handelt

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Um eine robuste und zuverlässige Anwendungsinfrastruktur aufrechtzuerhalten, sollten Sie sichere und automatisierte Bereitstellungsverfahren einführen. Dies beinhaltet das Entfernen von permanentem menschlichem Zugriff aus Produktionsumgebungen, die Verwendung von CI/CD-Tools für Bereitstellungen und die Externalisierung von umgebungsspezifischen Konfigurationsdaten. Mit diesem Ansatz können Sie die Sicherheit erhöhen, das Risiko menschlicher Fehler verringern und den Bereitstellungsprozess optimieren.

Sie können die Struktur Ihres AWS-Kontos so aufbauen, dass ein permanenter menschlicher Zugriff auf Produktionsumgebungen nicht mehr möglich ist. Durch diese Vorgehensweise wird das Risiko unbefugter oder versehentlicher Änderungen minimiert und so die Integrität Ihrer Produktionssysteme verbessert. Anstatt direkten menschlichen Zugriff zu gewähren, können Sie CI/CD-Tools wie [AWS CodeBuild](#) und [AWS CodePipeline](#) für Bereitstellungen verwenden. Mit diesen Services können Sie die Erstellungs-, Test- und Bereitstellungsprozesse automatisieren. Es sind dann weniger manuelle Eingriffe notwendig und es kann eine größere Konsistenz erzielt werden.

Um die Sicherheit und Rückverfolgbarkeit weiter zu verbessern, können Sie Ihre Anwendungspakete nach dem Testen signieren und diese Signaturen während der Bereitstellung validieren. Verwenden Sie hierfür kryptografische Tools wie [AWS Signer](#) oder [AWS Key Management Service \(AWS KMS\)](#). Durch das Signieren und Überprüfen von Paketen können Sie sicherstellen, dass Sie nur autorisierten und validierten Code in Ihren Umgebungen bereitstellen.

Darüber hinaus kann Ihr Team Ihren Workload so gestalten, dass umgebungsspezifische Konfigurationsdaten aus einer externen Quelle wie [AWS Systems Manager Parameter Store](#) abgerufen werden. Bei dieser Vorgehensweise wird der Anwendungscode von den Konfigurationsdaten getrennt, sodass Sie Konfigurationen unabhängig verwalten und aktualisieren können, ohne den Anwendungscode selbst ändern zu müssen.

Um die Bereitstellung und Verwaltung der Infrastruktur zu optimieren, sollten Sie die Verwendung von Infrastructure as Code (IaC)-Tools wie [AWS CloudFormation](#) oder [AWS CDK](#) in Betracht ziehen. Sie können diese Tools verwenden, um Ihre Infrastruktur als Code zu definieren und so eine bessere Konsistenz und Wiederholbarkeit von Bereitstellungen in verschiedenen Umgebungen zu erreichen.

Erwägen Sie Canary-Bereitstellungen, um die erfolgreiche Bereitstellung Ihrer Software zu validieren. Bei Canary-Bereitstellungen werden Änderungen an einer Teilmenge der Instances oder Benutzer vorgenommen, bevor sie in der gesamten Produktionsumgebung bereitgestellt werden. Sie können dann die Auswirkungen der Änderungen überwachen und bei Bedarf ein Rollback durchführen, wodurch das Risiko weit verbreiteter Probleme minimiert wird.

Folgen Sie den Empfehlungen im Whitepaper [Ihre AWS-Umgebung mit mehreren Konten organisieren](#). Dieses Whitepaper enthält Anleitungen zum Aufteilen von Umgebungen (z. B. Entwicklung, Staging und Produktion) auf verschiedene AWS-Konten, wodurch eine noch bessere Sicherheit und Isolierung erzielt werden kann.

Implementierungsschritte

1. Richten Sie die AWS-Kontostruktur ein:

- Folgen Sie den Anweisungen im Whitepaper [Ihre AWS-Umgebung mit mehreren Konten organisieren](#), um separate AWS-Konten für verschiedene Umgebungen zu erstellen (z. B. Entwicklung, Staging und Produktion).
- Konfigurieren Sie die entsprechenden Zugriffskontrollen und Berechtigungen für jedes Konto, um den direkten menschlichen Zugriff auf Produktionsumgebungen einzuschränken.

2. Implementieren Sie eine CI/CD-Pipeline:

- Richten Sie eine CI/CD-Pipeline mithilfe von Services wie [AWS CodeBuild](#) und [AWS CodePipeline](#) ein.
- Konfigurieren Sie die Pipeline so, dass Ihr Anwendungscode automatisch erstellt, getestet und in den jeweiligen Umgebungen bereitgestellt wird.
- Integrieren Sie Code-Repositorys in die CI/CD-Pipeline für die Versions- und Codeverwaltung.

3. Signieren und verifizieren Sie Anwendungspakete:

- Verwenden Sie [AWS Signer](#) oder [AWS Key Management Service \(AWS KMS\)](#), um Ihre Anwendungspakete zu signieren, nachdem sie getestet und validiert wurden.
- Konfigurieren Sie den Bereitstellungsprozess so, dass die Signaturen der Anwendungspakete überprüft werden, bevor Sie diese in den Zielumgebungen bereitstellen.

4. Externalisieren Sie Konfigurationsdaten:

- Speichern Sie umgebungsspezifische Konfigurationsdaten in [AWS Systems Manager Parameter Store](#).
- Ändern Sie Ihren Anwendungscode, um während der Bereitstellung oder Laufzeit Konfigurationsdaten aus dem Parameter Store abzurufen.

5. Implementieren Sie Infrastructure as Code (IaC):

- Verwenden Sie IaC-Tools wie [AWS CloudFormation](#) oder [AWS CDK](#), um Ihre Infrastruktur als Code zu definieren und zu verwalten.
- Erstellen Sie CloudFormation-Vorlagen oder CDK-Skripts, um die erforderlichen AWS-Ressourcen für Ihre Anwendung bereitzustellen und zu konfigurieren.
- Integrieren Sie IaC in Ihre CI/CD-Pipeline, um Infrastrukturänderungen zusammen mit Änderungen des Anwendungscode automatisch bereitzustellen.

6. Implementieren Sie Canary-Bereitstellungen:

- Konfigurieren Sie Ihren Bereitstellungsprozess so, dass er Canary-Bereitstellungen unterstützt, bei denen Änderungen auf eine Teilmenge der Instances oder Benutzer angewendet werden, bevor Sie sie in der gesamten Produktionsumgebung bereitstellen.
- Verwenden Sie Services wie [AWS CodeDeploy](#) oder [AWS ECS](#), um Canary-Bereitstellungen zu verwalten und die Auswirkungen von Änderungen zu überwachen.
- Implementieren Sie Rollback-Mechanismen, um zur vorherigen stabilen Version zurückzukehren, falls während der Canary-Bereitstellung Probleme festgestellt werden.

7. Überwachen und prüfen Sie den Prozess:

- Richten Sie Überwachungs- und Protokollierungsmechanismen ein, um Bereitstellungen, die Anwendungsleistung und Infrastrukturänderungen zu verfolgen.
- Verwenden Sie Services wie [Amazon CloudWatch](#) und [AWS CloudTrail](#), um Protokolle und Metriken zu sammeln und zu analysieren.
- Führen Sie Audits und Compliance-Prüfungen ein, um die Einhaltung bewährter Methoden für die Sicherheit und regulatorischer Anforderungen zu überprüfen.

8. Führen Sie kontinuierliche Verbesserungen ein:

- Überprüfen und aktualisieren Sie Ihre Bereitstellungspraktiken regelmäßig und berücksichtigen Sie dabei Feedback und Erfahrungen aus früheren Bereitstellungen.
- Automatisieren Sie den Bereitstellungsprozess so weit wie möglich, um manuelle Eingriffe und potenzielle menschliche Fehler zu reduzieren.
- Arbeiten Sie mit funktionsübergreifenden Teams (beispielsweise den Betriebs- oder Sicherheitsteams) zusammen, um die Bereitstellungspraktiken abzustimmen und kontinuierlich zu verbessern.

Wenn Sie diese Schritte befolgen, können Sie sichere und automatisierte Bereitstellungspraktiken in Ihrer AWS-Umgebung implementieren und so die Sicherheit erhöhen, das Risiko menschlicher Fehler verringern und den Bereitstellungsprozess optimieren.

Ressourcen

Zugehörige bewährte Methoden:

- [SEC11-BP02 Automatisieren des Testens während des Entwicklungs- und Veröffentlichungslebenszyklus](#)
- [DL.CI.2 Automatisches Auslösen von Builds bei Änderungen am Quellcode](#)

Zugehörige Dokumente:

- [Beschleunigen von Bereitstellungen in AWS mit effektiver Governance](#)
- [Automatisierung sicherer, vollautomatischer Bereitstellungen](#)
- [Signieren von Code mithilfe von AWS Certificate Manager Private CA und asymmetrischen Schlüsseln von AWS Key Management Service](#)
- [Signieren von Code: eine Vertrauens- und Integritätskontrolle für AWS Lambda](#)

Zugehörige Videos:

- [Vollständige Automatisierung: Automatisieren der Pipelines für kontinuierliche Bereitstellung bei Amazon](#)

Zugehörige Beispiele:

- [Blau/Grün-Bereitstellungen mit AWS Fargate](#)

SEC11-BP07 Regelmäßiges Bewerten von Sicherheitseigenschaften der Pipelines

Wenden Sie die Prinzipien der Well-Architected-Säule „Sicherheit“ bei Ihren Pipelines an und achten Sie dabei besonders auf die Trennung von Berechtigungen. Bewerten Sie die Sicherheitseigenschaften Ihrer Pipeline-Infrastruktur regelmäßig. Durch eine effektive Verwaltung der Pipeline-Sicherheit können Sie die Sicherheit der Software sicherstellen, die diese Pipelines durchläuft.

Gewünschtes Ergebnis: Für die Pipelines, die Sie zum Entwickeln und Bereitstellen Ihrer Software verwenden, sollten dieselben empfohlenen Praktiken verwendet werden wie für jeden anderen Workload in Ihrer Umgebung. Die Tests, die Sie in Ihren Pipelines implementieren, können von den Teams, die diese verwenden, nicht bearbeitet werden. Mithilfe temporärer Anmeldeinformationen geben Sie den Pipelines nur die erforderlichen Berechtigungen für die Bereitstellungen, die diese durchführen. Sie implementieren Sicherheitsvorkehrungen, um zu verhindern, dass über die Pipelines Bereitstellungen in den falschen Umgebungen erfolgen. Sie konfigurieren Ihre Pipelines so, dass sie einen Status ausgeben, damit die Integrität Ihrer Build-Umgebungen validiert werden kann.

Typische Anti-Muster:

- Sicherheitstests können von Entwicklern umgangen werden.
- Berechtigungen für Bereitstellungs-Pipelines sind übermäßig weit gefasst.
- Pipelines sind nicht für die Validierung von Eingaben konfiguriert.
- Berechtigungen im Zusammenhang mit Ihrer CI/CD-Infrastruktur werden nicht regelmäßig überprüft.
- Langfristige oder fest codierte Anmeldeinformationen werden verwendet.

Vorteile der Nutzung dieser bewährten Methode:

- Größeres Vertrauen in die Integrität der Software, die über die Pipelines entwickelt und bereitgestellt wird.
- Eine Bereitstellung kann angehalten werden, wenn es verdächtige Aktivitäten gibt.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Ihre Bereitstellungs-Pipelines sind eine wichtige Komponente Ihres Softwareentwicklungszyklus und sollten denselben Sicherheitsprinzipien und -praktiken folgen wie jeder andere Workload in Ihrer

Umgebung. Hierzu gehören die Implementierung geeigneter Zugriffskontrollen, die Validierung von Eingaben und die regelmäßige Überprüfung der Berechtigungen im Zusammenhang mit Ihrer CI/CD-Infrastruktur.

Stellen Sie sicher, dass die für die Entwicklung und Bereitstellung von Anwendungen verantwortlichen Teams die Sicherheitstests und -prüfungen, die in Ihren Pipelines implementiert sind, nicht bearbeiten oder umgehen können. Diese Trennung von Bereichen trägt dazu bei, die Integrität Ihrer Entwicklungs- und Bereitstellungsprozesse zu wahren.

Ziehen Sie als Ausgangspunkt die Verwendung der [Referenzarchitektur für AWS-Bereitstellungs-Pipelines](#) in Betracht. Diese Referenzarchitektur bietet eine sichere und skalierbare Grundlage für das Erstellen Ihrer CI/CD-Pipelines in AWS.

Darüber hinaus können Sie Services wie [AWS Identity and Access Management Access Analyzer](#) verwenden, um IAM-Richtlinien mit den geringsten Berechtigungen sowohl für Ihre Pipeline-Berechtigungen als auch als Schritt in Ihrer Pipeline zur Überprüfung von Workload-Berechtigungen zu erstellen. Auf diese Weise können Sie überprüfen, ob Ihre Pipelines und Workloads nur über die erforderlichen Berechtigungen für ihre spezifischen Funktionen verfügen, und so das Risiko unbefugter Zugriffe oder Aktionen verringern.

Implementierungsschritte

- Beginnen Sie mit der [Referenzarchitektur für AWS-Bereitstellungs-Pipelines](#).
- Erwägen Sie, [AWS IAM Access Analyzer](#) zu verwenden, um für die Pipelines programmatisch IAM-Richtlinien mit der geringsten Berechtigung zu erstellen.
- Integrieren Sie Ihre Pipelines mit Überwachung und Benachrichtigung, so dass Sie über unerwartete oder ungewöhnliche Aktivitäten benachrichtigt werden. Bei von AWS verwalteten Services können Sie mithilfe von [Amazon EventBridge](#) Daten an Ziele wie [AWS Lambda](#) oder [Amazon Simple Notification Service](#) (Amazon SNS) umleiten.

Ressourcen

Zugehörige Dokumente:

- [AWS Referenzarchitektur für -Bereitstellungs-Pipelines](#)
- [Überwachung von AWS CodePipeline](#)
- [Bewährte Methoden für die Sicherheit für AWS CodePipeline](#)

Zugehörige Beispiele:

- [DevOps-Überwachungs-Dashboard](#) (GitHub)

SEC11-BP08 Entwickeln eines Programms, das Workload-Teams die Verantwortung für die Sicherheit überträgt

Entwickeln Sie ein Programm oder einen Mechanismus, das bzw. der es Entwicklerteams ermöglicht, Entscheidungen bezüglich der Sicherheit der von ihnen erstellten Software zu treffen. Zwar muss Ihr Sicherheitsteam diese Entscheidungen immer noch während einer Überprüfung validieren, die Übertragung der Sicherheitsverantwortlichkeit auf Entwicklerteams ermöglicht jedoch eine schnellere und sicherere Workload-Entwicklung. Zudem fördert dieser Mechanismus eine Kultur der Verantwortlichkeit, die einen positiven Einfluss auf den Betrieb der von Ihnen entwickelten Systeme hat.

Gewünschtes Ergebnis: Sie haben die Verantwortung für Sicherheit und die Entscheidungsfindung in Ihren Teams verankert. Sie haben Ihre Teams entweder in Bezug auf die richtige Einstellung zur Sicherheit geschult oder die Teams um integrierte oder dem Team zugewiesene Sicherheitsmitarbeiter erweitert. So können Ihre Teams früher im Entwicklungszyklus bessere Sicherheitsentscheidungen treffen.

Typische Anti-Muster:

- Einem Sicherheitsteam werden alle Entscheidungen bezüglich des Sicherheitsdesigns überlassen.
- Sicherheitsanforderungen werden nicht früh genug im Entwicklungsprozess behandelt.
- Es wird kein Feedback von Entwicklern und Sicherheitsexperten zum Betrieb des Programms eingeholt.

Vorteile der Nutzung dieser bewährten Methode:

- Beschleunigung von Sicherheitsüberprüfungen.
- Verringerung von Sicherheitsproblemen, die erst in der Phase der Sicherheitsüberprüfung erkannt werden.
- Verbesserung der gesamten Qualität der Software, die geschrieben wird.
- Möglichkeit, systemische Probleme oder Bereiche mit hoher Wertverbesserung zu identifizieren und zu verstehen.

- Verringerung der erforderlichen Überarbeitung aufgrund von Erkenntnissen aus der Sicherheitsüberprüfung.
- Verbesserung bei der Wahrnehmung der Sicherheitsfunktion.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Niedrig

Implementierungsleitfaden

Beginnen Sie mit der Anleitung unter [SEC11-BP01 Schulen für Anwendungssicherheit](#). Bestimmen Sie danach das Betriebsmodell für das Programm, von dem Sie denken, dass es am besten für Ihre Organisation geeignet ist. Die zwei Hauptmuster sind, Entwickler zu schulen oder Sicherheitsexperten in Entwicklungsteams einzubetten. Nachdem Sie sich für eine anfängliche Verfahrensweise entschieden haben, sollten Sie eine Pilotphase mit einem einzelnen Team oder einer kleinen Gruppe von Workload-Teams durchführen, um sich zu vergewissern, dass das Modell für Ihre Organisation funktioniert. Unterstützung der Führungskräfte aus dem Entwicklungs- und Sicherheitsbereich der Organisation hilft bei der Durchführung und trägt zum Erfolg des Programms bei. Bei der Entwicklung dieses Programms ist es wichtig, Metriken auszuwählen, die Aufschluss über den Wert des Programms geben. Zu erfahren, wie AWS mit diesem Problem umgegangen ist, ist eine gute Lernerfahrung. Diese bewährte Methode konzentriert sich auf die Veränderung und Kultur der Organisation. Die von Ihnen eingesetzten Tools sollten die Zusammenarbeit zwischen der Entwickler- und Sicherheits-Community unterstützen.

Implementierungsschritte

- Beginnen Sie damit, Ihre Entwickler im Bereich der Anwendungssicherheit zu schulen.
- Schaffen Sie eine Community und ein Onboarding-Programm zur Schulung von Entwicklern.
- Geben Sie dem Programm einen Namen. Häufig wird etwas wie Guardians, Champions oder Advocates verwendet.
- Bestimmen Sie das Modell, das verwendet werden soll: Schulen Sie Entwickler, betten Sie Sicherheitstechniker ein oder verwenden Sie andere verwandte Sicherheitsrollen.
- Identifizieren Sie Projektsponsoren aus dem Sicherheits- und Entwicklungsbereich sowie gegebenenfalls aus anderen relevanten Gruppen.
- Verfolgen Sie Metriken für die Anzahl der am Programm beteiligten Personen, die für Überprüfungen erforderliche Zeit und das Feedback von Entwicklern und Sicherheitsexperten. Nutzen Sie diese Metriken für Verbesserungen.

Ressourcen

Zugehörige bewährte Methoden:

- [SEC11-BP01 Schulen für Anwendungssicherheit](#)
- [SEC11-BP02 Automatisieren des Testens während des Entwicklungs- und Veröffentlichungslebenszyklus](#)

Zugehörige Dokumente:

- [Konzepte für Bedrohungsmodellierung](#)
- [Betrachtung der Cloud-Sicherheits-Governance](#)
- [Wie AWS das Security-Guardians-Programm erstellt hat, einen Mechanismus zur Verteilung der Verantwortung in Bezug auf Sicherheit](#)
- [Erstellen eines Security-Guardians-Programms zur Verteilung der Verantwortung in Bezug auf Sicherheit](#)

Zugehörige Videos:

- [Proaktive Sicherheit: Überlegungen und Ansätze](#)
- [Tipps zu AppSec-Tools und Sicherheitskultur von AWS und Toyota Motor North America](#)

Zuverlässigkeit

Die Säule „Zuverlässigkeit“ umfasst die Fähigkeit einer Workload, die beabsichtigte Funktion erwartungsgemäß korrekt und konsistent auszuführen. Verbindliche Anleitungen zur Implementierung finden Sie im [Whitepaper „Säule der Zuverlässigkeit“](#).

Bereiche für bewährte Methoden

- [Grundlagen](#)
- [Workload-Architektur](#)
- [Änderungsmanagement](#)
- [Fehlerverwaltung](#)

Grundlagen

Fragen

- [REL 1. Wie werden Service Quotas und Serviceeinschränkungen verwaltet?](#)
- [REL 2. Wie wird die Netzwerktopologie geplant?](#)

REL 1. Wie werden Service Quotas und Serviceeinschränkungen verwaltet?

Für cloudbasierte Workload-Architekturen gibt es Service Quotas (auch als „Servicebeschränkungen“ bezeichnet). Mit diesen Kontingenten soll verhindert werden, dass versehentlich mehr Ressourcen bereitgestellt werden, als Sie benötigen, und es sollen die Anforderungsraten bei API-Vorgängen begrenzt werden, um Services vor Missbrauch zu schützen. Es gibt auch Einschränkungen für Ressourcen, z. B. im Bezug auf die Rate, mit der Bits über ein Glasfaserkabel übertragen werden können, oder die Menge an Speicherplatz auf einer physischen Festplatte.

Best Practices

- [REL01-BP01 Kenntnis von Service Quotas und Einschränkungen](#)
- [REL01-BP02 Verwalten von Service Quotas für mehrere Konten und Regionen](#)
- [REL01-BP03 Berücksichtigen von festen Servicekontingenten und Einschränkungen durch die Architektur](#)
- [REL01-BP04 Kontingente überwachen und verwalten](#)
- [REL01-BP05 Automatisieren der Kontingentverwaltung](#)
- [REL01-BP06 Sicherstellen eines ausreichenden Spielraums zwischen den aktuellen Kontingenten und der maximalen Nutzung, damit ein Failover möglich ist](#)

REL01-BP01 Kenntnis von Service Quotas und Einschränkungen

Sie wissen über die Standardkontingente Bescheid und verwalten Anfragen zur Kontingenterhöhung für Ihre Workload-Architektur. Außerdem wissen Sie, welche Ressourceneinschränkungen, z. B. bezüglich Datenträgern oder Netzwerken, potenziell große Auswirkungen haben.

Gewünschtes Ergebnis: Kunden können eine Beeinträchtigung oder Unterbrechung ihrer Services in ihrer AWS-Konten verhindern, indem sie geeignete Richtlinien für die Überwachung von Schlüsselkennzahlen, Infrastrukturüberprüfungen und Automatisierungsschritte zur Behebung von Problemen einführen, um sicherzustellen, dass Service Quotas und Einschränkungen, die eine Beeinträchtigung oder Unterbrechung der Dienste verursachen könnten, nicht erreicht werden.

Typische Anti-Muster:

- Bereitstellung eines Workloads ohne Kenntnis der harten oder weichen Quoten und ihrer Grenzen für die verwendeten Services.
- Bereitstellung eines Ersatz-Workloads, ohne die erforderlichen Quoten zu analysieren und neu zu konfigurieren oder den Support im Voraus zu kontaktieren.
- Annehmen, dass Cloud-Services keine Grenzen haben und die Service ohne Berücksichtigung von Tarifen, Grenzen, Zählungen und Mengen genutzt werden können.
- Annehmen, dass die Quoten automatisch erhöht werden.
- Keine Kenntnis des Prozesses und der Zeitleiste von Quotenanforderungen.
- Annehmen, dass das Standardkontingent für Cloud-Services für jeden Service im regionalen Vergleich identisch ist.
- Annehmen, dass die Servicebeschränkungen überschritten werden können und die Systeme automatisch skalieren oder das Limit über die Beschränkungen der Ressource hinaus erhöhen.
- Die Anwendung nicht bei Spitzenbelastungen testen, um die Auslastung der Ressourcen zu strapazieren.
- Bereitstellung der Ressource ohne Analyse der erforderlichen Ressourcengröße.
- Überbereitstellung von Kapazitäten durch Auswahl von Ressourcentypen, die weit über den tatsächlichen Bedarf oder die erwarteten Spitzen hinausgehen.
- Keine Bewertung des Kapazitätsbedarfs für neue Datenverkehrsniveaus im Vorfeld eines neuen Kundenereignisses und keine Einführung einer neuen Technologie.

Vorteile der Nutzung dieser bewährten Methode: Durch die Überwachung und automatisierte Verwaltung von Service Quotas und Ressourcenbeschränkungen können Ausfälle proaktiv reduziert werden. Änderungen in den Datenverkehrsmustern für den Service eines Kunden können zu einer Unterbrechung oder Verschlechterung führen, wenn die bewährten Methoden nicht befolgt werden. Durch die Überwachung und Verwaltung dieser Werte in allen Regionen und auf allen Konten können die Anwendungen bei ungünstigen oder ungeplanten Ereignissen besser geschützt werden.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Service Quotas ist ein AWS-Service, mit dem Sie Ihre Kontingente für über 250 AWS-Services von einem Standort aus verwalten können. Neben der Suche nach den Kontingentwerten können Sie

auch Kontingenterhöhungen über die Service Quotas-Konsole oder über das AWS SDK anfordern und nachverfolgen. AWS Trusted Advisor bietet eine Service Quotas-Prüfung, die Ihre Nutzung und Ihre Kontingente für bestimmte Aspekte einiger Services anzeigt. Die Standardkontingente pro Service finden Sie ebenfalls in der AWS-Dokumentation für den jeweiligen Service (weitere Informationen finden Sie unter [Amazon VPC-Kontingente](#)).

Einige Servicelimits wie Ratenlimits für gedrosselte APIs werden innerhalb des Amazon API Gateway selbst festgelegt. Dazu wird ein Nutzungsplan konfiguriert. Andere Limits, die für ihre jeweiligen Services konfiguriert werden, sind bereitgestellte IOPS, zugewiesener Amazon RDS-Speicher und Amazon EBS-Volume-Zuweisungen. Amazon Elastic Compute Cloud verfügt über ein eigenes Service Limits-Dashboard, mit dem Sie Ihre Limits für Instances, Amazon Elastic Block Store und Elastic IP-Adressen verwalten können. Wenn Sie einen Anwendungsfall haben, bei dem sich Service Quotas auf die Leistung Ihrer Anwendung auswirken und eine Anpassung an Ihre Anforderungen nicht möglich ist, wenden Sie sich an den Support, um zu ermitteln, ob es Lösungen gibt.

Service Quotas können spezifisch für eine Region oder auch global sein. Ein AWS-Service, der sein Kontingent erreicht hat, verhält sich bei normaler Nutzung nicht wie erwartet und es kann zu Unterbrechungen oder Beeinträchtigungen des Services kommen. Beispielsweise begrenzen Service Quotas die Anzahl der DL Amazon EC2, die in einer Region genutzt werden können, und dieses Limit kann während eines Ereignisses zur Skalierung des Datenverkehrs durch Auto Scaling-Gruppen (ASG) erreicht werden.

Service Quotas für die einzelnen Konten sollten regelmäßig auf ihre Nutzung hin überprüft werden, um festzustellen, welche Servicelimits für das jeweilige Konto angemessen sind. Diese Service Quotas dienen als betrieblicher Integritätsschutz, um zu verhindern, dass versehentlich mehr Ressourcen bereitgestellt werden, als Sie benötigen. Sie begrenzen auch die Anfrageraten bei API-Operationen, um Services vor Missbrauch zu schützen.

Serviceeinschränkungen und Service Quotas unterscheiden sich voneinander.

Serviceeinschränkungen stellen die Limits einer bestimmten Ressource dar, wie sie durch diesen Ressourcentyp definiert sind. Dabei kann es sich um die Speicherkapazität (z. B. hat gp2 eine Größengrenzung von 1 GB bis 16 TB) oder den Festplattendurchsatz handeln. Es ist von entscheidender Bedeutung, dass die Beschränkung eines Ressourcentyps konstruiert und ständig auf eine Nutzung geprüft wird, durch die das Limit erreicht werden könnte. Wenn eine Beschränkung unerwartet erreicht wird, können die Anwendungen oder Services des Kontos beeinträchtigt oder unterbrochen werden.

Wenn es einen Anwendungsfall gibt, bei dem sich Service Quotas auf die Leistung Ihrer Anwendung auswirken und eine Anpassung an die Anforderungen nicht möglich ist, wenden Sie sich an

den Support, um zu ermitteln, ob es Lösungen gibt. Weitere Einzelheiten zur Anpassung fester Kontingente finden Sie unter [REL01-BP03 Berücksichtigen von festen Servicekontingenten und Einschränkungen durch die Architektur](#).

Es gibt eine Reihe von AWS-Services und -Tools, die Sie bei der Überwachung und Verwaltung von Service Quotas unterstützen. Der Service und die Tools sollten genutzt werden, um automatische oder manuelle Überprüfungen der Kontingente zu ermöglichen.

- AWS Trusted Advisor bietet eine Service Quotas-Prüfung, die Ihre Nutzung und Ihre Kontingente für einige Aspekte einiger Services anzeigt. Es kann dabei helfen, Services zu identifizieren, die ihr Kontingent fast erreicht haben.
- AWS-Managementkonsole bietet Methoden, um Service-Quota-Werte für Services anzuzeigen, zu verwalten, neue Kontingente anzufordern, den Status von Kontingentanforderungen zu überwachen und den Verlauf von Kontingenten anzuzeigen.
- AWS CLI und CDKs bieten programmatische Methoden zur automatischen Verwaltung und Überwachung von Service Quotas und deren Nutzung.

Implementierungsschritte

Für Service Quotas:

- [Überprüfen Sie AWS Service Quotas](#).
- Bestimmen Sie die verwendeten Services (wie IAM Access Analyzer), damit Sie Ihre bestehenden Service Quotas kennen. Es gibt etwa 250 AWS-Services, für die Service Quotas gelten. Bestimmen Sie dann den spezifischen Service-Quota-Namen, der für jedes Konto und jede Region verwendet werden kann. Pro Region gibt es etwa 3 000 Service-Quota-Namen.
- Ergänzen Sie diese Kontingentanalyse um AWS Config, um alle [AWS-Ressourcen](#) zu finden, die in Ihren AWS-Konten verwendet werden.
- Bestimmen Sie anhand von [AWS CloudFormation-Daten](#) Ihre verwendeten AWS-Ressourcen. Sehen Sie sich die Ressourcen an, die in der AWS-Managementkonsole oder über den Befehl [list-stack-resources](#) AWS CLIBefehl erstellt wurden. Sie können zudem Ressourcen anzeigen, die für die Bereitstellung in der Vorlage selbst konfiguriert sind.
- Ermitteln Sie alle für die Workload erforderlichen Services durch Untersuchung des Bereitstellungscode.
- Ermitteln Sie die geltenden Service Quotas. Nutzen Sie die programmgesteuert über Trusted Advisor und Service Quotas zugänglichen Informationen. .

- Richten Sie eine automatisierte Überwachungsmethode ein (siehe [REL01-BP02 Verwalten von Service Quotas für mehrere Konten und Regionen](#) und [REL01-BP04 Kontingente überwachen und verwalten](#)), um zu warnen und zu informieren, wenn die Service Quotas fast erschöpft sind oder ihr Limit erreicht haben.
- Richten Sie eine automatische, programmatische Methode ein, um zu überprüfen, ob ein Service Quota in einer Region, aber nicht in anderen Regionen desselben Kontos geändert wurde (siehe [REL01-BP02 Verwalten von Service Quotas für mehrere Konten und Regionen](#) und [REL01-BP04 Kontingente überwachen und verwalten](#)).
- Automatisieren Sie das Scannen von Anwendungsprotokollen und Metriken, um festzustellen, ob Fehler beim Kontingent oder bei Serviceeinschränkungen vorliegen. Falls Fehler vorhanden sind, senden Sie Warnmeldungen an das Überwachungssystem.
- Führen Sie technische Verfahren zur Berechnung der erforderlichen Kontingentänderung ein (siehe [REL01-BP05 Automatisieren der Kontingentverwaltung](#)), wenn festgestellt wird, dass für bestimmte Services größere Kontingente erforderlich sind.
- Erstellen Sie einen Bereitstellungs- und Genehmigungs-Workflow, um Änderungen am Service Quota anzufordern. Dies sollte einen Ausnahme-Workflow für den Fall umfassen, dass ein Antrag abgelehnt oder nur teilweise genehmigt wird.
- Erstellen Sie eine technische Methode zur Überprüfung von Service Quotas vor der Bereitstellung und Nutzung neuer AWS-Services, und zwar vor dem Rollout in Produktionsumgebungen oder Umgebungen mit Last (z. B. Lasttestkonto).

Bei Serviceeinschränkungen:

- Führen Sie Überwachungs- und Messmethoden ein, um auf Ressourcen aufmerksam zu machen, die ihre Ressourceneinschränkungen fast erreicht haben. Nutzen Sie CloudWatch gegebenenfalls für Metriken oder Protokollüberwachung.
- Legen Sie Warnschwellenwerte für jede Ressource fest, die eine für die Anwendung oder das System bedeutsame Einschränkung hat.
- Erstellen Sie Verfahren für die Verwaltung von Workflows und Infrastrukturen, um den Ressourcentyp zu ändern, wenn die Nutzungseinschränkung fast erreicht ist. Dieser Workflow sollte Lasttests beinhalten, um zu überprüfen, ob der neue Typ der richtige Ressourcentyp mit den neuen Einschränkungen ist.
- Migrieren Sie die identifizierte Ressource unter Verwendung bestehender Verfahren und Prozesse auf den empfohlenen neuen Ressourcentyp.

Ressourcen

Zugehörige bewährte Methoden:

- [REL01-BP02 Verwalten von Service Quotas für mehrere Konten und Regionen](#)
- [REL01-BP03 Berücksichtigen von festen Servicekontingenten und Einschränkungen durch die Architektur](#)
- [REL01-BP04 Kontingente überwachen und verwalten](#)
- [REL01-BP05 Automatisieren der Kontingentverwaltung](#)
- [REL01-BP06 Sicherstellen eines ausreichenden Spielraums zwischen den aktuellen Kontingenten und der maximalen Nutzung, damit ein Failover möglich ist](#)
- [REL03-BP01 Wählen Sie, wie Sie Ihre Arbeitslast segmentieren möchten](#)
- [REL10-BP01 Bereitstellen des Workloads an mehreren Standorten](#)
- [REL11-BP01 Überwachen aller Komponenten der Workload auf Fehler](#)
- [REL11-BP03 Automatisieren der Reparatur auf allen Ebenen](#)
- [REL12-BP04 Testen der Ausfallsicherheit mit Chaos-Engineering](#)

Zugehörige Dokumente:

- [AWS Well-Architected Framework's Reliability Pillar: Availability](#)
- [AWS Service Quotas \(früher als Service Limits bezeichnet\)](#)
- [Bewährte AWS Trusted Advisor-Prüfungsmethoden \(siehe Abschnitt „Servicelimits“\)](#)
- [AWS Limit Monitor in AWS Answers](#)
- [Amazon EC2 Service Limits](#)
- [Was ist Service Quotas?](#)
- [How to Request Quota Increase](#) (So beantragen Sie eine Kontingenterhöhung)
- [Service endpoints and quotas](#) (Service-Endpunkte und -Quoten)
- [Service Quotas — Benutzerhandbuch](#)
- [Quota Monitor für AWS](#)
- [AWS Fault Isolation Boundaries](#)
- [Availability with redundancy](#)
- [AWS für Daten](#)

- [What is Continuous Integration?](#)
- [What is Continuous Delivery?](#)
- [APN-Partner: Partner, die Sie bei der Konfigurationsverwaltung unterstützen können](#)
- [Managing the account lifecycle in account-per-tenant SaaS environments on AWS](#)
- [Managing and monitoring API throttling in your workloads](#)
- [View AWS Trusted Advisor recommendations at scale with AWS Organizations](#)
- [Automating Service Limit Increases and Enterprise Support with AWS Control Tower](#)

Zugehörige Videos:

- [AWS Live re:Inforce 2019 – Service Quotas](#)
- [View and Manage Quotas for AWS Services Using Service Quotas](#)
- [AWS IAM Quotas Demo](#)

Zugehörige Tools:

- [Amazon CodeGuru Reviewer](#)
- [AWS CodeDeploy](#)
- [AWS CloudTrail](#)
- [Amazon CloudWatch](#)
- [Amazon EventBridge](#)
- [Amazon DevOps Guru](#)
- [AWS Config](#)
- [AWS Trusted Advisor](#)
- [AWS CDK](#)
- [AWS Systems Manager](#)
- [AWS Marketplace](#)

REL01-BP02 Verwalten von Service Quotas für mehrere Konten und Regionen

Wenn Sie mehrere Konten oder Regionen verwenden, fordern Sie die entsprechenden Kontingente in allen Umgebungen an, in denen die Produktions-Workloads ausgeführt werden.

Gewünschtes Ergebnis: Services und Anwendungen sollten bei Konfigurationen, die sich über Konten oder Regionen erstrecken oder die über ein Ausfallsicherheitsdesign mit Zonen-, Regions- oder Konto-Failover verfügen, nicht von der Erschöpfung des Service Quota betroffen sein.

Typische Anti-Muster:

- Es wird zugelassen, dass die Ressourcennutzung in einer Isolationsregion zunimmt, ohne dass es einen Mechanismus zur Aufrechterhaltung der Kapazität in den anderen Zonen gibt.
- Alle Kontingente werden manuell und in jeder Isolationsregion einzeln festgelegt.
- Nichtberücksichtigung der Auswirkungen von Ausfallsicherheitsarchitekturen (wie aktiv oder passiv) auf den künftigen Kontingentbedarf bei einer Verschlechterung in der nicht primären Region.
- Keine regelmäßige Bewertung der Kontingente und Durchführung der erforderlichen Änderungen in jeder Region und jedem Konto, in dem die Workload ausgeführt wird.
- Keine Nutzung von [Vorlagen für Kontingentanforderungen](#), um Erhöhungen für mehrere Regionen und Konten zu beantragen.
- Keine Aktualisierung von Service Quotas, weil man fälschlicherweise davon ausgeht, dass eine Erhöhung der Kontingente Kosten nach sich zieht, wie z. B. Anforderungen von Rechenkapazitäten.

Vorteile der Einführung dieser bewährten Methode: Überprüfen, ob Sie Ihre aktuelle Last in sekundären Regionen oder Konten bewältigen können, falls regionale Services nicht mehr verfügbar sind. Dies kann dazu beitragen, die Anzahl von Fehlern oder Verschlechterungen zu verringern, die beim Verlust von Regionen auftreten.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Service Quotas werden pro Konto aufgezeichnet. Sofern nicht anders angegeben, gilt jedes Kontingent für eine bestimmte AWS-Region. Zusätzlich zu den Produktionsumgebungen verwalten Sie auch Kontingente in allen anwendbaren Nicht-Produktionsumgebungen, damit Tests und Entwicklung nicht behindert werden. Die Aufrechterhaltung eines hohen Maßes an Ausfallsicherheit setzt voraus, dass die Service Quotas ständig überprüft werden (entweder automatisch oder manuell).

Da durch die Implementierung von Designs mit den Ansätzen Aktiv/Aktiv, Aktiv/Passiv – Hot, Aktiv/Passiv – Cold und Aktiv/Passiv – Pilot Light immer mehr Workloads auf die Regionen verteilt werden,

ist es wichtig, alle Kontingente für Regionen und Konten zu kennen. Frühere Datenverkehrsmuster sind nicht immer ein guter Indikator dafür, ob das Service Quota korrekt eingestellt ist.

Ebenso wichtig ist, dass das Namenslimit für das Service Quota nicht immer für alle Regionen gleich ist. In einer Region kann der Wert fünf sein, in einer anderen zehn. Die Verwaltung dieser Kontingente muss sich auf dieselben Services, Konten und Regionen erstrecken, um eine gleichmäßige Ausfallsicherheit unter Last zu gewährleisten.

Stimmen Sie alle Unterschiede zwischen den Service Quotas in den verschiedenen Regionen (aktive oder passive Region) ab und schaffen Sie Prozesse, um diese Unterschiede kontinuierlich abzugleichen. Die Testpläne für passive Regions-Failover sind selten auf die aktive Spitzenkapazität skaliert, was bedeutet, dass es im Ernstfall oder bei Tabletop-Übungen nicht gelingen kann, Unterschiede bei den Service Quotas zwischen den Regionen festzustellen und die korrekten Limits einzuhalten.

Service-Quota-Abweichung, d. h. der Umstand, dass die Service-Quota-Limits für ein bestimmtes benanntes Kontingent in einer Region und nicht in allen Regionen geändert werden, müssen unbedingt verfolgt und bewertet werden. Es sollte erwogen werden, die Kontingente in Regionen mit Datenverkehr oder potenziellem Datenverkehr zu ändern.

- Wählen Sie relevante Konten und Regionen anhand von Serviceanforderungen, regulatorischen Anforderungen sowie Anforderungen für die Latenz und die Notfallwiederherstellung aus.
- Ermitteln Sie Service Quotas für alle relevanten Konten, Regionen und Availability Zones. Die Limits gelten für ein Konto und eine Region. Diese Werte sollten auf Unterschiede hin verglichen werden.

Implementierungsschritte

- Überprüfen Sie die Service Quotas-Werte, die über eine Risikostufe der Nutzung hinausgehen. AWS Trusted Advisor bietet Warnungen bei Überschreitung der Schwellenwerte von 80 % und 90 %.
- Überprüfen Sie die Werte für Service Quotas in allen passiven Regionen (in einem Aktiv/Passiv-Design). Stellen Sie sicher, dass die Last in den sekundären Regionen bei einem Ausfall in der primären Region erfolgreich ausgeführt werden kann.
- Automatisieren Sie die Bewertung, ob es zu einer Verschiebung der Service Quotas zwischen den Regionen desselben Kontos gekommen ist, und handeln Sie entsprechend, um die Limits zu ändern.

- Wenn die Organisationseinheiten (OU) des Kunden in der unterstützten Weise strukturiert sind, sollten die Vorlagen für Service Quotas aktualisiert werden, um Änderungen an Kontingenten widerzuspiegeln, die auf mehrere Regionen und Konten angewendet werden sollen.
- Erstellen Sie eine Vorlage und weisen Sie der Kontingentänderung Regionen zu.
- Überprüfen Sie alle bestehenden Vorlagen für Service Quotas auf erforderliche Änderungen (Region, Limits und Konten).

Ressourcen

Zugehörige bewährte Methoden:

- [REL01-BP01 Kenntnis von Service Quotas und Einschränkungen](#)
- [REL01-BP03 Berücksichtigen von festen Servicekontingenten und Einschränkungen durch die Architektur](#)
- [REL01-BP04 Kontingente überwachen und verwalten](#)
- [REL01-BP05 Automatisieren der Kontingentverwaltung](#)
- [REL01-BP06 Sicherstellen eines ausreichenden Spielraums zwischen den aktuellen Kontingenten und der maximalen Nutzung, damit ein Failover möglich ist](#)
- [REL03-BP01 Wählen Sie, wie Sie Ihre Arbeitslast segmentieren möchten](#)
- [REL10-BP01 Bereitstellen des Workloads an mehreren Standorten](#)
- [REL11-BP01 Überwachen aller Komponenten der Workload auf Fehler](#)
- [REL11-BP03 Automatisieren der Reparatur auf allen Ebenen](#)
- [REL12-BP04 Testen der Ausfallsicherheit mit Chaos-Engineering](#)

Zugehörige Dokumente:

- [AWS Well-Architected Framework's Reliability Pillar: Availability](#)
- [AWS Service Quotas \(früher als Service Limits bezeichnet\)](#)
- [AWS Trusted Advisor Bewährte -Prüfungsmethoden \(siehe Abschnitt „Servicelimits“\)](#)
- [AWS Limit Monitor in AWS Answers](#)
- [Amazon EC2 Service Limits](#)
- [Was ist Service Quotas?](#)
- [How to Request Quota Increase](#)

- [Service endpoints and quotas](#)
- [Service Quotas-Benutzerhandbuch](#)
- [Quota Monitor for AWS](#)
- [AWS Fault Isolation Boundaries](#)
- [Availability with redundancy](#)
- [AWS für Data](#)
- [What is Continuous Integration?](#)
- [What is Continuous Delivery?](#)
- [APN-Partner: Partner, die Sie bei der Konfigurationsverwaltung unterstützen können](#)
- [Managing the account lifecycle in account-per-tenant SaaS environments on AWS](#)
- [Managing and monitoring API throttling in your workloads](#)
- [View AWS Trusted Advisor recommendations at scale with AWS Organizations](#)
- [Automating Service Limit Increases and Enterprise Support with AWS Control Tower](#)

Zugehörige Videos:

- [AWS Live re:Inforce 2019 – Service Quotas](#)
- [View and Manage Quotas for AWS Services Using Service Quotas](#)
- [AWS IAM Quotas Demo](#)

Zugehörige Services:

- [Amazon CodeGuru Reviewer](#)
- [AWS CodeDeploy](#)
- [AWS CloudTrail](#)
- [Amazon CloudWatch](#)
- [Amazon EventBridge](#)
- [Amazon DevOps Guru](#)
- [AWS Config](#)
- [AWS Trusted Advisor](#)
- [AWS CDK](#)
- [AWS Systems Manager](#)

- [AWS Marketplace](#)

REL01-BP03 Berücksichtigen von festen Servicekontingenten und Einschränkungen durch die Architektur

Achten Sie auf nicht änderbare Service Quotas, Servicebeschränkungen und physische Ressourcen-Limits. Entwerfen Sie Architekturen für Anwendungen und Services, um zu verhindern, dass sich diese Limits auf die Zuverlässigkeit auswirken.

Beispiele hierfür sind die Netzwerkbandbreite, die Datengröße beim Aufrufen von Serverless-Funktionen, die Drosselung der Burst-Rate eines API-Gateways und die gleichzeitig mit einer Datenbank verbundenen Benutzer.

Gewünschtes Ergebnis: Die Anwendung oder der Service erbringt unter normalen Bedingungen und bei hohem Datenverkehr die erwartete Leistung. Sie wurden so konzipiert, dass sie innerhalb der für diese Ressource festgelegten Beschränkungen oder Service-Kontingente arbeiten.

Typische Anti-Muster:

- Auswahl eines Designs, das eine Ressource eines Service verwendet, ohne zu wissen, dass es Design-Einschränkungen gibt, die dazu führen, dass dieses Design beim Skalieren versagt.
- Sie führen ein Benchmarking durch, das unrealistisch ist und mit dem während der Tests die festen Kontingente für den Service erreicht werden. Sie führen beispielsweise Tests mit einem Burst-Limit durch, diese aber für einen längeren Zeitraum.
- Sie wählen ein Design aus, das nicht skaliert oder geändert werden kann, wenn feste Service-Kontingente überschritten werden müssen. Ein Beispiel wäre ein SQS-Payload von 256 KB.
- Die Überwachungsfunktion wurde nicht zur Überwachung und Benachrichtigung von/für Schwellenwerte/n für Service-Kontingente entwickelt und implementiert, die bei hohem Datenverkehr gefährdet sein könnten.

Vorteile der Nutzung dieser bewährten Methode: Es wird sichergestellt, dass die Anwendung unter allen prognostizierten Last-Levels der Services ohne Unterbrechung oder Beeinträchtigung läuft.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Im Gegensatz zu Soft-Kontingenten für Services oder Ressourcen, die durch Einheiten mit höherer Kapazität ersetzt werden können, können feste Kontingente für AWS-Services nicht geändert

werden. Das bedeutet, dass alle AWS-Services dieser Art auf potenzielle harte Kapazitätsgrenzen geprüft werden müssen, wenn sie in einer Anwendung zum Einsatz kommen.

Feste Beschränkungen werden in der Service Quotas-Konsole angezeigt. Wenn in den Spalten ADJUSTABLE = No angezeigt wird, gibt es eine feste Beschränkung für den Service. Auch auf einigen Konfigurationsseiten für Ressourcen werden feste Beschränkungen angezeigt. Für Lambda gibt es zum Beispiel bestimmte feste Beschränkungen, die nicht angepasst werden können.

Wenn Sie beispielsweise eine Python-Anwendung entwerfen, die in einer Lambda-Funktion ausgeführt werden soll, sollte die Anwendung daraufhin geprüft werden, ob die Möglichkeit besteht, dass Lambda länger als 15 Minuten läuft. Wenn die Codeausführung länger als dieses Service-Kontingent dauert, müssen alternative Technologien oder Designs in Betracht gezogen werden. Wird diese Beschränkung nach der Bereitstellung in der Produktion erreicht, wird die Anwendung beeinträchtigt und gestört, bis sie wiederhergestellt werden kann. Im Gegensatz zu Soft-Kontingenten gibt es keine Möglichkeit, diese Beschränkungen zu ändern – selbst wenn ein Ereignis des Schweregrads 1 eintritt.

Sobald die Anwendung in einer Testumgebung bereitgestellt wurde, sollten Strategien eingesetzt werden, um herauszufinden, ob feste Beschränkungen erreicht werden könnten. Stresstests, Lasttests und Chaostests sollten Teil des Einführungstestplans sein.

Implementierungsschritte

- Sehen Sie sich die vollständige Liste der AWS-Services an. Diese können Sie in der Entwurfsphase der Anwendung verwenden.
- Sehen Sie sich die Soft-Kontingentbeschränkungen und Hard-Kontingentbeschränkungen der Services an. Nicht alle Beschränkungen werden in der Service Quotas-Konsole angezeigt. Einige Services [zeigen die Beschränkungen an anderen Stellen an](#).
- Prüfen Sie bei der Entwicklung Ihrer Anwendung die geschäftlichen und technologischen Faktoren Ihres Workloads, wie z. B. Geschäftsergebnisse, Anwendungsfälle, abhängige Systeme, Verfügbarkeitsziele und Objekte für die Notfallwiederherstellung. Lassen Sie sich von Ihren geschäftlichen und technologischen Faktoren leiten, um das richtige verteilte System für Ihren Workload zu finden.
- Analysieren Sie die Last des Services über Regionen und Konten hinweg. Viele feste Beschränkungen für Services basieren auf Regionen. Einige Beschränkungen sind jedoch kontobasiert.
- Analysieren Sie die Architekturen zur Ausfallsicherheit der Ressourcen bei einem zonenbezogenen Fehler und einem Fehler in einer Region. Bei der Entwicklung von Multi-Regionen-Designs

mit Aktiv/Aktiv-, Aktiv/Passiv-Hot-, Aktiv/Passiv-Cold- und Aktiv/Passiv-Pilot-Light-Ansätzen werden diese Fehlerfälle eine höhere Auslastung verursachen. Dies schafft einen potenziellen Anwendungsfall für feste Beschränkungen.

Ressourcen

Zugehörige bewährte Methoden:

- [REL01-BP01 Kenntnis von Service Quotas und Einschränkungen](#)
- [REL01-BP02 Verwalten von Service Quotas für mehrere Konten und Regionen](#)
- [REL01-BP04 Kontingente überwachen und verwalten](#)
- [REL01-BP05 Automatisieren der Kontingentverwaltung](#)
- [REL01-BP06 Sicherstellen eines ausreichenden Spielraums zwischen den aktuellen Kontingenten und der maximalen Nutzung, damit ein Failover möglich ist](#)
- [REL03-BP01 Wählen Sie, wie Sie Ihre Arbeitslast segmentieren möchten](#)
- [REL10-BP01 Bereitstellen des Workloads an mehreren Standorten](#)
- [REL11-BP01 Überwachen aller Komponenten der Workload auf Fehler](#)
- [REL11-BP03 Automatisieren der Reparatur auf allen Ebenen](#)
- [REL12-BP04 Testen der Ausfallsicherheit mit Chaos-Engineering](#)

Zugehörige Dokumente:

- [AWS Well-Architected Framework's Reliability Pillar: Availability](#)
- [AWS Service Quotas \(früher als Service Limits bezeichnet\)](#)
- [AWS Trusted Advisor Bewährte -Prüfungsmethoden \(siehe Abschnitt „Servicelimits“\)](#)
- [AWS Limit Monitor in AWS Answers](#)
- [Amazon EC2 Service Limits](#)
- [Was ist Service Quotas?](#)
- [How to Request Quota Increase](#)
- [Service endpoints and quotas](#)
- [Service Quotas-Benutzerhandbuch](#)
- [Quota Monitor for AWS](#)

- [AWS Fault Isolation Boundaries](#)
- [Availability with redundancy](#)
- [AWS für Data](#)
- [What is Continuous Integration?](#)
- [What is Continuous Delivery?](#)
- [APN-Partner: Partner, die Sie bei der Konfigurationsverwaltung unterstützen können](#)
- [Managing the account lifecycle in account-per-tenant SaaS environments on AWS](#)
- [Managing and monitoring API throttling in your workloads](#)
- [View AWS Trusted Advisor recommendations at scale with AWS Organizations](#)
- [Automating Service Limit Increases and Enterprise Support with AWS Control Tower](#)
- [Aktionen, Ressourcen und Bedingungsschlüssel für Service Quotas](#)

Zugehörige Videos:

- [AWS Live re:Inforce 2019 – Service Quotas](#)
- [View and Manage Quotas for AWS Services Using Service Quotas](#)
- [AWS IAM Quotas Demo](#)
- [AWS re:Invent 2018: Close Loops and Opening Minds: How to Take Control of Systems, Big and Small](#)

Zugehörige Tools:

- [AWS CodeDeploy](#)
- [AWS CloudTrail](#)
- [Amazon CloudWatch](#)
- [Amazon EventBridge](#)
- [Amazon DevOps Guru](#)
- [AWS Config](#)
- [AWS Trusted Advisor](#)
- [AWS CDK](#)
- [AWS Systems Manager](#)

- [AWS Marketplace](#)

REL01-BP04 Kontingente überwachen und verwalten

Überprüfen Sie die potenzielle Nutzung und erhöhen Sie Ihre Kontingente entsprechend, um einen geplanten Nutzungsanstieg zu ermöglichen.

Gewünschtes Ergebnis: Es wurden aktive und automatisierte Verwaltungs- und Überwachungssysteme bereitgestellt. Diese operativen Lösungen reagieren, wenn die Schwellenwerte für die Kontingentnutzung fast erreicht werden. Sie lösen die Situation durch die proaktiven Änderungen des Kontingents.

Typische Anti-Muster:

- Keine Konfigurationsüberwachung zur Prüfung von Schwellenwerten für das Service-Kontingent.
- Keine Konfigurationsüberwachung für feste Beschränkungen, auch wenn diese Werte nicht geändert werden können.
- Sie gehen davon aus, dass eine Änderung des Soft-Kontingents direkt stattfindet oder nur wenig Zeit erfordert.
- Es werden Warnungen für den Fall konfiguriert, dass Servicekontingente erreicht werden, aber es gibt keinen Prozess für die Reaktion auf eine entsprechende Warnung.
- Nur Alarmer für Dienste konfigurieren, die von AWS Service Quotas unterstützt werden, und keine Überwachung anderer AWS Dienste.
- Keine Berücksichtigung der Verwaltung von Kontingenten für die Ausfallsicherheit mehrerer Regionen, wie z. B. Aktiv/Aktiv-, Aktiv/Passiv-Hot-, Aktiv/Passiv-Cold- und Aktiv/Passiv-Pilot-Light-Ansätze.
- Keine Bewertung der Kontingentunterschiede zwischen den Regionen.
- Keine Bewertung des Bedarfs in jeder Region für eine bestimmte Kontingenterhöhung.
- Keine Nutzung von [Vorlagen für die Verwaltung von Kontingenten für mehrere Regionen](#).

Vorteile der Einführung dieser bewährten Methode: Durch die automatische Verfolgung der AWS Service Quotas und die Überwachung Ihrer Nutzung anhand dieser Kontingente können Sie erkennen, wann Sie sich einer Kontingentbegrenzung nähern. Sie können diese Überwachungsdaten außerdem nutzen, um Verschlechterungen aufgrund einer Kontingentausschöpfung zu begrenzen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Bei unterstützten Services können Sie Ihre Kontingente überwachen, indem Sie verschiedene Services zur Bewertung und anschließenden Versendung von Warnungen konfigurieren. Auf diese Weise können Sie die Nutzung überwachen und werden auf sich nähernde Kontingentgrenzen aufmerksam gemacht. Diese Alarme können von Lambda-Funktionen AWS Config CloudWatch, Amazon oder von aufgerufen werden. AWS Trusted Advisor Sie können auch Metrikfilter für CloudWatch Protokolle verwenden, um Muster in Protokollen zu suchen und zu extrahieren, um festzustellen, ob sich die Nutzung den Kontingentschwellenwerten nähert.

Implementierungsschritte

Für die Überwachung:

- Erfassen Sie den aktuellen Ressourcenverbrauch (z. B. Buckets oder Instances). Verwenden Sie API Servicebetriebe wie Amazon EC2 DescribeInstancesAPI, um den aktuellen Ressourcenverbrauch zu ermitteln.
- Erfassen Sie Ihre aktuellen Kontingente, die für die Services wesentlich und anwendbar sind. Nutzen Sie dazu:
 - AWS Service Quotas
 - AWS Trusted Advisor
 - AWS Dokumentation
 - AWS dienstspezifische Seiten
 - AWS Command Line Interface (AWS CLI)
 - AWS Cloud Development Kit (AWS CDK)
- Verwenden Sie AWS Service Quotas, einen AWS Service, mit dem Sie Ihre Kontingente für über 250 AWS Dienste von einem Standort aus verwalten können.
- Verwenden Sie Trusted Advisor Servicelimits, um Ihre aktuellen Servicelimits bei verschiedenen Schwellenwerten zu überwachen.
- Prüfen Sie anhand der Historie der Servicekontingenten (Konsole oder AWS CLI), ob regionale Erhöhungen vorliegen.
- Vergleichen Sie die Änderungen der Service-Kontingente in jeder Region und jedem Konto, um bei Bedarf auszugleichen.

Für die Verwaltung:

- Automatisiert: Richten Sie eine AWS Config benutzerdefinierte Regel ein, um Servicekontingente in verschiedenen Regionen zu scannen und Unterschiede zu vergleichen.
- Automatisiert: Richten Sie eine geplante Lambda-Funktion ein, um Service-Kontingente in den Regionen zu scannen und Abweichungen zu ermitteln.
- Manuell: Scannen Sie die Service-Kontingente über AWS CLI API, oder die AWS Konsole, um Servicekontingente in verschiedenen Regionen zu scannen und auf Unterschiede zu vergleichen. Erstellen Sie einen Bericht zu den Abweichungen.
- Wenn Abweichungen in den Kontingenten zwischen den Regionen festgestellt werden, fordern Sie bei Bedarf eine Kontingentänderung an.
- Überprüfen Sie das Ergebnis aller Anforderungen.

Ressourcen

Zugehörige bewährte Methoden:

- [REL01-BP01 Kenntnis von Service Quotas und Einschränkungen](#)
- [REL01-BP02 Verwalten von Service Quotas für mehrere Konten und Regionen](#)
- [REL01-BP03 Berücksichtigen von festen Servicekontingenten und Einschränkungen durch die Architektur](#)
- [REL01-BP05 Automatisieren der Kontingentverwaltung](#)
- [REL01-BP06 Sicherstellen eines ausreichenden Spielraums zwischen den aktuellen Kontingenten und der maximalen Nutzung, damit ein Failover möglich ist](#)
- [REL03-BP01 Wählen Sie, wie Sie Ihre Arbeitslast segmentieren möchten](#)
- [REL10-BP01 Bereitstellen des Workloads an mehreren Standorten](#)
- [REL11-BP01 Überwachen aller Komponenten der Workload auf Fehler](#)
- [REL11-BP03 Automatisieren der Reparatur auf allen Ebenen](#)
- [REL12-BP04 Testen der Ausfallsicherheit mit Chaos-Engineering](#)

Zugehörige Dokumente:

- [AWS Die Zuverlässigkeitssäule von Well-Architected Framework: Verfügbarkeit](#)
- [AWS Service Quotas \(früher als Service Limits bezeichnet\)](#)
- [AWS Trusted Advisor Prüfungen bewährter Verfahren \(siehe Abschnitt Service Limits\)](#)

- [AWS Beschränken Sie den Monitor auf AWS Antworten](#)
- [EC2Amazon-Servicebeschränkungen](#)
- [Was ist Service Quotas?](#)
- [How to Request Quota Increase](#)
- [Service endpoints and quotas](#)
- [Service Quotas-Benutzerhandbuch](#)
- [Kontingentmonitor für AWS](#)
- [AWS Grenzen der Fehlerisolierung](#)
- [Availability with redundancy](#)
- [AWS für Daten](#)
- [What is Continuous Integration?](#)
- [What is Continuous Delivery?](#)
- [APNPartner: Partner, die beim Konfigurationsmanagement helfen können](#)
- [Verwaltung des Kontolebenszyklus in account-per-tenant SaaS-Umgebungen auf AWS](#)
- [Verwaltung und Überwachung der API Drosselung Ihrer Workloads](#)
- [Sehen Sie sich AWS Trusted Advisor Empfehlungen in großem Umfang an mit AWS Organizations](#)
- [Automatisierung von Service-Limit-Erhöhungen und Unternehmenssupport mit AWS Control Tower](#)
- [Aktionen, Ressourcen und Bedingungsschlüssel für Service Quotas](#)

Zugehörige Videos:

- [AWS Live re:inForce 2019 — Service Quotas](#)
- [Kontingente für AWS Dienste mithilfe von Service Quotas anzeigen und verwalten](#)
- [AWS IAMDemo der Kontingente](#)
- [AWS re:Invent 2018: Kreisläufe schließen und neue Denkansätze eröffnen: Wie man die Kontrolle über große und kleine Systeme übernimmt](#)

Zugehörige Tools:

- [AWS CodeDeploy](#)
- [AWS CloudTrail](#)

- [Amazon CloudWatch](#)
- [Amazon EventBridge](#)
- [DevOpsAmazon-Guru](#)
- [AWS Config](#)
- [AWS Trusted Advisor](#)
- [AWS CDK](#)
- [AWS Systems Manager](#)
- [AWS Marketplace](#)

REL01-BP05 Automatisieren der Kontingentverwaltung

Service Quotas, in AWS-Services auch als Einschränkungen bezeichnet, sind die Höchstwerte für die Ressourcen in Ihrem AWS-Konto. Jeder AWS-Service definiert eine Reihe von Kontingenten und ihre Standardwerte. Um Ihrem Workload Zugriff auf alle benötigten Ressourcen zu gewähren, müssen Sie möglicherweise Ihre Service Quota-Werte erhöhen.

Ein Anstieg des Verbrauchs von AWS-Ressourcen durch Workloads kann die Workload-Stabilität gefährden und sich negativ auf die Benutzererfahrung auswirken, wenn Kontingente überschritten werden. Implementieren Sie Tools, um benachrichtigt zu werden, wenn Ihr Workload die Grenzwerte erreicht, und ziehen Sie die automatische Erstellung von Anfragen zur Kontingenterhöhung in Betracht.

Gewünschtes Ergebnis: Die Kontingente werden für die Workloads, die in den einzelnen AWS-Konto-Regionen ausgeführt werden, angemessen konfiguriert.

Typische Anti-Muster:

- Sie betrachten die Kontingente nicht und passen sie nicht entsprechend an die Workload-Anforderungen an.
- Sie verfolgen Kontingente und Nutzung mithilfe von Methoden nach, die möglicherweise veraltet sind, z. B. mit Tabellenkalkulationen.
- Sie aktualisieren die Servicegrenzen nur in regelmäßigen Abständen.
- Ihre Organisation besitzt keine operativen Prozesse für die Überprüfung vorhandener Kontingente und die Anforderung von Service-Quota-Erhöhungen, wenn notwendig.

Vorteile der Nutzung dieser bewährten Methode:

- **Verbesserte Workload-Resilienz:** Sie verhindern Fehler, die durch Überschreitung von AWS-Ressourcenkontingenten verursacht werden.
- **Vereinfachte Notfallwiederherstellung:** Sie können automatische Mechanismen zur Kontingentverwaltung, die während der DR-Einrichtung in der primären Region eingerichtet wurden, in einer anderen AWS-Region wiederverwenden.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Zeigen Sie aktuelle Kontingente an und verfolgen Sie den laufenden Kontingentverbrauch mithilfe von Mechanismen wie der Konsole in AWS Service Quotas, AWS Command Line Interface (AWS CLI), und AWS SDKs nach. Sie können auch Ihre Datenbanken für die Konfigurationsverwaltung (Configuration Management Databases, CMDB) und Ihre Systeme für die Verwaltung von IT-Services (IT Service Management, ITSM) in die APIs von AWS Service Quotas integrieren.

Generieren Sie automatische Benachrichtigungen, wenn die Kontingentnutzung die definierten Schwellenwerte erreicht, und definieren Sie einen Prozess für das Einreichen von Anfragen zur Erhöhung des Kontingents, wenn Sie Benachrichtigungen erhalten. Wenn die zugrundeliegenden Workloads kritisch für Ihr Unternehmen sind, können Sie die Anfragen zur Quotenerhöhung automatisieren. Testen Sie die Automatisierung jedoch sorgfältig, um zu vermeiden, dass sie außer Kontrolle gerät, indem beispielsweise eine wachsende Feedbackschleife entsteht.

Kleinere Kontingenterhöhungen werden häufig automatisch genehmigt. Größere Kontingentanfragen müssen möglicherweise manuell vom AWS-Support bearbeitet werden, was zusätzliche Zeit in Anspruch nehmen kann, um sie zu prüfen und zu verarbeiten. Planen Sie zusätzliche Zeit ein, um mehrere Anfragen oder Anfragen für größere Erhöhungen zu bearbeiten.

Implementierungsschritte

- Implementieren Sie die automatische Überwachung von Service Quotas und geben Sie Warnmeldungen aus, wenn sich die Ressourcenauslastung Ihrer Workloads den Kontingentgrenzen nähert. [Quota Monitor](#) für AWS kann beispielsweise die automatische Überwachung von Service Quotas ermöglichen. Dieses Tool ist mit AWS Organizations integriert und wird mithilfe von CloudFormation StackSets bereitgestellt, sodass neue Konten ab Erstellung automatisch überwacht werden.
- Verwenden Sie Funktionen wie [Service-Quotas-Anforderungsvorlagen](#) oder [AWS Control Tower](#), um die Einrichtung von Service Quotas für neue Konten zu vereinfachen.

- Erstellen Sie Dashboards über Ihre aktuelle Nutzung von Service Quotas in allen AWS-Konten und AWS-Regionen und verweisen Sie auf diese, wenn notwendig, um eine Überschreitung Ihrer Kontingente zu verhindern. Das Dashboard [Trusted Advisor Organizational \(TAO\)](#) ist Teil von [Cloud Intelligence Dashboards](#) und ermöglicht Ihnen die schnelle Erstellung dieser Dashboards.
- Verfolgen Sie Anfragen zur Erhöhung von Servicebegrenzungen nach. Konsolidierte Einblicke aus mehreren Konten ([Consolidated Insights from Multiple Accounts \(CIMA\)](#)) bieten Ihnen einen Überblick über alle Ihre Anforderungen auf Organisationsebene.
- Testen Sie die Generierung von Warnmeldungen und die Automatisierung von Anforderungen zur Quotenerhöhung, indem Sie niedrigere Kontingentschwellenwerte für Konten festlegen, die nicht in der Produktion verwendet werden. Führen Sie diese Tests nicht in einem Produktionskonto durch.

Ressourcen

Zugehörige bewährte Methoden:

- [OPS10-BP07 Automatisieren von Reaktionen auf Ereignisse](#)

Zugehörige Dokumente:

- [APN-Partner: Partner, die Sie bei der Konfigurationsverwaltung unterstützen können](#)
- [AWS Marketplace: CMDB-Produkte zur Nachverfolgung von Limits](#)
- [AWS AWS Service Quotas \(früher als Service Limits bezeichnet\)](#)
- [Bewährte AWS Trusted Advisor-Prüfungsmethoden \(siehe Abschnitt „Servicelimits“\)](#)
- [Lösung für die Kontingentüberwachung n AWS – AWS-Lösung](#)
- [Was ist Service Quotas?](#)
- [Was sind Anforderungsvorlagen für Service Quotas?](#)

Zugehörige Videos:

- [AWS Live re:Inforce 2019 – Service Quotas](#)
- [Automatisierung der Erhöhungen von Servicegrenzen und des Unternehmenssupport mit AWS Control Tower](#)

Zugehörige Tools:

- [Quota Monitor for AWS](#)

REL01-BP06 Sicherstellen eines ausreichenden Spielraums zwischen den aktuellen Kontingenten und der maximalen Nutzung, damit ein Failover möglich ist

In diesem Artikel wird erläutert, wie Sie den Abstand zwischen dem Ressourcenkontingent und Ihrer Nutzung beibehalten können und wie Ihr Unternehmen davon profitieren kann. Wenn Sie eine Ressource nicht mehr nutzen, wird diese Ressource möglicherweise noch auf ein Kontingent angerechnet. Dies kann zu einer fehlgeschlagenen oder nicht mehr erreichbaren Ressource führen. Überprüfen Sie, ob Ihre Kontingente die Überschneidung von ausgefallenen oder nicht zugreifbaren Ressourcen und deren Ersatz abdecken. Bei der Berechnung dieser Lücke sollten Sie Anwendungsfälle wie Netzwerkfehler, Fehler in der Availability Zone oder Fehler in einer Region berücksichtigen.

Gewünschtes Ergebnis: Kleine oder große Fehler bei Ressourcen oder der Ressourcenzugänglichkeit können innerhalb der aktuellen Service-Schwellenwerte abgedeckt werden. Zonenfehler, Netzwerkfehler oder sogar regionale Fehler wurden bei der Ressourcenplanung berücksichtigt.

Typische Anti-Muster:

- Es werden Servicekontingente auf Grundlage des aktuellen Bedarfs eingerichtet, ohne dass Failover-Szenarien berücksichtigt werden.
- Keine Berücksichtigung des Prinzips der statischen Stabilität bei der Berechnung des Spitzenkontingents für einen Service.
- Keine Berücksichtigung des Potenzials nicht zugreifbarer Ressourcen bei der Berechnung des für jede Region benötigten Gesamtkontingents.
- Keine Berücksichtigung der AWS-Grenzen für die Fehlerisolierung bei einigen Services und ihrer potenziell anormalen Nutzungsmuster.

Vorteile der Nutzung dieser bewährten Methode: Wenn die Verfügbarkeit von Anwendungen durch eine Service-Störung beeinträchtigt wird, bietet Ihnen die Cloud die Möglichkeit zur Implementierung von Strategien zur Abschwächung dieser Ereignisse oder der Wiederherstellung. Zu solchen Strategien gehört oft die Erstellung zusätzlicher Ressourcen, um ausgefallene oder unzugängliche Ressourcen zu ersetzen. Ihre Kontingent-Strategie muss diese Failover-Bedingungen berücksichtigen und würde nicht zu einer zusätzlichen Verschlechterung aufgrund des Erreichens von Service-Beschränkungen führen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Berücksichtigen Sie bei der Bewertung der Kontingente auch Failover-Fälle, die aufgrund einer Verschlechterung auftreten können. Die folgenden Arten von Failover-Fällen sollten in Betracht gezogen werden:

- Eine VPC, die gestört oder auf die nicht zugreifbar ist.
- Ein Subnetz, auf das nicht mehr zugegriffen werden kann.
- Eine Availability Zone wurde so stark beeinträchtigt, dass die Erreichbarkeit vieler Ressourcen beeinträchtigt ist.
- Verschiedene Netzwerk-Routen oder Ingress- und Egress-Punkte sind blockiert oder verändert.
- Eine Region ist so stark gestört, dass die Erreichbarkeit vieler Ressourcen beeinträchtigt ist.
- Es gibt mehrere Ressourcen, aber nicht alle sind von einem Fehler in einer Region oder einer Availability Zone betroffen.

Die Entscheidung für einen Failover ist für jede Situation und jeden Kunden individuell, da die Auswirkungen auf den Geschäftsbetrieb sehr unterschiedlich sein können. Wenn Sie sich jedoch operativ für einen Failover von Anwendungen oder Services entscheiden, müssen Sie sich vor dem Ereignis mit der Kapazitätsplanung der Ressourcen am Failover-Standort und den entsprechenden Kontingenten befassen.

Überprüfen Sie die Service-Kontingente für jeden Service und berücksichtigen Sie dabei die möglichen Spitzenwerte. Diese Spitzen können mit Ressourcen zusammenhängen, die über Netzwerkproblemen oder Berechtigungen zwar noch aktiv, aber nicht erreichbar sind. Nicht beendete aktive Ressourcen werden weiterhin auf das Kontingent des Service angerechnet.

Implementierungsschritte

- Vergewissern Sie sich, dass zwischen Ihrem Service-Kontingent und Ihrer maximalen Nutzung genügend Spielraum besteht, um einen Failover oder den Verlust der Erreichbarkeit aufzufangen.
- Ermitteln Sie die Servicekontingente unter Berücksichtigung von Bereitstellungsmustern, der Verfügbarkeitsanforderungen und des Nutzungsanstiegs.
- Fordern Sie bei Bedarf Kontingenterhöhungen an. Planen Sie den erforderlichen Zeitraum bis zur Bewilligung von Kontingenterhöhungen.
- Bestimmen Sie Ihre Anforderungen an die Zuverlässigkeit (Anzahl der Neunen).

- Legen Sie Fehlerszenarien fest (z. B. Verlust einer Komponente, Availability Zone oder Region).
- Führen Sie eine Bereitstellungsmethode ein (z. B. Canary, Blau/Grün-Bereitstellung, Rot/Schwarz-Bereitstellung oder schrittweise).
- Berücksichtigen Sie einen angemessenen Puffer (z. B. 15 %) in aktuellen Limits.
- Berücksichtigen Sie gegebenenfalls Berechnungen zur statischen Stabilität (zonenbezogen und regional).
- Planen Sie den Nutzungsanstieg (z. B. durch Überwachen des Nutzungstrends).
- Berücksichtigen Sie die Auswirkungen der statischen Stabilität für Ihre kritischsten Workloads. Bewerten Sie Ressourcen entsprechend eines statisch stabilen Systems in allen Regionen und Availability Zones.
- Ziehen Sie den Einsatz von On-Demand-Kapazitätsreservierungen in Betracht, um vor einem Failover Kapazitäten zu reservieren. Diese Strategie kann während kritischer Geschäftszeiten sinnvoll sein, um potenzielle Risiken bei der Beschaffung der richtigen Menge und Art von Ressourcen während eines Failovers zu verringern.

Ressourcen

Zugehörige bewährte Methoden:

- [REL01-BP01 Kenntnis von Service Quotas und Einschränkungen](#)
- [REL01-BP02 Verwalten von Service Quotas für mehrere Konten und Regionen](#)
- [REL01-BP03 Berücksichtigen von festen Servicekontingenten und Einschränkungen durch die Architektur](#)
- [REL01-BP04 Kontingente überwachen und verwalten](#)
- [REL01-BP05 Automatisieren der Kontingentverwaltung](#)
- [REL03-BP01 Wählen Sie, wie Sie Ihre Arbeitslast segmentieren möchten](#)
- [REL10-BP01 Bereitstellen des Workloads an mehreren Standorten](#)
- [REL11-BP01 Überwachen aller Komponenten der Workload auf Fehler](#)
- [REL11-BP03 Automatisieren der Reparatur auf allen Ebenen](#)
- [REL12-BP04 Testen der Ausfallsicherheit mit Chaos-Engineering](#)

Zugehörige Dokumente:

- [AWS Well-Architected Framework's Reliability Pillar: Availability](#)

- [AWS Service Quotas \(früher als Service Limits bezeichnet\)](#)
- [AWS Trusted Advisor Bewährte -Prüfungsmethoden \(siehe Abschnitt „Servicelimits\)](#)
- [AWS Limit Monitor in AWS Answers](#)
- [Amazon EC2 Service Limits](#)
- [Was ist Service Quotas?](#)
- [How to Request Quota Increase](#)
- [Service endpoints and quotas](#)
- [Service Quotas-Benutzerhandbuch](#)
- [Quota Monitor for AWS](#)
- [AWS Fault Isolation Boundaries](#)
- [Availability with redundancy](#)
- [AWS für Data](#)
- [What is Continuous Integration?](#)
- [What is Continuous Delivery?](#)
- [APN-Partner: Partner, die Sie bei der Konfigurationsverwaltung unterstützen können](#)
- [Managing the account lifecycle in account-per-tenant SaaS environments on AWS](#)
- [Managing and monitoring API throttling in your workloads](#)
- [View AWS Trusted Advisor recommendations at scale with AWS Organizations](#)
- [Automating Service Limit Increases and Enterprise Support with AWS Control Tower](#)
- [Aktionen, Ressourcen und Bedingungsschlüssel für Service Quotas](#)

Zugehörige Videos:

- [AWS Live re:Inforce 2019 – Service Quotas](#)
- [View and Manage Quotas for AWS Services Using Service Quotas](#)
- [AWS IAM Quotas Demo](#)
- [AWS re:Invent 2018: Close Loops and Opening Minds: How to Take Control of Systems, Big and Small](#)

Zugehörige Tools:

- [AWS CodeDeploy](#)

- [AWS CloudTrail](#)
- [Amazon CloudWatch](#)
- [Amazon EventBridge](#)
- [Amazon DevOps Guru](#)
- [AWS Config](#)
- [AWS Trusted Advisor](#)
- [AWS CDK](#)
- [AWS Systems Manager](#)
- [AWS Marketplace](#)

REL 2. Wie wird die Netzwerktopologie geplant?

Workloads existieren oft in mehreren Umgebungen. Dazu gehören mehrere Cloud-Umgebungen (sowohl öffentlich zugänglich als auch privat) und möglicherweise Ihre bestehende Rechenzentrumsinfrastruktur. Die Pläne müssen Netzwerkaspekte wie systeminterne und systemübergreifende Konnektivität, Verwaltung öffentlicher IP-Adressen, Verwaltung privater IP-Adressen und Auflösung von Domainnamen beinhalten.

Best Practices

- [REL02-BP01 Bereitstellen einer hochverfügbaren Netzwerkkonnektivität für öffentliche Endpunkte der Workload](#)
- [REL02-BP02 Bereitstellen redundanter Konnektivität zwischen privaten Netzwerken in der Cloud und in On-Premises-Umgebungen](#)
- [REL02-BP03 Berücksichtigen von Erweiterungen und Verfügbarkeit bei der Zuweisung von IP-Adressen für Subnetze](#)
- [REL02-BP04 Vorziehen von Hub-and-Spoke-Topologien gegenüber M-zu-N-Netzen](#)
- [REL02-BP05 Erzwingen von sich nicht überschneidenden privaten IP-Adressbereichen in allen privaten Adressbereichen, in denen eine Verbindung besteht](#)

REL02-BP01 Bereitstellen einer hochverfügbaren Netzwerkkonnektivität für öffentliche Endpunkte der Workload

Der Aufbau einer hochverfügbaren Netzwerkkonnektivität zu öffentlichen Endpunkten Ihres Workloads kann Ihnen helfen, Ausfallzeiten aufgrund von Konnektivitätsverlusten zu reduzieren und

die Verfügbarkeit und SLA Ihres Workloads zu verbessern. Verwenden Sie dazu hochverfügbares DNS, Content Delivery Networks (CDNs), API-Gateways, Load-Balancing oder Reverse-Proxies.

Gewünschtes Ergebnis: Es ist von entscheidender Bedeutung, eine hochverfügbare Netzwerkkonnektivität für Ihre öffentlichen Endpunkte zu planen, aufzubauen und in Betrieb zu nehmen. Wenn Ihr Workload aufgrund eines Konnektivitätsverlustes nicht mehr erreichbar ist, sehen Ihre Kunden Ihr System als ausgefallen an – selbst wenn Ihr Workload läuft und verfügbar ist. Durch die Kombination einer hochverfügbaren und stabilen Netzwerkkonnektivität für die öffentlichen Endpunkte Ihres Workloads mit einer stabilen Architektur für Ihren Workload selbst können Sie Ihren Kunden die bestmögliche Verfügbarkeit und das bestmögliche Serviceniveau bieten.

AWS Global Accelerator, Amazon CloudFront, Amazon API Gateway, AWS Lambda-Funktions-URLs, AWS AppSync-APIs und Elastic Load Balancing (ELB) bieten alle hochverfügbare öffentliche Endpunkte. Amazon Route 53 bietet einen hochverfügbaren DNS-Service für die Auflösung von Domain-Namen, um sicherzustellen, dass die Adressen Ihrer öffentlichen Endpunkte aufgelöst werden können.

Sie können außerdem AWS Marketplace-Software-Appliances für das Load-Balancing und für Proxys nutzen.

Typische Anti-Muster:

- Entwurf eines hochverfügbaren Workloads, ohne eine DNS- und Netzwerkkonnektivität mit hoher Verfügbarkeit einzuplanen.
- Verwendung öffentlicher Internetadressen auf einzelnen Instances oder Containern und Verwalten der Konnektivität zu diesen per DNS.
- Verwendung von IP-Adressen anstelle von Domain-Namen zur Lokalisierung von Services.
- Keine Tests von Szenarien, in denen die Konnektivität zu Ihren öffentlichen Endpunkten verloren geht.
- Keine Analyse des Bedarfs für den Netzwerkdurchsatz und die Verteilungsmuster im Netzwerk.
- Keine Tests und Planungen für Szenarien, in denen die Internet-Netzwerkkonnektivität zu Ihren öffentlichen Endpunkten der Workloads unterbrochen werden könnte.
- Bereitstellen von Inhalten (z. B. Webseiten, statische Komponenten oder Mediendateien) für ein großes geografisches Gebiet ohne Verwendung eines Content-Delivery-Networks.
- Keine Planung für Distributed Denial of Service (DDoS)-Angriffe. Bei DDoS-Angriffen besteht die Gefahr, dass der legitime Datenverkehr unterbrochen wird und die Verfügbarkeit für Ihre Benutzer sinkt.

Vorteile der Nutzung dieser bewährten Methode: Die Planung einer hochverfügbaren und stabilen Netzwerkkonnektivität stellt sicher, dass Ihr Workload für Ihre Benutzer zugreifbar und verfügbar ist.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Das Wichtigste beim Aufbau einer hochverfügbaren Netzwerkkonnektivität zu Ihren öffentlichen Endpunkten ist das Routing des Datenverkehrs. Um sicherzustellen, dass Ihr Datenverkehr die Endpunkte erreichen kann, muss das DNS in der Lage sein, die Domain-Namen in die entsprechenden IP-Adressen aufzulösen. Verwenden Sie ein hochverfügbares und skalierbares [Domain Name System \(DNS\)](#) wie Amazon Route 53, um die DNS-Einträge Ihrer Domäne zu verwalten. Sie können außerdem die von Amazon Route 53 bereitgestellten Zustandsprüfungen verwenden. Die Zustandsprüfungen überprüfen, ob Ihre Anwendung erreichbar, verfügbar und funktionstüchtig ist. Sie können so eingerichtet werden, dass sie das Verhalten Ihres Benutzers nachahmen, z. B. das Anfordern einer Webseite oder einer bestimmten URL. Im Falle eines Fehlers reagiert Amazon Route 53 auf DNS-Auflösungsanfragen und leitet den Datenverkehr nur an Health-Endpunkte weiter. Sie können außerdem die von Amazon Route 53 angebotenen Funktionen für Geo-DNS und latenzbasiertes Routing nutzen.

Um zu überprüfen, ob Ihr Workload selbst hochverfügbar ist, verwenden Sie Elastic Load Balancing (ELB). Amazon Route 53 kann verwendet werden, um den Datenverkehr an ELB zu leiten, das den Datenverkehr an die Ziel-Computing-Instances verteilt. Sie können Amazon API Gateway außerdem zusammen mit AWS Lambda für eine Serverless-Lösung verwenden. Kunden können Workloads zudem in mehreren AWS-Regionen ausführen. Mit einem [Multi-Site Aktiv/Aktiv-Muster](#) kann der Workload den Datenverkehr aus mehreren Regionen bedienen. Bei einem Multi-Site Aktiv/Passiv-Muster bedient der Workload den Datenverkehr aus der aktiven Region, während die Daten in die sekundäre Region repliziert werden, die im Falle eines Fehlers in der primären Region aktiv wird. Mit Amazon Route 53-Zustandsprüfungen können Sie dann das DNS-Failover von einem beliebigen Endpunkt in einer primären Region zu einem Endpunkt in einer sekundären Region steuern und so sicherstellen, dass Ihr Workload erreichbar und für Ihre Benutzer verfügbar ist.

Amazon CloudFront bietet eine einfache API für die Verteilung von Inhalten mit geringer Latenz und hohen Datenübertragungsraten, indem Anfragen über ein Netzwerk von Edge-Standorten auf der ganzen Welt bedient werden. Content Delivery Networks (CDNs) dienen den Kunden, indem sie Inhalte bereitstellen, die sich in der Nähe des Benutzers befinden oder dort zwischengespeichert werden. Dies verbessert auch die Verfügbarkeit Ihrer Anwendung, da die Last der Inhalte von Ihren Servern auf die [Edge-Standorte](#) von CloudFront verlagert wird. Die Edge-Standorte und regionalen

Edge-Caches halten zwischengespeicherte Kopien Ihrer Inhalte in der Nähe Ihrer Benutzer vor, was einen schnellen Abruf ermöglicht und die Erreichbarkeit und Verfügbarkeit Ihres Workloads erhöht.

Bei Workloads mit geografisch verteilten Benutzern hilft AWS Global Accelerator Ihnen, die Verfügbarkeit und Leistung der Anwendungen zu verbessern. AWS Global Accelerator bietet statische Anycast-IP-Adressen, die als fester Zugangspunkt zu Ihrer Anwendung dienen, die in einer oder mehreren AWS-Regionen gehostet wird. Dadurch kann der Datenverkehr so nah wie möglich an Ihren Benutzern in das globale AWS Netzwerk geleitet werden, was die Erreichbarkeit und Verfügbarkeit Ihres Workloads verbessert. AWS Global Accelerator überwacht außerdem den Zustand Ihrer Anwendungsendpunkte mithilfe von TCP-, HTTP- und HTTPS-Zustandsprüfungen. Jede Änderung im Zustand oder in der Konfiguration Ihrer Endpunkte leitet den Benutzerverkehr auf funktionierende Endpunkte weiter, die Ihren Benutzern die beste Leistung und Verfügbarkeit bieten. Darüber hinaus verfügt AWS Global Accelerator über ein fehlerisolierendes Design, das zwei statische IPv4-Adressen verwendet, die von unabhängigen Netzwerkzonen bedient werden und die Verfügbarkeit Ihrer Anwendungen erhöhen.

AWS bietet AWS Shield Standard, um Kunden vor DDoS-Angriffen zu schützen. Shield Standard wird automatisch aktiviert und schützt vor gängigen Infrastrukturangriffen (Layer 3 und 4) wie SYN/UDP-Floods und Reflection-Angriffen, um die hohe Verfügbarkeit Ihrer Anwendungen auf AWS zu unterstützen. Für zusätzlichen Schutz vor ausgefeilteren und größeren Angriffen (wie UDP-Floods), State-Exhaustion-Angriffen (wie TCP-SYN-Floods) und zum Schutz Ihrer Anwendungen, die auf Amazon Elastic Compute Cloud (Amazon EC2), Elastic Load Balancing (ELB), Amazon CloudFront, AWS Global Accelerator und Amazon Route 53 ausgeführt werden, können Sie AWS Shield Advanced verwenden. Zum Schutz vor Angriffen auf der Anwendungsebene wie HTTP-POST- oder GET-Floods verwenden Sie AWS WAF. AWS WAF kann IP-Adressen, HTTP-Header, HTTP-Body, URI-Strings, SQL-Injections und Cross-Site-Scripting-Bedingungen verwenden, um zu bestimmen, ob eine Anfrage blockiert oder zugelassen werden soll.

Implementierungsschritte

1. Amazon Route 53 ist ein hochverfügbarer und skalierbarer [Domain Name System \(DNS\)](#)-Web-Service. Route 53 verbindet Benutzeranfragen mit Internetanwendungen, die auf AWS oder On-Premises ausgeführt werden. Weitere Informationen finden Sie unter [Konfigurieren von Amazon Route 53 als DNS-Service](#).
2. Richten Sie Zustandsprüfungen ein: Wenn Sie Amazon Route 53 verwenden, vergewissern Sie sich, dass nur korrekt funktionierende Ziele auflösbar sind. Starten Sie mit der [Erstellung von Route 53-Zustandsprüfungen und der Konfiguration des DNS-Failovers](#). Bei der Einrichtung von Zustandsprüfungen sind die folgenden Aspekte zu beachten:

- a. [So ermittelt Amazon Route 53, ob eine Zustandsprüfung fehlerfrei ist](#)
 - b. [Erstellen, Aktualisieren und Löschen von Zustandsprüfungen](#)
 - c. [Den Status von Zustandsprüfungen überwachen und Benachrichtigungen erhalten](#)
 - d. [Bewährte Methoden für Amazon Route 53 DNS](#)
3. [Verbinden Sie Ihren DNS-Service mit Ihren Endpunkten.](#)
- a. Wenn Sie Elastic Load Balancing als Ziel für Ihren Datenverkehr verwenden, erstellen Sie einen [Alias-Eintrag](#) mit Amazon Route 53, der auf den regionalen Endpunkt Ihres Load-Balancers verweist. Setzen Sie bei der Erstellung des Alias-Eintrags die Option „Zielzustand evaluieren“ auf „Ja“. Setzen Sie bei der Erstellung des Alias-Eintrags die Option „Zielzustand evaluieren“ auf „Ja“.
 - b. Verwenden Sie bei der Nutzung von API Gateway für Serverless-Workloads oder private APIs Route 53, [um den Datenverkehr zu API Gateway zu routen](#).
4. Entscheiden Sie sich für ein Content Delivery Netzwerk.
- a. Informieren Sie sich zunächst über [die Art und Weise, wie CloudFront-Inhalte über Edge-Standorte in der Nähe des Benutzers bereitgestellt werden](#).
 - b. Starten Sie mit einer [einfachen CloudFront-Verteilung](#). CloudFront weiß dann, von wo aus die Inhalte ausgeliefert werden sollen, und kennt die Details zur Nachverfolgung und Verwaltung der Content-Bereitstellung. Die folgenden Aspekte sollten Sie kennen und berücksichtigen, wenn Sie die CloudFront-Verteilung einrichten:
 - i. [Funktionsweise der Zwischenspeicherung mit CloudFront-Edge-Standorten](#)
 - ii. [Erhöhen des Anteils der Anforderungen, die direkt von den CloudFront-Caches bereitgestellt werden \(Cache-Trefferrate\)](#)
 - iii. [Verwenden von Amazon CloudFront Origin Shield](#)
 - iv. [Optimieren der Hochverfügbarkeit mit CloudFront-Ursprungs-Failover](#)
5. Einrichten des Schutzes auf der Anwendungsebene: AWS WAF hilft Ihnen, sich gegen gängige Web-Exploits und Bots zu schützen, die die Verfügbarkeit beeinträchtigen, die Sicherheit gefährden oder übermäßig viele Ressourcen verbrauchen können. Um ein tieferes Verständnis zu erlangen, lesen Sie [How AWS WAF works](#). Wenn Sie bereit sind, den Schutz vor HTTP-POST- und -GET-Floods auf der Anwendungsebene zu implementieren, lesen Sie [Getting started with AWS WAF](#). Sie können außerdem AWS WAF mit CloudFront verwenden. In der Dokumentation erfahren Sie, [wie AWS WAF mit Amazon CloudFront-Funktionen arbeitet](#).
6. Richten Sie einen zusätzlichen DDoS-Schutz ein: Standardmäßig erhalten alle Kunden von AWS mit AWS Shield Standard ohne zusätzliche Kosten einen Schutz gegen die gängigsten DDoS-

Angriffe auf Netzwerk- und Transportebene, die sich gegen Ihre Website oder Anwendung richten. Für zusätzlichen Schutz von Anwendungen, die auf Amazon EC2, Elastic Load Balancing, Amazon CloudFront, AWS Global Accelerator und Amazon Route 53 ausgeführt werden, können Sie [AWS Shield Advanced](#) einsetzen und sich [Beispiele für DDoS-resistente Architekturen ansehen](#). Um Ihren Workload und Ihre öffentlichen Endpunkte vor DDoS-Angriffen zu schützen, lesen Sie [Getting started with AWS Shield Advanced](#).

Ressourcen

Zugehörige bewährte Methoden:

- [REL10-BP01 Bereitstellen des Workloads an mehreren Standorten](#)
- [REL11-BP04 Nutzen der Datenebene und nicht der Steuerebene während der Wiederherstellung](#)
- [REL11-BP06 Senden von Benachrichtigungen, wenn sich Ereignisse auf die Verfügbarkeit auswirken](#)

Zugehörige Dokumente:

- [APN-Partner: Partner, die Sie bei der Planung Ihres Netzwerks unterstützen können](#)
- [AWS Marketplace für Netzwerkinfrastruktur](#)
- [Was ist AWS Global Accelerator?](#)
- [Was ist Amazon CloudFront?](#)
- [Was ist Amazon Route 53?](#)
- [Was ist Elastic Load Balancing?](#)
- [Network Connectivity capability - Establishing Your Cloud Foundations](#)
- [Was ist Amazon API Gateway?](#)
- [Was sind AWS WAF, AWS Shield und AWS Firewall Manager?](#)
- [Was ist Amazon Application Recovery Controller?](#)
- [Benutzerdefinierte Zustandsprüfungen für das DNS-Failover konfigurieren](#)

Zugehörige Videos:

- [AWS re:Invent 2022 – Improve performance and availability with AWS Global Accelerator](#)
- [AWS re:Invent 2020 – Global traffic management with Amazon Route 53](#)

- [AWS re:Invent 2022 – Operating highly available Multi-AZ applications](#)
- [AWS re:Invent 2022 – Dive deep on AWS networking infrastructure](#)
- [AWS re:Invent 2022 – Building resilient networks](#)

Zugehörige Beispiele:

- [Notfallwiederherstellung mit Amazon Application Recovery Controller \(ARC\)](#)
- [AWS Global Accelerator Workshop](#)

REL02-BP02 Bereitstellen redundanter Konnektivität zwischen privaten Netzwerken in der Cloud und in On-Premises-Umgebungen

Implementieren Sie Redundanz in Ihren Verbindungen zwischen privaten Netzwerken in der Cloud und On-Premises-Umgebungen, um die Stabilität der Konnektivität zu erreichen. Dies kann erreicht werden, indem zwei oder mehr Verbindungen und Datenverkehrspfade bereitgestellt werden, sodass die Konnektivität bei Netzerkassenfällen erhalten bleibt.

Typische Anti-Muster:

- Sie verlassen sich auf nur eine Netzwerkverbindung, was zu einer einzigen Fehlerquelle führt.
- Sie verwenden nur einen VPN-Tunnel oder mehrere Tunnel, die in derselben Availability Zone enden.
- Sie verlassen sich bei der VPN-Konnektivität auf einen ISP, was bei ISP-Ausfällen zu kompletten Ausfällen führen kann.
- Keine Implementierung dynamischer Routing-Protokolle wie BGP, die für die Umleitung des Datenverkehrs bei Netzwerkunterbrechungen von entscheidender Bedeutung sind.
- Sie ignorieren die Bandbreitenbeschränkungen von VPN-Tunneln und überschätzen deren Backup-Fähigkeiten.

Vorteile der Nutzung dieser bewährten Methode: Durch die Implementierung redundanter Konnektivität zwischen Ihrer Cloud-Umgebung und Ihrer Unternehmens- bzw. On-Premises-Umgebung wird die sichere Kommunikation der abhängigen Services zwischen den beiden Umgebungen gewährleistet.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Wenn Sie AWS Direct Connect verwenden, um Ihr On-Premises-Netzwerk mit AWS zu verbinden, können Sie maximale Netzwerkstabilität (SLA von 99,99 %) erreichen, indem Sie separate Verbindungen verwenden, die auf verschiedenen Geräten an mehr als einem On-Premises-Standort und mehr als einem AWS Direct Connect-Standort enden. Diese Topologie bietet Widerstandsfähigkeit gegen Geräteausfälle, Verbindungsprobleme und komplette Standortausfälle. Alternativ können Sie eine hohe Ausfallsicherheit (SLA von 99,9 %) erreichen, indem Sie zwei einzelne Verbindungen zu mehreren Standorten verwenden (jeder On-Premises-Standort ist mit einem einzigen Direct-Connect-Standort verbunden). Dieser Ansatz schützt vor Verbindungsunterbrechungen, die durch getrennte Glasfaserkabel oder Geräteausfälle verursacht werden, und trägt dazu bei, die Auswirkungen kompletter Standortausfälle zu mindern. Das Direct Connect Resiliency Toolkit unterstützt Sie beim Entwerfen Ihrer AWS Direct Connect-Topologie.

Sie können auch erwägen, dass AWS Site-to-Site VPN auf einem AWS Transit Gateway endet, um ein kostengünstiges Backup Ihrer primären Verbindung mit AWS Direct Connect zu erhalten. Dieses Setup ermöglicht Equal-Cost-Multipath (ECMP)-Routing über mehrere VPN-Tunnel und ermöglicht einen Durchsatz von bis zu 50 Gbit/s, obwohl jeder VPN-Tunnel auf 1,25 Gbit/s begrenzt ist. Es ist jedoch wichtig zu beachten, dass AWS Direct Connect immer noch die effektivste Wahl ist, um Netzwerkunterbrechungen zu minimieren und eine stabile Konnektivität bereitzustellen.

Wenn Sie VPNs über das Internet verwenden, um Ihre Cloud-Umgebung mit Ihrem On-Premises-Rechenzentrum zu verbinden, konfigurieren Sie zwei VPN-Tunnel als Teil einer einzigen Site-to-Site-VPN-Verbindung. Jeder Tunnel sollte aus Gründen der Hochverfügbarkeit in einer anderen Availability Zone enden und redundante Hardware verwenden, um Ausfälle von On-Premises-Geräten zu verhindern. Erwägen Sie außerdem mehrere Internetverbindungen von verschiedenen Internetdienstanbietern (ISPs) an Ihrem On-Premises-Standort, um eine vollständige Unterbrechung der VPN-Konnektivität durch den Ausfall eines einzigen ISP zu vermeiden. Die Auswahl von ISPs mit unterschiedlichem Routing und Infrastruktur, insbesondere solchen mit separaten physischen Pfaden zu AWS-Endpunkten, sorgt für eine hohe Konnektivitätsverfügbarkeit.

Neben der physischen Redundanz mit mehreren AWS Direct Connect-Verbindungen und VPN-Tunneln (oder einer Kombination aus beiden) ist auch die Implementierung des dynamischen Routings des Border Gateway Protocol (BGP) von entscheidender Bedeutung. Dynamisches BGP ermöglicht die automatische Umleitung des Datenverkehrs von einem Pfad zum nächsten, basierend auf Netzwerkbedingungen in Echtzeit und konfigurierten Richtlinien. Dieses dynamische Verhalten ist besonders vorteilhaft zur Aufrechterhaltung der Netzwerkverfügbarkeit und Servicekontinuität bei

Verbindungs- oder Netzerkausfällen. Es wählt schnell alternative Pfade aus und verbessert so die Ausfallsicherheit und Zuverlässigkeit des Netzwerks.

Implementierungsschritte

- Erwerben Sie hochverfügbare Konnektivität zwischen AWS und Ihrer On-Premises-Umgebung.
 - Verwenden Sie mehrere AWS Direct Connect-Verbindungen oder VPN-Tunnel zwischen separat bereitgestellten privaten Netzwerken.
 - Verwenden Sie für eine hohe Verfügbarkeit mehrere Direct Connect-Standorte.
 - Wenn Sie mehrere AWS-Regionen verwenden, sorgen Sie in mindestens zwei davon für Redundanz.
- Verwenden Sie AWS Transit Gateway, wenn möglich, um Ihre [VPN-Verbindung](#) zu beenden.
- Beurteilen Sie AWS Marketplace-Appliances, um VPNs zu beenden oder [erweitern Sie Ihr SD-WAN auf AWS](#). Stellen Sie bei Verwendung von AWS Marketplace-Appliances redundante Instances bereit, um eine hohe Verfügbarkeit in verschiedenen Availability Zones zu gewährleisten.
- Stellen Sie auf Ihrer On-Premises-Umgebung eine redundante Verbindung her.
 - Möglicherweise benötigen Sie redundante Verbindungen zu mehreren AWS-Regionen, um die erforderliche Verfügbarkeit zu gewährleisten.
- Verwenden Sie das [Direct Connect Resiliency Toolkit](#), um loszulegen.

Ressourcen

Zugehörige Dokumente:

- [AWS Direct Connect Resiliency Recommendations](#)
- [Using Redundant Site-to-Site VPN Connections to Provide Failover](#)
- [Routing policies and BGP communities](#)
- [Active/Active and Active/Passive Configurations in AWS Direct Connect](#)
- [APN-Partner: Partner, die Sie bei der Planung Ihres Netzwerks unterstützen können](#)
- [AWS Marketplace für Netzwerkinfrastruktur](#)
- Whitepaper [Amazon Virtual Private Cloud Connectivity Options](#)
- [Building a Scalable and Secure Multi-VPC AWS Network Infrastructure](#)
- [Using redundant Site-to-Site VPN connections to provide failover](#)
- [Using the Direct Connect Resiliency Toolkit to get started](#)

- [VPC Endpoints and VPC Endpoint Services \(AWS PrivateLink\)](#)
- [Was ist Amazon VPC?](#)
- [What is a transit gateway?](#)
- [Was ist AWS Site-to-Site VPN?](#)
- [Working with Direct Connect gateways](#)

Zugehörige Videos:

- [AWS re:Invent 2018: Advanced VPC Design and New Capabilities for Amazon VPC](#)
- [AWS re:Invent 2019: AWS Transit Gateway reference architectures for many VPCs](#)

REL02-BP03 Berücksichtigen von Erweiterungen und Verfügbarkeit bei der Zuweisung von IP-Adressen für Subnetze

Die IP-Adressbereiche für Amazon VPC müssen ausreichend groß sein, um die Anforderungen eines Workloads zu erfüllen. Dabei sind zukünftige Erweiterungen und Zuweisungen von IP-Adressen zu Subnetzen in verschiedenen Availability Zones zu berücksichtigen. Dies betrifft Load Balancer, EC2-Instances sowie containerbasierte Anwendungen.

Wenn Sie Ihre Netzwerktopologie planen, besteht der erste Schritt in der Definition des IP-Adressbereichs. Private IP-Adressbereiche (gemäß RFC 1918-Richtlinien) sollten jeder VPC zugewiesen werden. Berücksichtigen Sie im Rahmen dieses Prozesses die folgenden Anforderungen:

- Ermöglichen Sie einen IP-Adressbereich für mehr als eine VPC pro Region.
- Planen Sie innerhalb einer VPC Platz für mehrere Subnetze ein, damit Sie mehrere Availability Zones abdecken können.
- Lassen Sie für eine zukünftige Erweiterung stets Raum für nicht verwendete CIDR-Blöcke innerhalb einer VPC.
- Stellen Sie sicher, dass ein IP-Adressbereich vorhanden ist, um die Anforderungen von temporären Amazon EC2-Instances zu erfüllen, die Sie möglicherweise verwenden, z. B. Spot-Flotten für Machine Learning, Amazon EMR-Cluster oder Amazon Redshift-Cluster. Ähnliche Überlegungen sollten für Kubernetes-Cluster wie Amazon Elastic Kubernetes Service (Amazon EKS) getroffen werden, da jedem Kubernetes-Pod standardmäßig eine routbare Adresse aus dem VPC-CIDR-Block zugewiesen wird.

- Beachten Sie, dass die ersten vier IP-Adressen und die letzte IP-Adresse in jedem Subnetz-CIDR-Block reserviert und nicht für Sie verfügbar sind.
- Beachten Sie, dass der VPC CIDR-Block, der anfänglich Ihrer VPC zugewiesen war, nicht geändert oder gelöscht werden kann. Sie können der VPC jedoch zusätzliche, nicht überlappende CIDR-Blöcke hinzufügen. IPv4-CIDRs für Subnetze können nicht geändert werden, IPv6 CIDRs jedoch schon.
- Der größte mögliche VPC-CIDR-Block entspricht /16 und der kleinste /28.
- Berücksichtigen Sie andere verbundene Netzwerke (VPC, On-Premises oder sonstige Cloud-Anbieter) und stellen Sie sicher, dass sich der IP-Adressraum nicht überschneidet. Weitere Informationen finden Sie unter [Erzwingen von sich nicht überschneidenden privaten IP-Adressbereichen in allen privaten Adressbereichen, in denen eine Verbindung besteht](#).

Gewünschtes Ergebnis: Ein skalierbares IP-Subnetz kann Ihnen helfen, zukünftiges Wachstum zu bewältigen und unnötige Verschwendung zu vermeiden.

Typische Anti-Muster:

- Wenn zukünftiges Wachstum nicht berücksichtigt wird, sind die CIDR-Blöcke zu klein und müssen neu konfiguriert werden, was zu Ausfallzeiten führen kann.
- Es wird falsch eingeschätzt, wie viele IP-Adressen ein Elastic Load Balancer verwenden kann.
- Es werden viele Load Balancer mit hohem Datenverkehr in denselben Subnetzen bereitgestellt.
- Es werden automatische Skalierungsmechanismen verwendet, während der Verbrauch von IP-Adressen nicht überwacht wird.
- Die Definition übermäßig großer CIDR-Bereiche liegt weit über den zukünftigen Wachstumserwartungen, was zu Schwierigkeiten beim Peering mit anderen Netzwerken mit überlappenden Adressbereichen führen kann.

Vorteile der Nutzung dieser bewährten Methode: So wird sichergestellt, dass Sie das Wachstum Ihrer Workloads bewältigen können und beim Hochskalieren weiterhin die entsprechende Verfügbarkeit bereitstellen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Berücksichtigen Sie bei der Planung Ihres Netzwerks Ihr zukünftiges Wachstum, die Einhaltung gesetzlicher Vorschriften sowie die Kompatibilität mit anderen Netzwerken. Das

Wachstum kann unterschätzt werden, gesetzliche Vorschriften können sich ändern, und bei Unternehmensübernahmen oder privaten Netzwerkverbindungen kann die Implementierung ohne fundierte Planung zur Herausforderung werden.

- Wählen Sie relevante AWS-Konten und Regionen anhand von Serviceanforderungen, regulatorischen Anforderungen sowie Anforderungen für die Latenz und die Notfallwiederherstellung aus.
- Identifizieren Sie Ihre Anforderungen bezüglich regionaler VPC-Bereitstellungen.
- Ermitteln Sie die erforderliche Größe der VPCs.
 - Ermitteln Sie, ob Multi-VPC-Konnektivität bereitgestellt werden soll.
 - [Was ist ein Transit-Gateway?](#)
 - [Multi-VPC-Konnektivität in einer Region](#)
- Ermitteln Sie, ob aufgrund von Compliance-Anforderungen getrennte Netzwerke erforderlich sind.
- Erstellen Sie VPCs mit CIDR-Blöcken in geeigneter Größe, um Ihren aktuellen und zukünftigen Anforderungen gerecht zu werden.
 - Wenn Sie unbekannte Wachstumsprognosen haben, sollten Sie sich für größere CIDR-Blöcke entscheiden, um das Potenzial einer zukünftigen Neukonfiguration zu verringern.
- Erwägen Sie die Verwendung von [IPv6-Adressierung](#) für Subnetze als Teil einer Dual-Stack-VPC. IPv6 eignet sich gut für den Einsatz in privaten Subnetzen, die Flotten kurzlebiger Instances oder Container enthalten, für die andernfalls eine große Anzahl von IPv4-Adressen erforderlich wäre.

Ressourcen

Zugehörige bewährte Methoden für Well-Architected:

- [REL02-BP05 Erzwingen von sich nicht überschneidenden privaten IP-Adressbereichen in allen privaten Adressbereichen, in denen eine Verbindung besteht](#)

Zugehörige Dokumente:

- [APN-Partner: Partner, die Sie bei der Planung Ihres Netzwerks unterstützen können](#)
- [AWS Marketplace für Netzwerkinfrastruktur](#)
- [Whitepaper Amazon Virtual Private Cloud Connectivity Options](#)

- [Multiple data center HA network connectivity](#)
- [Multi-VPC-Konnektivität in einer Region](#)
- [Was ist Amazon VPC?](#)
- [IPv6 auf AWS](#)
- [IPv6 in Referenzarchitekturen](#)
- [Amazon Elastic Kubernetes Service launches IPv6 support](#)
- [Empfehlungen für Ihre VPC – Classic Load Balancer](#)
- [Availability-Zone-Subnetze – Application Load Balancer](#)
- [Availability Zones – Network Load Balancer](#)

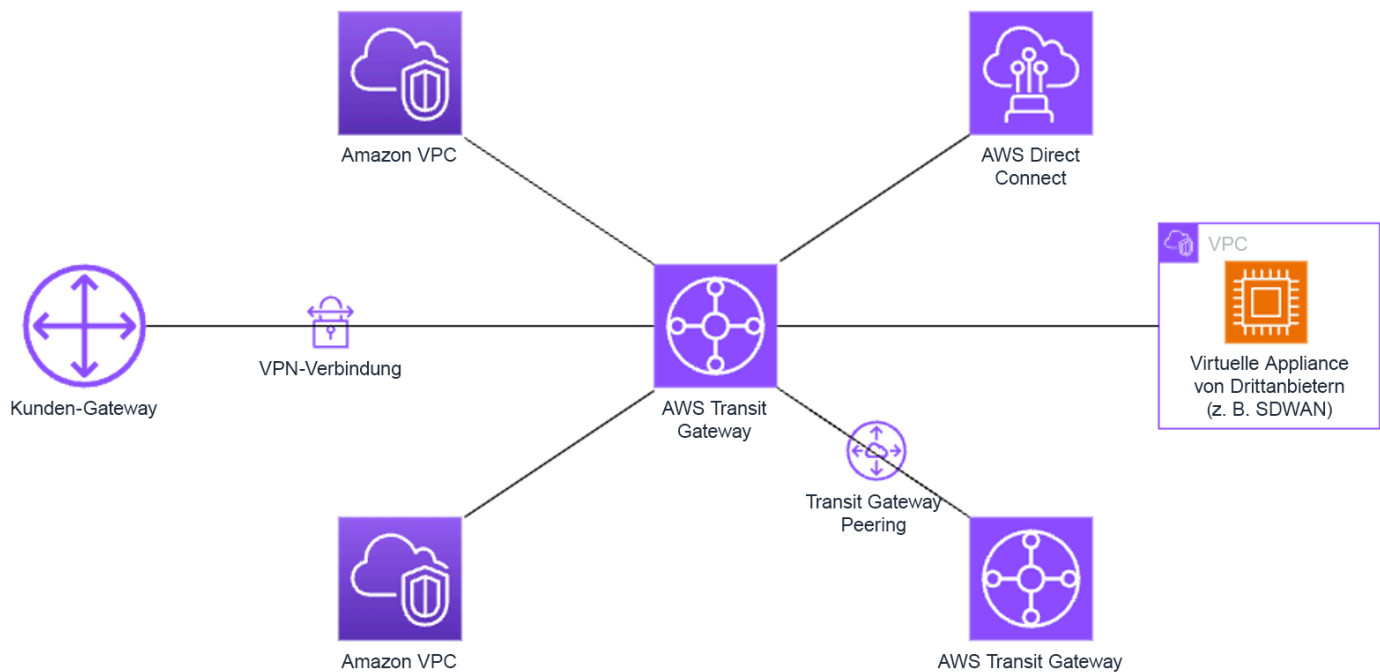
Zugehörige Videos:

- [AWS re:Invent 2018: Advanced VPC Design and New Capabilities for Amazon VPC \(NET303\)](#)
- [AWS re:Invent 2019: AWS Transit Gateway reference architectures for many VPCs \(NET406-R1\)](#)
- [AWS re:Invent 2023: AWS Ready for what's next? Designing networks for growth and flexibility \(NET310\)](#)

REL02-BP04 Vorziehen von Hub-and-Spoke-Topologien gegenüber M-zu-N-Netzen

Wenn Sie mehrere private Netzwerke wie Virtual Private Clouds (VPCs) und On-Premises-Netzwerke verbinden, sollten Sie sich für eine Hub-and-Spoke-Topologie statt für eine verflochtene Topologie entscheiden. Im Gegensatz zu verflochtenen Topologien, bei denen jedes Netzwerk direkt mit dem anderen verbunden ist, was die Komplexität und den Verwaltungsaufwand erhöht, zentralisiert die Hub-and-Spoke-Architektur Verbindungen über einen einzigen Hub. Diese Zentralisierung vereinfacht die Netzwerkstruktur und verbessert deren Bedienbarkeit, Skalierbarkeit und Kontrolle.

AWS Transit Gateway ist ein verwalteter, skalierbarer und hochverfügbarer Service, der für den Aufbau von Hub-and-Spoke-Netzwerken auf AWS entwickelt wurde. Er dient als zentraler Knotenpunkt Ihres Netzwerks, der Netzwerksegmentierung, zentralisiertes Routing und die vereinfachte Verbindung zu Cloud- und On-Premises-Umgebungen ermöglicht. Die folgende Abbildung zeigt, wie Sie Ihre Hub-and-Spoke-Topologie mit AWS Transit Gateway entwickeln können.



Gewünschtes Ergebnis: Sie haben Ihre Virtual Private Clouds (VPCs) und On-Premises-Netzwerke über einen zentralen Hub verbunden. Sie konfigurieren Ihre Peering-Verbindungen über den Hub, der als hoch skalierbarer Cloud-Router fungiert. Das Routing ist vereinfacht, da Sie nicht mit komplexen Peering-Beziehungen arbeiten müssen. Der Datenverkehr zwischen Netzwerken ist verschlüsselt und Sie haben die Möglichkeit, Netzwerke zu isolieren.

Typische Anti-Muster:

- Sie erstellen komplexe Netzwerk-Peering-Regeln.
- Sie stellen Routen zwischen Netzwerken bereit, die nicht miteinander kommunizieren sollten (z. B. separate Workloads, die keine gegenseitigen Abhängigkeiten haben).
- Die Verwaltung der Hub-Instance ist ineffektiv.

Vorteile der Nutzung dieser bewährten Methode: Wenn die Anzahl der verbundenen Netzwerke zunimmt, wird die Verwaltung und Erweiterung der Mesh-Konnektivität immer schwieriger. Eine Mesh-Architektur bringt zusätzliche Herausforderungen mit sich, z. B. zusätzliche Infrastrukturkomponenten, Konfigurationsanforderungen und Überlegungen zur Bereitstellung. Das Mesh bringt außerdem zusätzlichen Aufwand für die Verwaltung und Überwachung der Komponenten der Datenebene und der Steuerebene mit sich. Sie müssen darüber nachdenken, wie eine hohe

Verfügbarkeit der Mesh-Architektur gewährleistet werden kann, wie der Zustand und die Leistung des Mesh überwacht werden können und wie mit Upgrades der Mesh-Komponenten umzugehen ist.

Ein Hub-and-Spoke-Modell hingegen ermöglicht die zentrale Weiterleitung des Datenverkehrs über mehrere Netzwerke hinweg. Es bietet einen einfacheren Ansatz für die Verwaltung und Überwachung der Komponenten der Datenebene und der Steuerebene.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Erstellen Sie ein Network-Services-Konto, wenn keins vorhanden ist. Platzieren Sie den Hub im Network-Services-Konto der Organisation. Dies ermöglicht die zentrale Verwaltung des Hubs durch Netzwerktechniker.

Im Hub-and-Spoke-Modell dient der Hub als virtueller Router für den Datenverkehr zwischen Ihren Virtual Private Clouds (VPCs) und On-Premises-Netzwerken. Dies reduziert die Netzwerkkomplexität und vereinfacht die Behebung von Netzwerkproblemen.

Berücksichtigen Sie Ihr Netzwerkdesign, einschließlich der VPCs, AWS Direct Connect und die Site-to-Site-VPN-Verbindungen, die Sie herstellen möchten.

Verwenden Sie für jede angefügte Transit-Gateway-VPC ein separates Subnetz. Verwenden Sie für jedes Subnetz einen kleinen CIDR-Wert (z. B. /28), damit Sie mehr Adressraum für Computing-Ressourcen haben. Erstellen Sie eine Netzwerk-Zugriffskontrollliste (ACL) und weisen Sie diese allen Subnetzen zu, die mit dem Hub verbunden sind. Halten Sie die Netzwerk-ACL sowohl in der Richtung für eingehenden als auch in der Richtung für ausgehenden Datenverkehr geöffnet.

Entwerfen und implementieren Sie Ihre Routing-Tabellen so, dass Routen nur zwischen Netzwerken bereitgestellt werden, die miteinander kommunizieren sollen. Lassen Sie Routen zwischen Netzwerken aus, die nicht miteinander kommunizieren sollen (z. B. zwischen separaten Workloads, die keine gegenseitigen Abhängigkeiten haben).

Implementierungsschritte

1. Planen Sie Ihr Netzwerk. Ermitteln Sie, welche Netzwerke Sie verbinden möchten, und stellen Sie sicher, dass diese keine sich überschneidenden CIDR-Bereiche teilen.
2. Erstellen Sie ein AWS Transit Gateway und hängen Sie Ihre VPCs an.
3. Erstellen Sie VPN-Verbindungen oder Direct-Connect-Gateways und verknüpfen Sie diese mit dem Transit-Gateway, wenn notwendig.

4. Definieren Sie durch die Konfiguration Ihrer Routing-Tabelle für den Transit-Gateway, wie der Datenverkehr zwischen den verbundenen VPCs und anderen Verbindungen geleitet wird.
5. Verwenden Sie Amazon CloudWatch, um Konfigurationen zu überwachen und bei Bedarf zur Leistungs- und Kostenoptimierung anzupassen.

Ressourcen

Zugehörige bewährte Methoden:

- [REL02-BP03 Berücksichtigen von Erweiterung und Verfügbarkeit bei der Zuweisung von IP-Adressen für Subnetze](#)
- [REL02-BP05 Durchsetzen von sich nicht überschneidenden privaten IP-Adressbereichen in allen privaten Adressbereichen, in denen eine Verbindung besteht](#)

Zugehörige Dokumente:

- [Was ist ein Transit-Gateway?](#)
- [Bewährte Methoden für das Transit-Gateway-Design](#)
- [Building a Scalable and Secure Multi-VPC AWS Network Infrastructure](#)
- [Building a global network using AWS Transit Gateway Inter-Region peering](#)
- [Amazon Virtual Private Cloud Connectivity Options](#)
- [APN-Partner: Partner, die Sie bei der Planung Ihres Netzwerks unterstützen können](#)
- [AWS Marketplace für Netzwerkinfrastruktur](#)

Zugehörige Videos:

- [AWS re:Invent 2.023 - AWS networking foundations](#)
- [AWS re:Invent 2.023 - Advanced VPC designs and new capabilities](#)

Zugehörige Workshops:

- [AWS Transit Gateway-Workshop](#)

REL02-BP05 Erzwingen von sich nicht überschneidenden privaten IP-Adressbereichen in allen privaten Adressbereichen, in denen eine Verbindung besteht

Die IP-Adressbereiche Ihrer VPCs dürfen sich nicht überschneiden, wenn sie per Peering oder über Transit Gateway oder VPN verbunden sind. Vermeiden Sie IP-Adresskonflikte zwischen einer VPC und On-Premises-Umgebungen oder anderen verwendeten Cloud-Anbietern. Sie müssen bei Bedarf auch die Möglichkeit haben, private IP-Adressbereiche zuzuweisen. Ein IP-Adressenverwaltungssystem (IPAM) kann bei der Automatisierung helfen.

Gewünschtes Ergebnis:

- Keine Konflikte mit IP-Adressbereichen zwischen VPCs, On-Premises-Umgebungen oder anderen Cloud-Anbietern.
- Eine angemessene IP-Adressverwaltung ermöglicht eine einfachere Skalierung der Netzwerkinfrastruktur, um wachsenden und sich wandelnden Netzwerkanforderungen gerecht zu werden.

Typische Anti-Muster:

- Verwendung desselben IP-Bereichs in Ihrer VPC wie On-Premises, in Ihrem Unternehmensnetzwerk oder bei anderen Cloud-Anbietern.
- Keine Verfolgung von IP-Bereichen von VPCs, die zur Bereitstellung der Workloads verwendet werden.
- Alleinige Nutzung manueller IP-Adressverwaltungsprozesse wie Tabellenkalkulationen.
- Über- oder Unterdimensionierung von CIDR-Blöcken, was zu einer Verschwendung von IP-Adressen oder zu wenig Adressbereichen für Ihren Workload führt.

Vorteile der Nutzung dieser bewährten Methode: Mit der aktiven Planung des Netzwerks stellen Sie sicher, dass dieselbe IP-Adresse in miteinander verbunden Netzwerken nicht mehrmals vorkommt. So wird verhindert, dass Routing-Probleme in Teilen der Workload auftreten, die die verschiedenen Anwendungen verwenden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Verwenden Sie ein IPAM, z. B. den [Amazon VPC IP Address Manager](#), um Ihre CIDR-Nutzung zu überwachen und zu verwalten. Im AWS Marketplace stehen auch mehrere IPAMs zur Verfügung.

Bewerten Sie die potenzielle Nutzung in AWS, fügen Sie vorhandenen VPCs CIDR-Bereiche hinzu und erstellen Sie neue VPCs, um das geplante Wachstum abzudecken.

Implementierungsschritte

- Ermitteln Sie den aktuellen CIDR-Umfang (z. B. VPCs und Subnetze).
 - Erfassen Sie über die Service API den aktuellen CIDR-Umfang.
 - Verwenden Sie den [Amazon VPC IP Address Manager, um Ressourcen zu entdecken](#).
- Erfassen Sie die aktuelle Subnetzauslastung.
 - [Ermitteln](#) Sie über die Service-API die in jeder Region pro VPC vorhandenen Subnetze.
 - Verwenden Sie den [Amazon VPC IP Address Manager, um Ressourcen zu entdecken](#).
- Zeichnen Sie die aktuelle Auslastung auf.
- Prüfen Sie, ob sich IP-Bereiche überschneiden.
- Berechnen Sie die freie Kapazität.
- Identifizieren Sie sich überschneidende IP-Bereiche. Sie können wahlweise zu einem neuen Adressbereich migrieren oder Techniken wie [privates NAT-Gateway](#) oder [AWS PrivateLink](#) verwenden, wenn Sie die sich überschneidenden Bereiche verbinden müssen.

Ressourcen

Zugehörige bewährte Methoden:

- [Schutz von Netzwerken](#)

Zugehörige Dokumente:

- [APN-Partner: Partner, die Sie bei der Planung Ihres Netzwerks unterstützen können](#)
- [AWS Marketplace für Netzwerkinfrastruktur](#)
- Whitepaper [Amazon Virtual Private Cloud Connectivity Options](#)
- [Multiple data center HA network connectivity](#)
- [Connecting Networks with Overlapping IP Ranges](#)
- [Was ist Amazon VPC?](#)
- [Was ist IPAM?](#)

Zugehörige Videos:

- [AWS re:Invent 2023: Advanced VPC designs and new capabilities](#)
- [AWS re:Invent 2019: AWS Transit Gateway reference architectures for many VPCs](#)
- [AWS re:Invent 2023: Ready for what's next? Designing networks for growth and flexibility](#)
- [AWS re:Invent 2021: {New Launch} Manage your IP addresses at scale on AWS](#)

Workload-Architektur

Fragen

- [REL 3. Wie entwerfen Sie Ihre Workload-Service-Architektur?](#)
- [REL 4. Wie lassen sich Interaktionen in einem verteilten System so gestalten, dass Ausfälle vermieden werden?](#)
- [REL 5. Wie lassen sich Interaktionen in einem verteilten System so gestalten, dass Ausfälle abgemildert oder bewältigt werden?](#)

REL 3. Wie entwerfen Sie Ihre Workload-Service-Architektur?

Erstellen Sie hoch skalierbare und zuverlässige Workloads mithilfe einer serviceorientierten Architektur (SOA) oder einer Microservices-Architektur. Eine serviceorientierte Architektur (SOA) hat zum Ziel, Softwarekomponenten über Service-Schnittstellen wiederverwendbar zu machen. Die Microservices-Architektur geht noch weiter, um Komponenten kleiner und einfacher zu machen.

Best Practices

- [REL03-BP01 Wählen Sie, wie Sie Ihre Arbeitslast segmentieren möchten](#)
- [REL03-BP02 Entwickeln von Services, die sich auf bestimmte Geschäftsdomänen und Funktionen konzentrieren](#)
- [REL03-BP03 Stellen Sie Serviceverträge bereit per API](#)

REL03-BP01 Wählen Sie, wie Sie Ihre Arbeitslast segmentieren möchten

Die Workload-Segmentierung ist wichtig, wenn es um die Festlegung der Resilienzanforderungen Ihrer Anwendung geht. Eine monolithische Architektur sollte vermieden werden, wann immer möglich. Stattdessen sollten Sie sorgfältig überlegen, welche Anwendungskomponenten in Microservices aufgeteilt werden können. Je nach Ihren Anwendungsanforderungen kann es sich dabei, wenn

möglich, um eine Kombination aus einer serviceorientierten Architektur (SOA) und Microservices handeln. Workloads, die zustandslos sein können, können eher als Microservices bereitgestellt werden.

Gewünschtes Ergebnis: Workloads sollten unterstützbar, skalierbar und so lose miteinander verbunden sein wie möglich.

Wiegen Sie bei Entscheidungen zur Segmentierung von Workloads die Vorteile und die Komplexitäten miteinander ab. Was für ein neues Produkt richtig ist, das gerade auf dem Markt eingeführt wird, unterscheidet sich von den Anforderungen eines Workloads, der von Anfang an skalierbar sein muss. Bei einem Faktorwechsel für einen vorhandenen Monolith müssen Sie berücksichtigen, wie gut dieser aufgeteilt und in zustandslose Anwendungen transformiert werden kann. Die Aufteilung von Services in kleinere Teile ermöglicht kleinen, klar definierten Teams, diese weiterzuentwickeln und zu verwalten. Kleinere Services können jedoch Komplexitäten wie eine möglicherweise erhöhte Latenz, ein komplexeres Debugging und einen erhöhten operativen Aufwand einführen.

Typische Anti-Muster:

- Der [Microservice Death Star](#) ist eine Situation, in der die einzelnen Komponenten so stark voneinander abhängig werden, dass der Ausfall einer einzigen Komponente einen wesentlich größeren Ausfall bewirkt. Das bedeutet, dass die Komponenten so starr und anfällig wie ein Monolith sind.

Vorteile der Nutzung dieser bewährten Methode:

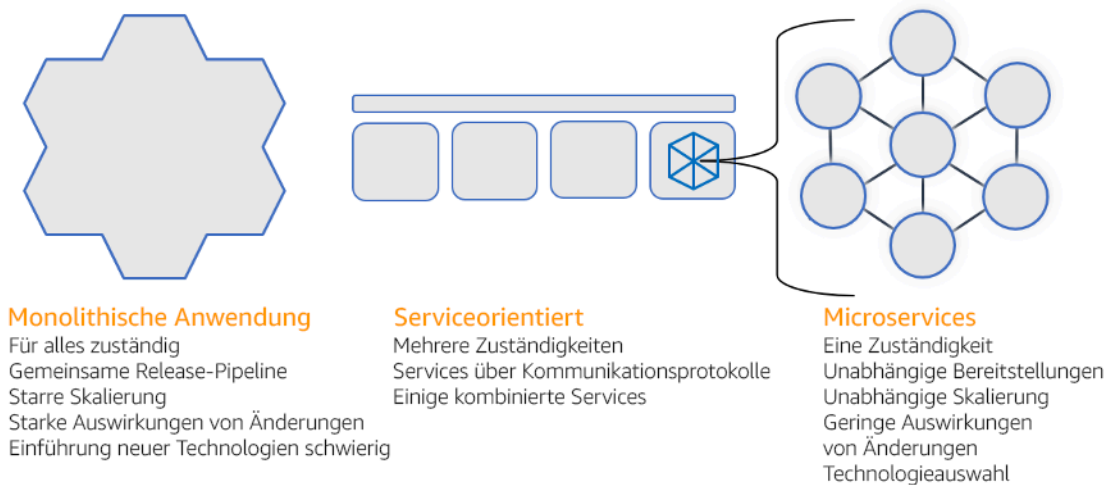
- Spezifischere Segmente führen zu einer größeren Agilität, zu organisatorischer Flexibilität und zu Skalierbarkeit.
- Die Auswirkungen von Service-Unterbrechungen werden reduziert.
- Die einzelnen Komponenten einer Anwendung besitzen möglicherweise unterschiedliche Anforderungen an die Verfügbarkeit, die von einer stärkeren Segmentierung besser unterstützt werden können.
- Die Verantwortlichkeiten der Teams, die den Workload unterstützen, sind klar definiert.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Wählen Sie Ihren Architekturtyp basierend auf der Segmentierung Ihres Workloads aus. Wählen Sie eine SOA oder eine Microservices-Architektur (oder in einigen seltenen Fällen eine monolithische Architektur). Selbst wenn Sie sich dafür entscheiden, mit einer Monolith-Architektur zu beginnen, müssen Sie sicherstellen, dass diese modular ist und sich letztendlich zu SOA Microservices weiterentwickeln kann, wenn Ihr Produkt mit der Benutzerakzeptanz skaliert. SOA und Microservices bieten jeweils eine geringere Segmentierung, was als moderne, skalierbare und zuverlässige Architektur bevorzugt wird. Allerdings müssen auch Kompromisse berücksichtigt werden, insbesondere bei der Implementierung einer Microservice-Architektur.

Aufgrund ihrer verteilten Computing-Architektur kann es schwieriger sein, die Latenzanforderungen von Benutzern zu erfüllen. Außerdem sind das Debugging und die Nachverfolgung von Benutzerinteraktionen komplexer. Zur Lösung dieses Problems können Sie AWS X-Ray verwenden. Ein weiterer Effekt ist die erhöhte operative Komplexität, da die Anzahl der von Ihnen verwalteten Anwendungen zunimmt. In der Folge müssen Sie eine größere Zahl voneinander unabhängiger Komponenten bereitstellen.



Monolithische, serviceorientierte und Microservice-Architekturen

Implementierungsschritte

- Ermitteln Sie die richtige Architektur für den Faktorwechsel oder die Entwicklung Ihrer Anwendung. SOA und Microservices bieten jeweils eine geringere Segmentierung, was als moderne, skalierbare und zuverlässige Architektur bevorzugt wird. SOA kann ein guter Kompromiss sein, um eine kleinere Segmentierung zu erreichen und gleichzeitig einige der Komplexitäten von Microservices zu vermeiden. Weitere Informationen finden Sie unter [Microservice Trade-Offs](#).

- Wenn Ihre Workload für sie zugänglich ist und Ihre Organisation sie unterstützen kann, sollten Sie eine Microservices-Architektur verwenden, um die beste Agilität und Zuverlässigkeit zu erzielen. [Weitere Informationen finden Sie unter Implementierung von Microservices auf AWS](#)
- Sie sollten das Muster mit der Bezeichnung [Strangler Fig \(„Würgefeige“\)](#) verwenden, um einen Faktorwechsel für einen Monolithen durchzuführen, bei dem Sie diesen in kleinere Komponenten aufteilen. Dies umfasst die schrittweise Ersetzung spezifischer Anwendungskomponenten durch neue Anwendungen und Services. [AWS Migration Hub Refactor Spaces](#) dient als Ausgangspunkt für den inkrementellen Faktorwechsel. Weitere Informationen finden Sie unter [Seamlessly migrate on-premises legacy workloads using a strangler pattern](#).
- Für die Implementierung von Microservices ist möglicherweise ein Service Discovery-Mechanismus erforderlich, der es diesen verteilten Diensten ermöglicht, miteinander zu kommunizieren. [AWS App Mesh](#) kann mit serviceorientierten Architekturen verwendet werden, um eine zuverlässige Erkennung und den zuverlässigen Zugriff auf Dienste zu ermöglichen. [AWS Cloud Map](#) kann auch für die dynamische, DNS basierte Serviceerkennung verwendet werden.
- Wenn Sie von einem Monolith zu Amazon MQ migrieren, kann [Amazon MQ](#) Ihnen helfen, die Lücke als Service Bus bei der Neugestaltung älterer Anwendungen in der Cloud zu schließen.
- Im Fall vorhandener Monolithen mit einer einzigen, geteilten Datenbank müssen Sie entscheiden, wie Sie die Daten neu in kleineren Segmenten organisieren. Dabei kann es sich um Geschäftsbereiche, Zugriffsmuster oder Datenstrukturen handeln. An diesem Punkt des Refactoring-Prozesses sollten Sie sich dafür entscheiden, mit einem relationalen oder einem nicht-relationalen Datenbanktyp (Nein) fortzufahren. SQL [Weitere Informationen finden Sie unter Von bis Nein. SQL SQL](#)

Aufwand für den Implementierungsplan: Hoch

Ressourcen

Zugehörige bewährte Methoden:

- [REL03-BP02 Entwickeln von Services, die sich auf bestimmte Geschäftsdomänen und Funktionen konzentrieren](#)

Zugehörige Dokumente:

- [Amazon API Gateway: Konfiguration eines REST API mit Open API](#)
- [Was ist eine serviceorientierte Architektur?](#)

- [Bounded Context \(a central pattern in Domain-Driven Design\)](#)
- [Implementierung von Microservices auf AWS](#)
- [Microservice Trade-Offs](#)
- [Microservices - a definition of this new architectural term](#)
- [Microservices auf AWS](#)
- [Was ist AWS App Mesh?](#)

Zugehörige Beispiele:

- [Workshop für die iterative App-Modernisierung](#)

Zugehörige Videos:

- [Exzellenz mit Microservices erreichen AWS](#)

REL03-BP02 Entwickeln von Services, die sich auf bestimmte Geschäftsdomänen und Funktionen konzentrieren

Eine serviceorientierte Architektur (SOA) definiert Services mit genau abgegrenzten Funktionen, die von Geschäftsanforderungen definiert werden. Microservices verwenden Domain-Modelle und begrenzten Kontext, um Servicegrenzen entlang der Grenzen des Geschäftskontextes zu ziehen. Die Konzentration auf Geschäftsdomänen und Funktionen hilft Teams dabei, unabhängige Zuverlässigkeitsanforderungen für ihre Services zu definieren. Begrenzte Kontexte isolieren und kapseln die Geschäftslogik, sodass Teams besser überlegen können, wie mit Fehlern umzugehen ist.

Gewünschtes Ergebnis: Ingenieure und geschäftliche Interessenvertreter definieren gemeinsam begrenzte Kontexte und verwenden sie, um Systeme als Services zu entwerfen, die bestimmte Geschäftsfunktionen erfüllen. Diese Teams verwenden etablierte Praktiken wie Event Storming, um Anforderungen zu definieren. Neue Anwendungen sind als Services mit klar definierten Grenzen und losen Verkopplungen definiert. Bestehende Monolithen werden in [begrenzte Kontexte](#) zerlegt und Systemdesigns bewegen sich in Richtung SOA- oder Microservice-Architekturen. Bei der Refaktorisierung von Monolithen kommen etablierte Ansätze wie Bubble-Kontexte und Monolith-Zerlegung zur Anwendung.

Domain-orientierte Services werden als ein oder mehrere Prozesse ausgeführt, die keinen gemeinsamen Zustand haben. Sie reagieren selbstständig auf Nachfrageschwankungen und behandeln Störszenarien anhand Domain-spezifischer Anforderungen.

Typische Anti-Muster:

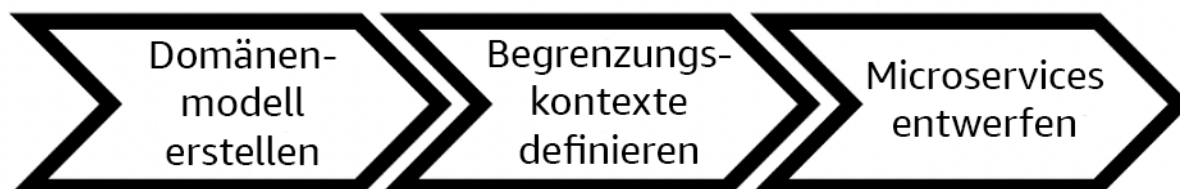
- Teams werden für bestimmte technische Bereiche wie UI und UX, Middleware oder Datenbank gebildet, anstatt für bestimmte Geschäftsdomänen.
- Anwendungen erstrecken sich über die Zuständigkeiten der einzelnen Domains. Services, die sich über begrenzte Kontexte erstrecken, können schwieriger zu verwalten sein, erfordern einen größeren Testaufwand und erfordern die Teilnahme mehrerer Domain-Teams an Softwareupdates.
- Domänenabhängigkeiten wie Domain-Entity-Bibliotheken werden von allen Services gemeinsam genutzt, sodass Änderungen für eine Servicedomäne Änderungen an anderen Service-Domänen erfordern.
- Serviceverträge und Geschäftslogik formulieren Entitäten nicht in einer gemeinsamen und konsistenten Domain-Sprache, was zu Übersetzungsebenen führt, die Systeme komplizieren und den Debugging-Aufwand erhöhen.

Vorteile der Nutzung dieser bewährten Methode: Anwendungen sind als unabhängige Services konzipiert, die durch Geschäftsdomänen begrenzt sind und eine gemeinsame Geschäftssprache verwenden. Services sind unabhängig voneinander testbar und einsetzbar. Services erfüllen die Domain-spezifischen Resilienzanforderungen für die implementierte Domain.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Domain-driven Design (DDD, domänengesteuertes Design) ist der grundlegende Ansatz für das Entwerfen und Entwickeln von Software rund um Geschäftsdomänen. Bei der Entwicklung von Services, die sich auf Geschäftsdomänen konzentrieren, ist es hilfreich, mit einem vorhandenen Framework zu arbeiten. Wenn Sie mit bestehenden monolithischen Anwendungen arbeiten, können Sie die Vorteile von Zerlegungsmustern nutzen, die etablierte Techniken zur Modernisierung von Anwendungen in Services bereitstellen.



Domain-gesteuertes Design

Implementierungsschritte

- Teams können [Event-Storming-Workshops](#) veranstalten, um rasch Ereignisse, Befehle, Mengen und Domänen in einem unkomplizierten Notizformat zu sammeln.
- Sobald Domain-Entitäten und -Funktionen in einem Domain-Kontext gebildet wurden, können Sie Ihre Domain mithilfe eines [begrenzten Kontexts](#), weiter in kleinere Modelle unterteilt, wobei Entitäten mit ähnlichen Funktionen und Attributen in Gruppen sortiert werden. Wenn das Modell in Kontexte unterteilt ist, entsteht eine Vorlage für die Begrenzung von Microservices.
 - Für die Website Amazon.com können Entitäten beispielsweise Pakete, Zustellung, Zeitplan, Preise, Rabatte und Währung enthalten.
 - Paket, Zustellung und Zeitplan werden dem Versandkontext zugeordnet, während Preis, Rabatt und Währung dem Preiskontext zugeordnet sind.
- Unter [Decomposing monoliths into microservices](#) wird das Muster für das Refactoring von Microservices skizziert. Die Verwendung von Mustern für die Unterteilung nach Geschäftsfähigkeit, Subdomäne oder Transaktion passt gut zu Domain-gesteuerten Ansätzen.
- Taktische Techniken wie der [Bubble-Kontext](#) ermöglichen es Ihnen, DDD in bestehenden oder älteren Anwendungen einzuführen, ohne dass Sie im Voraus Änderungen vornehmen und sich voll und ganz auf DDD verlassen müssen. Bei einem Bubble-Kontext-Ansatz wird mithilfe von Service-Mapping und -koordination ein kleiner begrenzter Kontext oder eine [Ebene zur Korruptionsbekämpfung](#) erstellt, die das neu definierte Domain-Modell vor äußeren Einflüssen schützt.

Nachdem die Teams eine Domain-Analyse durchgeführt und Entitäten und Serviceverträge definiert haben, können sie AWS-Services nutzen, um ihr Domain-gesteuertes Design als Cloud-basierte Services zu implementieren.

- Beginnen Sie Ihre Entwicklung, indem Sie Tests definieren, die die Geschäftsregeln Ihrer Domain anwenden. Test-driven Development (TDD, Testgetriebene Entwicklung) und Behavior-driven Development (BDD, verhaltensgetriebene Entwicklung) helfen Teams dabei, die Services auf die Lösung von Geschäftsproblemen zu konzentrieren.
- Wählen Sie [AWS-Services](#) die den Anforderungen Ihrer Geschäfts-Domain und Ihrer [Microservice-Architektur am besten entsprechen](#):
 - [AWS Serverless](#) ermöglicht es Ihrem Team, sich auf eine bestimmte Domain-Logik zu konzentrieren, anstatt Server und Infrastruktur zu verwalten.

- [Container in AWS](#) vereinfachen die Verwaltung Ihrer Infrastruktur, sodass Sie sich auf Ihre Domain-Anforderungen konzentrieren können.
- [Speziell entwickelte Datenbanken](#) helfen Ihnen dabei, Ihre Domain-Anforderungen dem am besten geeigneten Datenbanktyp zuzuordnen.
- [Hexagonale Architekturen auf AWS](#) skizzieren ein Framework zur Integration von Geschäftslogik in Services. Dabei wird rückwärts von der Geschäfts-Domain aus gearbeitet, um funktionale Anforderungen zu erfüllen und dann Integrationsadapter zu implementieren. Muster, die Schnittstellendetails von der Geschäftslogik mit AWS-Services trennen, helfen Teams, sich auf die Funktionalität der Domain zu konzentrieren und die Softwarequalität zu verbessern.

Ressourcen

Zugehörige bewährte Methoden:

- [REL03-BP01 Wählen Sie, wie Sie Ihre Arbeitslast segmentieren möchten](#)
- [REL03-BP03 Stellen Sie Serviceverträge bereit per API](#)

Zugehörige Dokumente:

- [AWS Microservices](#)
- [Implementieren von Microservices in AWS](#)
- [How to break a Monolith into Microservices](#)
- [Getting Started with DDD when Surrounded by Legacy Systems](#)
- [Domain-Driven Design: Tackling Complexity in the Heart of Software](#)
- [Building hexagonal architectures on AWS](#)
- [Decomposing monoliths into microservices](#)
- [Event Storming](#)
- [Messages Between Bounded Contexts](#)
- [Microservices](#)
- [Testgetriebene Entwicklung](#)
- [Verhaltensgetriebene Entwicklung](#)

Zugehörige Beispiele:

- [Designing Cloud Native Microservices on AWS \(from DDD/EventStormingWorkshop\)](#)

Zugehörige Tools:

- [AWS Cloud-Datenbanken](#)
- [Serverless in AWS](#)
- [Container in AWS](#)

REL03-BP03 Stellen Sie Serviceverträge bereit per API

Dienstleistungsverträge sind dokumentierte Vereinbarungen zwischen API Herstellern und Verbrauchern, die in einer maschinenlesbaren API Definition definiert sind. Eine Strategie zur Versionierung von Verträgen ermöglicht es den Verbrauchern, die vorhandenen weiterhin zu verwenden API und ihre Anwendungen auf eine neuere zu migrieren, API sobald sie bereit sind. Die Bereitstellung durch den Produzenten kann jederzeit erfolgen, solange der Vertrag eingehalten wird. Serviceteams können den Technologie-Stack ihrer Wahl verwenden, um den API Vertrag zu erfüllen.

Gewünschtes Ergebnis: Anwendungen, die mit serviceorientierten Architekturen oder Microservice-Architekturen erstellt wurden, können unabhängig voneinander betrieben werden und verfügen gleichzeitig über eine integrierte Laufzeitabhängigkeit. Änderungen, die an einem API Verbraucher oder Hersteller vorgenommen werden, beeinträchtigen nicht die Stabilität des Gesamtsystems, wenn beide Seiten einen gemeinsamen Vertrag einhalten. API Komponenten, die über den Service kommunizieren, APIs können unabhängige funktionale Releases durchführen, Upgrades auf Laufzeitabhängigkeiten durchführen oder ein Failover zu einem Disaster Recovery (DR) -Standort durchführen, ohne sich gegenseitig zu beeinträchtigen. Darüber hinaus können spezialisierte Services unabhängig voneinander skaliert werden und können dabei den Ressourcenbedarf absorbieren, ohne dass andere Services ebenfalls skaliert werden müssen.

Typische Anti-Muster:

- Ein Dienst wird APIs ohne stark typisierte Schemas erstellt. Dies führt dazu APIs, dass sie nicht zum Generieren von API Bindungen und Nutzlasten verwendet werden können, die nicht programmgesteuert validiert werden können.
- Es wird keine Versionierungsstrategie angewendet, die API Verbraucher dazu zwingt, Updates und Releases vorzunehmen, andernfalls scheitern sie, wenn sich die Serviceverträge weiterentwickeln.
- Fehlermeldungen, die Details der zugrundeliegenden Service-Implementierung preisgeben, anstatt Integrationsfehler im Kontext und in der Sprache der Domain zu beschreiben.

- Keine Nutzung von API Verträgen zur Entwicklung von Testfällen und API Scheinimplementierungen, um unabhängige Tests von Servicekomponenten zu ermöglichen.

Vorteile der Einführung dieser bewährten Methode: Verteilte Systeme, die aus Komponenten bestehen, die über API Serviceverträge miteinander kommunizieren, können die Zuverlässigkeit verbessern. Entwickler können potenzielle Probleme frühzeitig im Entwicklungsprozess catch, indem sie während der Kompilierung eine Typprüfung durchführen, um sicherzustellen, dass Anfragen und Antworten dem API Vertrag entsprechen und die erforderlichen Felder vorhanden sind. APIVerträge bieten eine klare, sich selbst dokumentierende Schnittstelle für eine bessere Interoperabilität zwischen verschiedenen Systemen APIs und Programmiersprachen und sorgen für eine bessere Interoperabilität.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Sobald Sie Geschäftsbereiche identifiziert und Ihre Workload-Segmentierung festgelegt haben, können Sie Ihren Service weiterentwickeln. APIs Definieren Sie zunächst maschinenlesbare Serviceverträge für APIs und implementieren Sie anschließend eine API Versionierungsstrategie. Wenn Sie bereit sind REST, Dienste über gängige Protokolle wie GraphQL oder asynchrone Ereignisse zu integrieren, können Sie AWS Dienste in Ihre Architektur integrieren, um Ihre Komponenten mit stark API typisierten Verträgen zu integrieren.

AWS Dienste für Serviceverträge API

Integrieren Sie AWS Services wie [Amazon API Gateway](#) und [Amazon EventBridge](#) in Ihre Architektur [AWS AppSync](#), um API Serviceverträge in Ihrer Anwendung zu verwenden. Amazon API Gateway unterstützt Sie bei der direkten Integration mit nativen AWS Diensten und anderen Webdiensten. APIGateway unterstützt die [APIOpen-Spezifikation](#) und Versionierung. AWS AppSync ist ein verwalteter [GraphQL-Endpunkt](#), den Sie konfigurieren, indem Sie ein GraphQL-Schema definieren, um eine Serviceschnittstelle für Abfragen, Mutationen und Abonnements zu definieren. Amazon EventBridge verwendet Ereignisschemas, um Ereignisse zu definieren und Codebindungen für Ihre Ereignisse zu generieren.

Implementierungsschritte

- Definieren Sie zunächst einen Vertrag für Ihren. API Ein Vertrag drückt die Fähigkeiten eines aus und API definiert stark typisierte Datenobjekte und Felder für die API Eingabe und Ausgabe.

- Bei der Konfiguration APIs in API Gateway können Sie Open API Specifications für Ihre Endgeräte importieren und exportieren.
 - Der [Import einer offenen API Definition](#) vereinfacht die Erstellung Ihrer Definition API und kann als Codetools wie [AWS Serverless Application Model](#) und [AWS Cloud Development Kit \(AWS CDK\)](#) in die AWS Infrastruktur integriert werden.
 - Das [Exportieren einer API Definition](#) vereinfacht die Integration mit API Testtools und bietet dem Servicenutzer eine Integrationsspezifikation.
- Sie können APIs GraphQL definieren und verwalten, AWS AppSync indem Sie [eine GraphQL-Schemadatei definieren](#), um Ihre Vertragsschnittstelle zu generieren und die Interaktion mit komplexen REST Modellen, mehreren Datenbanktabellen oder älteren Diensten zu vereinfachen.
- [AWS Amplify](#) Projekte, die integriert sind, AWS AppSync generieren stark typisierte JavaScript Abfragedateien zur Verwendung in Ihrer Anwendung sowie eine AWS AppSync GraphQL-Clientbibliothek für [Amazon DynamoDB-Tabellen](#).
- Wenn Sie Serviceereignisse von Amazon nutzen EventBridge, entsprechen Ereignisse Schemas, die bereits in der Schemaregistrierung vorhanden sind oder die Sie mit der Open API Spec definieren. Mit einem in der Registrierung definierten Schema können Sie auch Client-Bindungen aus dem Schemavertrag generieren, um Ihren Code in Ereignisse zu integrieren.
- Erweiterung oder Versionierung Ihrer API Das Erweitern von API ist eine einfachere Option, wenn Felder hinzugefügt werden, die mit optionalen Feldern oder Standardwerten für Pflichtfelder konfiguriert werden können.
 - JSONbasierte Verträge für Protokolle wie REST GraphQL können sich gut für eine Vertragsverlängerung eignen.
 - XMLVerträge, die auf Protokollen basieren, SOAP sollten mit den Nutzern der Dienste getestet werden, um festzustellen, ob eine Vertragsverlängerung durchführbar ist.
- Bei der Versionierung einer sollten Sie die Implementierung einer Proxy-Versionierung in Betracht ziehenAPI, bei der eine Fassade zur Unterstützung von Versionen verwendet wird, sodass die Logik in einer einzigen Codebasis verwaltet werden kann.
 - Mit API Gateway können Sie [Anfragen- und Antwortzuordnungen](#) verwenden, um die Übernahme von Vertragsänderungen zu vereinfachen, indem Sie eine Fassade einrichten, um Standardwerte für neue Felder bereitzustellen oder entfernte Felder aus einer Anfrage oder Antwort zu entfernen. Mit diesem Ansatz kann der zugrunde liegende Service mit einer einzelnen Codebasis betrieben werden.

Ressourcen

Zugehörige bewährte Methoden:

- [REL03-BP01 Wählen Sie, wie Sie Ihre Arbeitslast segmentieren möchten](#)
- [REL03-BP02 Entwickeln von Services, die sich auf bestimmte Geschäftsdomänen und Funktionen konzentrieren](#)
- [REL04-BP02 Lose gekoppelte Abhängigkeiten implementieren](#)
- [REL05-BP03 Steuerung und Begrenzung von Wiederholungsaufrufen](#)
- [REL05-BP05 Client-Timeouts festlegen](#)

Zugehörige Dokumente:

- [Was ist eine API \(Anwendungsprogrammierschnittstelle\)?](#)
- [Implementierung von Microservices auf AWS](#)
- [Microservice Trade-Offs](#)
- [Microservices - a definition of this new architectural term](#)
- [Microservices auf AWS](#)
- [Ich arbeite mit API Gateway-Erweiterungen für Open API](#)
- [APIOpen-Spezifikation](#)
- [GraphQL: Schemata und Typen](#)
- [EventBridge Amazon-Codebindungen](#)

Zugehörige Beispiele:

- [Amazon API Gateway: Konfiguration eines REST API mit Open API](#)
- [CRUDAnwendung Amazon API Gateway to Amazon DynamoDB mithilfe von Open API](#)
- [Moderne Anwendungsintegrationsmuster in einem serverlosen Zeitalter: API Gateway Service Integration](#)
- [Implementierung einer Header-basierten API Gateway-Versionierung mit Amazon CloudFront](#)
- [AWS AppSync: Building a client application](#)

Zugehörige Videos:

- [Verwenden von Open API in AWS SAM zur Verwaltung von Gateway API](#)

Zugehörige Tools:

- [APIAmazon-Gateway](#)
- [AWS AppSync](#)
- [Amazon EventBridge](#)

REL 4. Wie lassen sich Interaktionen in einem verteilten System so gestalten, dass Ausfälle vermieden werden?

Verteilte Systeme sind auf Kommunikationsnetzwerke angewiesen, um Komponenten wie Server oder Services miteinander zu verbinden. Ihre Workload muss trotz Datenverlust oder Latenz in diesen Netzwerken zuverlässig funktionieren. Die Komponenten des verteilten Systems müssen so funktionieren, dass sie sich nicht negativ auf andere Komponenten oder die Workload auswirken. Diese bewährten Methoden verhindern Ausfälle und verbessern die mittlere Betriebsdauer zwischen Ausfällen (MTBF).

Best Practices

- [REL04-BP01 Bestimmen Sie, von welcher Art von verteilten Systemen Sie abhängig sind](#)
- [REL04-BP02 Lose gekoppelte Abhängigkeiten implementieren](#)
- [REL04-BP03 Arbeite ständig](#)
- [REL04-BP04 Festlegen von Mutationsoperationen als idempotent](#)

REL04-BP01 Bestimmen Sie, von welcher Art von verteilten Systemen Sie abhängig sind

Verteilte Systeme verwenden die synchrone, asynchrone oder Stapelverarbeitung. Synchrone Systeme müssen Anfragen so schnell wie möglich verarbeiten und miteinander kommunizieren, indem sie synchrone Anfrage- und Antwortaufrufe mithilfe von HTTP/S-, REST- oder RPC-Protokollen (Remote Procedure Call) durchführen. Asynchrone Systeme kommunizieren miteinander, indem sie Daten asynchron über einen Zwischenservice austauschen, ohne einzelne Systeme zu koppeln. Systeme mit Stapelverarbeitung empfangen eine große Menge an Eingabedaten, führen automatisierte Datenprozesse ohne menschliches Eingreifen aus und generieren Ausgabedaten.

Gewünschtes Ergebnis: Entwerfen Sie einen Workload, der effektiv mit synchronen, asynchronen und Batch-Abhängigkeiten interagiert.

Typische Anti-Muster:

- Der Workload wartet auf unbestimmte Zeit auf eine Antwort von seinen Abhängigkeiten, was dazu führen kann, dass Workload-Clients das Zeitlimit überschreiten und nicht wissen, ob ihre Anfrage eingegangen ist.
- Der Workload verwendet eine Kette von abhängigen Systemen, die sich gegenseitig synchron aufrufen. Der Erfolg der gesamten Kette hängt davon ab, dass jedes System verfügbar ist und Anfragen erfolgreich verarbeitet, was zu instabilem Verhalten und eingeschränkter Gesamtverfügbarkeit führen kann.
- Der Workload kommuniziert asynchron mit seinen Abhängigkeiten und stützt sich auf das Konzept der garantierten einmaligen Zustellung von Nachrichten, obwohl es oft immer noch möglich ist, doppelte Nachrichten zu empfangen.
- Der Workload verwendet keine geeigneten Tools zur Batchplanung und ermöglicht die gleichzeitige Ausführung desselben Batchjobs.

Vorteile der Nutzung dieser bewährten Methode: Es ist üblich, dass ein bestimmter Workload einen oder mehrere Kommunikationsstile der synchronen, asynchronen und Stapelverarbeitung implementiert. Diese bewährte Methode hilft Ihnen dabei, die verschiedenen Kompromisse zu identifizieren, die mit den einzelnen Kommunikationsstilen verbunden sind, damit Ihr Workload Störungen in allen Abhängigkeiten tolerieren kann.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Die folgenden Abschnitte enthalten sowohl allgemeine als auch spezifische Implementierungshinweise für jede Art von Abhängigkeit.

General guidance (Allgemeine Anleitung)

- Stellen Sie sicher, dass die Leistungs- und Zuverlässigkeits-Servicelevel-Ziele (SLOs), die Ihre Abhängigkeiten bieten, den Leistungs- und Zuverlässigkeitsanforderungen Ihres Workloads entsprechen.
- Verwenden Sie [AWS Observability Services](#), um [Reaktionszeiten und Fehlerraten zu überwachen](#), und so sicherzustellen, dass Ihre Abhängigkeit den von Ihrem Workload benötigten Service bietet.
- Identifizieren Sie die potenziellen Herausforderungen, mit denen Ihr Workload bei der Kommunikation mit seinen Abhängigkeiten konfrontiert sein könnte. Verteilte Systeme [sind mit einer Vielzahl von Herausforderungen verbunden](#), die die architektonische Komplexität,

den Betriebsaufwand und die Kosten erhöhen können. Zu den häufigsten Herausforderungen gehören Latenz, Netzwerkunterbrechungen, Datenverlust, Skalierung und Verzögerungen bei der Datenreplikation.

- Implementieren Sie eine robuste Fehlerbehandlung und [-protokollierung](#), um Probleme zu beheben, wenn es in Ihrer Abhängigkeit zu Problemen kommt.

Synchrone Abhängigkeit

Bei synchroner Kommunikation sendet Ihr Workload eine Anfrage an seine Abhängigkeit und blockiert den Vorgang, der auf eine Antwort wartet. Wenn ihre Abhängigkeit die Anfrage erhält, versucht sie, sie so schnell wie möglich zu bearbeiten, und sendet eine Antwort zurück an den Workload. Eine große Herausforderung bei synchroner Kommunikation besteht darin, dass sie zu einer zeitlichen Kopplung führt, was erfordert, dass der Workload und dessen Abhängigkeiten gleichzeitig verfügbar sind. Beachten Sie die folgenden Hinweise, wenn der Workload synchron mit seinen Abhängigkeiten kommunizieren muss:

- Der Workload sollte sich nicht auf mehrere synchrone Abhängigkeiten stützen, um eine einzelne Funktion auszuführen. Diese Kette von Abhängigkeiten erhöht die allgemeine Instabilität, da alle Abhängigkeiten im Pfad verfügbar sein müssen, damit die Anfrage erfolgreich abgeschlossen werden kann.
- Wenn eine Abhängigkeit fehlerhaft oder nicht verfügbar ist, bestimmen Sie Ihre Strategie zur Fehlerbehandlung und versuchen Sie es erneut. Vermeiden Sie bimodales Verhalten. Bimodales Verhalten liegt vor, wenn sich der Workload im Normalmodus und im Fehlermodus unterschiedlich verhält. Weitere Informationen zu bimodalem Verhalten finden Sie unter [REL11-BP05 Verhindern von bimodalem Verhalten mithilfe statischer Stabilität](#).
- Denken Sie daran, dass es besser ist, schnell zu scheitern, als Ihren Workload warten zu lassen. Im [AWS Lambda Entwicklerleitfaden](#) wird beispielsweise beschrieben, wie Wiederholungen und Fehlschläge beim Aufrufen von Lambda-Funktionen behandelt werden.
- Legen Sie Timeouts fest, wenn Ihr Workload seine Abhängigkeit aufruft. Dadurch wird vermieden, zu lange oder unbegrenzt auf eine Antwort zu warten. Eine hilfreiche Diskussion zu diesem Problem finden Sie unter [Tuning AWS Java SDK HTTP request settings for latency-aware Amazon DynamoDB applications](#).
- Reduzieren Sie die Anzahl der Aufrufe von Ihrem Workload an seine Abhängigkeit, um eine einzelne Anfrage zu erfüllen. Durch zahlreiche Aufrufe erhöht sich die Kopplung und Latenz.

Asynchrone Abhängigkeit

Um den Workload zeitlich von dessen Abhängigkeiten zu entkoppeln, sollte die Kommunikation asynchron erfolgen. Bei einem asynchronen Ansatz kann der Workload mit jeder anderen Verarbeitung fortfahren, ohne auf eine Antwort der Abhängigkeit oder Kette von Abhängigkeiten warten zu müssen.

Beachten Sie die folgenden Hinweise, wenn der Workload asynchron mit seiner Abhängigkeit kommunizieren muss:

- Entscheiden Sie je nach Anwendungsfall und Anforderungen, ob Sie Messaging oder Ereignis-Streaming verwenden möchten. Beim [Messaging](#) kann der Workload mit seiner Abhängigkeit kommunizieren, indem er Nachrichten über einen Message Broker sendet und empfängt. Beim [Ereignis-Streaming](#) können der Workload und seine Abhängigkeiten einen Streaming-Dienst verwenden, um Ereignisse zu veröffentlichen und zu abonnieren, die als kontinuierliche Datenströme bereitgestellt werden und so schnell wie möglich verarbeitet werden müssen.
- Messaging und Ereignis-Streaming behandeln Nachrichten unterschiedlich, sodass Sie basierend auf den folgenden Faktoren Abwägungsentscheidungen treffen müssen:
 - Nachrichtenpriorität: Message Broker können Nachrichten mit hoher Priorität vor normalen Nachrichten verarbeiten. Beim Ereignis-Streaming haben alle Nachrichten dieselbe Priorität.
 - Nachrichtenverbrauch: Message Broker stellen sicher, dass die Verbraucher die Nachricht erhalten. Ereignis-Streaming-Verbraucher müssen den Überblick über die zuletzt gelesene Nachricht behalten.
 - Nachrichtenreihenfolge: Beim Messaging ist der Empfang von Nachrichten in der genauen Reihenfolge, in der sie gesendet wurden, nicht garantiert, es sei denn, Sie verwenden einen FIFO-Ansatz (First-in-First-Out). Beim Ereignis-Streaming wird immer die Reihenfolge beibehalten, in der die Daten erzeugt wurden.
 - Löschen von Nachrichten: Beim Messaging muss der Verbraucher die Nachricht nach der Verarbeitung löschen. Der Ereignis-Streaming-Dienst hängt die Nachricht an einen Stream an und verbleibt dort, bis die Aufbewahrungsfrist der Nachricht abläuft. Diese Löschrichtlinie macht das Ereignis-Streaming für die Wiedergabe von Nachrichten geeignet.
- Definieren Sie, wie Ihr Workload weiß, wann seine Abhängigkeit seine Arbeit beendet hat. Wenn der Workload beispielsweise [asynchron eine Lambda-Funktion](#) aufruft, stellt Lambda das Ereignis in eine Warteschlange und gibt eine Erfolgsantwort ohne zusätzliche Informationen zurück. Nach Abschluss der Verarbeitung kann die Lambda-Funktion das [Ergebnis an ein Ziel senden](#), das je nach Erfolg oder Misserfolg konfiguriert werden kann.

- Entwickeln Sie Ihren Workload so, dass er doppelte Nachrichten verarbeiten kann, indem Sie Idempotenz nutzen. Idempotenz bedeutet, dass sich die Ergebnisse des Workloads nicht ändern, auch wenn der Workload mehrmals für dieselbe Nachricht generiert wird. Es ist wichtig, darauf hinzuweisen, dass [Messaging](#)- oder [Streaming](#)-Dienste eine Nachricht erneut übermitteln, wenn ein Netzwerkausfall auftritt oder wenn keine Bestätigung eingegangen ist.
- Wenn Ihr Workload keine Antwort von seiner Abhängigkeit erhält, muss er die Anfrage erneut einreichen. Erwägen Sie, die Anzahl der Wiederholungsversuche zu begrenzen, um die CPU-, Arbeitsspeicher- und Netzwerkressourcen des Workloads für die Bearbeitung anderer Anfragen zu schonen. In der [AWS Lambda-Dokumentation](#) wird gezeigt, wie Fehler beim asynchronen Aufruf behandelt werden.
- Nutzen Sie geeignete Beobachtbarkeits-, Debugging- und Tracing-Tools, um die asynchrone Kommunikation des Workloads mit seinen Abhängigkeiten zu verwalten und zu betreiben. Sie können [Amazon CloudWatch](#) verwenden, um [Nachrichten](#)- und [Ereignis-Streaming-Dienste](#) zu überwachen. Sie können auch Ihren Workload mit [AWS X-Ray](#) instrumentieren, um schnell [Erkenntnisse zur Problembehebung](#) zu gewinnen.

Batch-Abhängigkeit

Batch-Systeme nehmen Eingabedaten auf, initiieren eine Reihe von Aufgaben, um sie zu verarbeiten, und erzeugen einige Ausgabedaten, ohne dass manuelles Eingreifen erforderlich ist. Je nach Datengröße können Aufgaben in wenigen Minuten oder in einigen Fällen sogar in mehreren Tagen ausgeführt werden. Beachten Sie die folgenden Hinweise, wenn der Workload mit seiner Batch-Abhängigkeit kommuniziert:

- Definieren Sie das Zeitfenster, in dem der Workload den Batchjob ausführen soll. Der Workload kann ein Wiederholungsmuster einrichten, um ein Batchsystem aufzurufen, beispielsweise jede Stunde oder am Ende eines jeden Monats.
- Ermitteln Sie den Ort der Dateneingabe und der verarbeiteten Datenausgabe. Wählen Sie einen Speicherservice wie [Amazon Simple Storage Services \(Amazon S3\)](#), [Amazon Elastic File System \(Amazon EFS\)](#) und [Amazon FSx for Lustre](#), der es Ihrem Workload ermöglicht, Dateien in großem Umfang zu lesen und zu schreiben.
- Wenn Ihr Workload mehrere Batchjobs aufrufen muss, können Sie [AWS Step Functions](#) nutzen, um die Orchestrierung von Batchjobs, die in AWS oder On-Premises ausgeführt werden, zu vereinfachen. In diesem [Beispielprojekt](#) wird die Orchestrierung von Batchjobs mithilfe von Step Functions, [AWS Batch](#) und Lambda demonstriert.

- Überwachen Sie Batchjobs, um nach Auffälligkeiten zu suchen, z. B. wenn die Ausführung eines Jobs länger dauert, als sie sollte. Sie könnten Tools wie [CloudWatch Container Insights](#) verwenden, um AWS Batch-Umgebungen und Jobs zu überwachen. In diesem Fall würde Ihr Workload den Beginn des nächsten Jobs unterbrechen und die zuständigen Mitarbeiter über die Ausnahme informieren.

Ressourcen

Zugehörige Dokumente:

- [AWS Cloud-Operationen: Überwachung und Beobachtbarkeit](#)
- [Die Amazon Builders' Library: Herausforderungen für verteilte Systeme](#)
- [REL11-BP05 Verhindern von bimodalem Verhalten mithilfe statischer Stabilität](#)
- [AWS Lambda-Entwicklerhandbuch: Fehlerbehandlung und automatische Wiederholungsversuche in AWS Lambda](#)
- [Tuning AWS Java SDK HTTP request settings for latency-aware Amazon DynamoDB applications](#)
- [AWS-Messaging](#)
- [Was sind Streaming-Daten?](#)
- [AWS Lambda-Entwicklerhandbuch: Asynchroner Aufruf](#)
- [Amazon Simple Queue Service FAQ: FIFO-Warteschlangen](#)
- [Amazon Kinesis Data Streams-Entwicklerhandbuch: Umgang mit doppelten Datensätzen](#)
- [Amazon Simple Queue Service-Entwicklerhandbuch: Verfügbare CloudWatch-Metriken für Amazon SQS](#)
- [Amazon Kinesis Data Streams-Entwicklerhandbuch: Überwachung des Amazon Kinesis Data Streams-Service mit Amazon CloudWatch](#)
- [AWS X-Ray-Entwicklerhandbuch: AWS X-Ray-Konzepte](#)
- [AWS Beispiele auf GitHub: AWS Step functions Complex Orchestrator App](#)
- [AWS Batch-Benutzerhandbuch: AWS Batch CloudWatch Container Insights](#)

Zugehörige Videos:

- [AWS Summit SF 2022 - Full-stack observability and application monitoring with AWS \(COP310\)](#)

Zugehörige Tools:

- [Amazon CloudWatch](#)
- [Amazon CloudWatch Logs](#)
- [AWS X-Ray](#)
- [Amazon Simple Storage Service \(Amazon S3\)](#)
- [Amazon Elastic File System \(Amazon EFS\)](#)
- [Amazon FSx für Lustre](#)
- [AWS Step Functions](#)
- [AWS Batch](#)

REL04-BP02 Lose gekoppelte Abhängigkeiten implementieren

Abhängigkeiten etwa zwischen Warteschlangensystemen, Streaming-Systemen, Workflows und Load Balancern sind lose gekoppelt. Eine lose Verkopplung hilft, das Verhalten einer Komponente von anderen Komponenten zu isolieren, die von ihr abhängig sind. Dies verbessert Resilienz und Agilität.

Die Entkopplung von Abhängigkeiten wie Warteschlangensystemen, Streaming-Systemen und Workflows trägt dazu bei, die Auswirkungen von Änderungen oder Ausfällen auf ein System zu minimieren. Durch diese Trennung wird das Verhalten einer Komponente von Auswirkungen auf andere, die von ihr abhängig sind, isoliert und so die Widerstandsfähigkeit und Agilität verbessert.

In eng gekoppelten Systemen können Änderungen an einer Komponente Änderungen an anderen Komponenten erforderlich machen, die von ihr abhängen, was die Leistung aller Komponenten beeinträchtigt. Die lose Verkopplung unterbricht diese Abhängigkeit, sodass abhängige Komponenten nur die versionierte und veröffentlichte Schnittstelle kennen müssen. Die Implementierung einer losen Verkopplung zwischen Abhängigkeiten isoliert einen Ausfall. So wird verhindert, dass er sich auf andere Komponenten auswirkt.

Die lose Verkopplung ermöglicht Ihnen, einer Komponente zusätzlichen Code oder Features hinzuzufügen und gleichzeitig das Risiko für Komponenten zu minimieren, die von ihr abhängig sind. Sie ermöglicht auch eine granulare Ausfallsicherheit auf Komponentenebene, bei der Sie die zugrunde liegende Implementierung der Abhängigkeit aufskalieren oder sogar ändern können.

Um die Ausfallsicherheit durch lose Verkopplung weiter zu verbessern, legen Sie Komponenten-Interaktionen nach Möglichkeit als asynchron fest. Dieses Modell eignet sich für jede Interaktion, bei der keine sofortige Antwort benötigt wird, sondern die Bestätigung ausreicht, dass eine Anfrage registriert wurde. Es umfasst eine Komponente, die Ereignisse generiert, und eine andere Komponente, die sie konsumiert. Die beiden Komponenten werden nicht durch direkte point-to-point

Interaktion integriert, sondern normalerweise über eine dauerhafte Zwischenspeicherschicht, z. B. eine SQS Amazon-Warteschlange, eine Streaming-Datenplattform wie Amazon Kinesis oder AWS Step Functions.

Abbildung 4: Abhängigkeiten etwa zwischen Warteschlangensystemen und Load Balancer sind lose gekoppelt

SQSA Amazon-Warteschlangen und AWS Step Functions sind nur zwei Möglichkeiten, eine Zwischenschicht für lose Kopplung hinzuzufügen. Eventgesteuerte Architekturen können auch AWS Cloud mithilfe von Amazon erstellt werden EventBridge, wodurch Kunden (Event-Produzenten) von den Diensten, auf die sie angewiesen sind (Event-Konsumenten), abstrahieren können. Amazon Simple Notification Service (AmazonSNS) ist eine effektive Lösung, wenn Sie Push-basiertes Messaging mit hohem Durchsatz benötigen. many-to-many Mithilfe von SNS Amazon-Themen können Ihre Publisher-Systeme Nachrichten zur parallel Verarbeitung an eine große Anzahl von Abonnenten-Endpunkten fächern.

Warteschlangen bieten zwar mehrere Vorteile, doch Anfragen, die älter als ein Schwellenwert sind (oft Sekunden), sollten in den meisten harten Echtzeitsystemen als veraltet betrachtet (der Client hat aufgegeben und wartet nicht mehr auf eine Antwort) und nicht verarbeitet werden. Auf diese Weise können stattdessen neuere (und wahrscheinlich noch gültige Anfragen) verarbeitet werden.

Gewünschtes Ergebnis: Wenn Sie lose gekoppelte Abhängigkeiten implementieren, können Sie die Fehlerfläche auf Komponentenebene minimieren, was die Diagnose und Lösung von Problemen erleichtert. Außerdem vereinfacht es die Entwicklungszyklen, da die Teams Änderungen auf modularer Ebene implementieren können, ohne die Leistung anderer Komponenten, die davon abhängen, zu beeinträchtigen. Dieser Ansatz ermöglicht eine Aufskalierung auf Komponentenebene auf Grundlage des Ressourcenbedarfs sowie der Auslastung einer Komponente und trägt so zur Kosteneffizienz bei.

Typische Anti-Muster:

- Bereitstellen eines monolithischen Workloads.
- Direkter Aufruf APIs zwischen Workload-Stufen, ohne dass ein Failover oder eine asynchrone Verarbeitung der Anfrage möglich ist.
- Enge Verknüpfung mithilfe gemeinsam genutzter Daten. Lose gekoppelte Systeme sollten die gemeinsame Nutzung von Daten durch gemeinsam genutzte Datenbanken oder andere Formen der eng gekoppelten Datenspeicherung vermeiden, da dies wieder zu einer engen Verknüpfung führen und die Skalierbarkeit behindern kann.

- Gegendruck wird ignoriert. Ihr Workload sollte in der Lage sein, die eingehenden Daten zu verlangsamen oder zu stoppen, wenn eine Komponente sie nicht mit der gleichen Geschwindigkeit verarbeiten kann.

Vorteile der Nutzung dieser bewährten Methode: Eine lose Verkoppelung hilft dabei, das Verhalten einer Komponente von anderen Komponenten zu isolieren, die von ihr abhängen, wodurch die Resilienz und Agilität erhöht werden. Fehler in einer Komponente sind von anderen isoliert.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Implementieren lose gekoppelter Abhängigkeiten. Es gibt verschiedene Lösungen, mit denen Sie lose gekoppelte Anwendungen erstellen können. Dazu gehören Dienste für die Implementierung vollständig verwalteter Warteschlangen, automatisierter Workflows, die Reaktion auf Ereignisse und APIs vieles mehr, die dazu beitragen können, das Verhalten von Komponenten von anderen Komponenten zu isolieren und so die Widerstandsfähigkeit und Agilität zu erhöhen.

- Erstellen Sie ereignisgesteuerte Architekturen: [Amazon EventBridge](#) hilft Ihnen beim Aufbau lose gekoppelter und verteilter ereignisgesteuerter Architekturen.
- Implementieren Sie Warteschlangen in verteilten Systemen: Sie können [Amazon Simple Queue Service \(AmazonSQS\)](#) verwenden, um verteilte Systeme zu integrieren und zu entkoppeln.
- Komponenten als Microservices containerisieren: [Microservices](#) ermöglichen es Teams, Anwendungen zu erstellen, die aus kleinen unabhängigen Komponenten bestehen, die über klar definierte Kanäle kommunizieren. APIs [Amazon Elastic Container Service \(AmazonECS\)](#) und [Amazon Elastic Kubernetes Service \(AmazonEKS\)](#) können Ihnen helfen, schneller mit Containern zu beginnen.
- Verwalten Sie Workflows mit Step Functions: Mit [Step Functions](#) können Sie mehrere AWS Services zu flexiblen Workflows koordinieren.
- Nutzen Sie die Messaging-Architekturen Publish-Subscribe (Pub/Sub): [Amazon Simple Notification Service \(AmazonSNS\)](#) ermöglicht die Nachrichtenzustellung von Verlagen an Abonnenten (auch bekannt als Produzenten und Verbraucher).

Implementierungsschritte

- Komponenten in einer ereignisgesteuerten Architektur werden durch Ereignisse ausgelöst. Ereignisse sind Aktionen, die in einem System stattfinden, z. B. wenn ein Benutzer einen Artikel in

den Warenkorb legt. Wenn eine Aktion erfolgreich ist, wird ein Ereignis erzeugt, das die nächste Komponente des Systems auslöst.

- [Entwicklung ereignisgesteuerter Anwendungen mit Amazon EventBridge](#)
- [AWS re:Invent 2022 — Entwicklung ereignisgesteuerter Integrationen mit Amazon EventBridge](#)
- Verteilte Nachrichtensysteme haben drei Hauptbestandteile, die für eine Warteschlangenbasierte Architektur implementiert werden müssen. Sie umfassen Komponenten des verteilten Systems, die Warteschlange, die für die Entkopplung verwendet wird (verteilt auf SQS Amazon-Servern), und die Nachrichten in der Warteschlange. Ein typisches System hat einen Produzenten, der die Nachricht in die Warteschlange einstellt, und einen Verbraucher, der die Nachricht aus der Warteschlange empfängt. Die Warteschlange speichert Nachrichten aus Redundanzgründen auf mehreren SQS Amazon-Servern.
 - [Grundlegende SQS Amazon-Architektur](#)
 - [Send Messages Between Distributed Applications with Amazon Simple Queue Service](#)
- Wenn Microservices gut genutzt werden, verbessern sie die Wartbarkeit und die Skalierbarkeit, da lose gekoppelte Komponenten von unabhängigen Teams verwaltet werden. Sie ermöglichen zudem die Isolierung von Verhaltensweisen auf eine einzelne Komponente im Falle von Änderungen.
 - [Implementierung von Microservices auf AWS](#)
 - [Let's Architect! Architektur von Microservices mit Containern](#)
- Mit können AWS Step Functions Sie unter anderem verteilte Anwendungen erstellen, Prozesse automatisieren und Microservices orchestrieren. Die Orchestrierung mehrerer Komponenten in einem automatisierten Workflow ermöglicht es Ihnen, Abhängigkeiten in Ihrer Anwendung zu entkoppeln.
 - [Erstellen Sie einen serverlosen Workflow mit und AWS Step FunctionsAWS Lambda](#)
 - [Erste Schritte mit AWS Step Functions](#)

Ressourcen

Zugehörige Dokumente:

- [AmazonEC2: Idempotenz sicherstellen](#)
- [Die Amazon Builders' Library: Herausforderungen für verteilte Systeme](#)
- [Die Amazon Builders' Library: Zuverlässigkeit, stetige Ausführung und eine gute Tasse Kaffee](#)
- [Was ist Amazon EventBridge?](#)

- [Was ist Amazon Simple Queue Service?](#)
- [Break up with your monolith](#)
- [Orchestrieren Sie warteschlangenbasierte Microservices mit und Amazon AWS Step Functions SQS](#)
- [Grundlegende SQS Amazon-Architektur](#)
- [Warteschlangenbasierte Architektur](#)

Zugehörige Videos:

- [AWS New York Summit 2019: Einführung in ereignisgesteuerte Architekturen und Amazon \(05\) EventBridge MAD2](#)
- [AWS re:Invent 2018: Closed Loops and Opening Minds: Wie man die Kontrolle über große und kleine Systeme übernimmt ARC337 \(beinhaltet lockere Kopplung, konstante Arbeit, statische Stabilität\)](#)
- [AWS re:Invent 2019: Umstellung auf ereignisgesteuerte Architekturen \(08\) SVS3](#)
- [AWS re:Invent 2019: Skalierbare, serverlose, ereignisgesteuerte Anwendungen mit Amazon und Lambda SQS](#)
- [AWS re:Invent 2022 — Entwicklung ereignisgesteuerter Integrationen mit Amazon EventBridge](#)
- [AWS re:Invent 2017: Elastic Load Balancing im Detail und Best Practices](#)

REL04-BP03 Arbeite ständig

Bei größeren, schnellen Lastveränderungen können Systeme ausfallen. Wenn Ihr Workload beispielsweise eine Zustandsprüfung ausführt, die den Zustand vieler tausend Server überwacht, sollte er jedes Mal die gleiche Nutzlast senden (einen vollständigen Snapshot des aktuellen Status). Unabhängig davon, ob keine Server oder alle Server ausfallen, führt das System für die Zustandsprüfung die Aufgaben stetig und ohne große, schnelle Änderungen aus.

Wenn das Zustandsprüfungssystem beispielsweise 100 000 Server überwacht, ist die Last darauf angesichts der normalerweise geringen Serverausfallrate nominal. Wenn jedoch ein großes Ereignis die Hälfte dieser Server fehlerhaft macht, wäre das Zustandsprüfungssystem überfordert, wenn es versucht, Benachrichtigungssysteme zu aktualisieren und den Status an seine Clients zu kommunizieren. Stattdessen sollte das Zustandsprüfungssystem jedes Mal den vollständigen Snapshot des aktuellen Status senden. 100 000 Server-Zustände, die jeweils durch ein Bit dargestellt werden, entsprechen nur eine Nutzlast von 12,5 KB. Unabhängig davon, ob keine oder alle Server

ausfallen – das System für die Zustandsprüfung erledigt seine Arbeit konstant und große, schnelle Änderungen stellen keine Bedrohung für die Systemstabilität dar. Auf diese Weise führt Amazon Route 53 Zustandsprüfungen für Endpunkte (wie z. B. IP-Adressen) durch, um zu ermitteln, wie Endbenutzer an diese weitergeleitet werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Niedrig

Implementierungsleitfaden

- Führen Sie Aufgaben konstant aus, sodass auch bei großen, schnellen Lastveränderungen keine Fehler auf Systemen auftreten.
- Implementieren Sie lose gekoppelte Abhängigkeiten. Abhängigkeiten etwa zwischen Warteschlangensystemen, Streaming-Systemen, Workflows und Load Balancern sind lose gekoppelt. Eine lose Verkoppelung hilft, das Verhalten einer Komponente von anderen Komponenten zu isolieren, die von ihr abhängig sind. Dies verbessert Resilienz und Agilität.
 - [Die Amazon Builders' Library: Zuverlässigkeit, stetige Ausführung und eine gute Tasse Kaffee](#)
 - [AWS re:Invent 2018: Regelkreise schließen und neue Denkansätze eröffnen: Wie man die Kontrolle über große und kleine Systeme übernimmt ARC337 \(beinhaltet ständige Arbeit\)](#)
 - Beispiel: Zustandsprüfungssystem, das 100.000 Server überwacht: Entwickeln Sie die Workloads so, dass die Nutzlastgrößen unabhängig von der Anzahl der Erfolge oder Ausfälle konstant bleiben.

Ressourcen

Zugehörige Dokumente:

- [AmazonEC2: Idempotenz sicherstellen](#)
- [Die Amazon Builders' Library: Herausforderungen für verteilte Systeme](#)
- [Die Amazon Builders' Library: Zuverlässigkeit, stetige Ausführung und eine gute Tasse Kaffee](#)

Zugehörige Videos:

- [AWS New York Summit 2019: Einführung in ereignisgesteuerte Architekturen und Amazon \(05\) EventBridge MAD2](#)
- [AWS re:Invent 2018: Close-Loops und offene Denkansätze: Wie man die Kontrolle über große und kleine Systeme übernimmt \(beinhaltet ständige Arbeit\) ARC337](#)

- [AWS re:Invent 2018: Close Loops und Opening Minds: So übernehmen Sie die Kontrolle über große und kleine Systeme ARC337 \(beinhaltet lockere Kopplung, konstante Arbeit, statische Stabilität\)](#)
- [AWS re:Invent 2019: Umstellung auf ereignisgesteuerte Architekturen \(08\) SVS3](#)

REL04-BP04 Festlegen von Mutationsoperationen als idempotent

Ein idempotenter Service garantiert, dass jede Anfrage genau einmal verarbeitet wird. Das bedeutet, dass das Senden mehrerer identischer Anfragen den gleichen Effekt hat wie das Senden einer einzelnen Anfrage. Dies vereinfacht es für einen Client, Wiederholungen zu implementieren. So muss nicht befürchtet werden, dass eine Anfrage fälschlicherweise mehrfach verarbeitet wird. Zu diesem Zweck können Clients API-Anfragen mit einem Idempotenz-Token ausgeben. Das gleiche Token wird verwendet, wenn die Anfrage wiederholt wird. Eine idempotente Service-API gibt mithilfe des Tokens eine Antwort zurück, die identisch mit der Antwort ist, die beim ersten Abschluss der Anfrage zurückgegeben wurde, selbst wenn sich der zugrunde liegende Zustand des Systems geändert hat.

In einem verteilten System ist es relativ einfach, eine Aktion höchstens einmal (der Client führt nur eine Anforderung aus) oder mindestens einmal (Anforderung wird ausgeführt, bis der Client eine Erfolgsbenachrichtigung erhält) durchzuführen. Es ist jedoch schwieriger, zu gewährleisten, dass eine Aktion genau einmal ausgeführt wird, so dass das Erstellen mehrerer identischer Anfragen den gleichen Effekt hat wie das Erstellen einer einzelnen Anfrage. Durch die Verwendung von idempotenten Tokens in APIs können Services einmal oder mehrmals eine mutierende Anforderung erhalten, ohne dass doppelte Datensätze erstellt werden müssen oder andere Probleme entstehen.

Gewünschtes Ergebnis: Sie verfügen über einen konsistenten, gut dokumentierten und häufig verwendeten Ansatz zur Sicherstellung der Idempotenz für alle Komponenten und Services.

Typische Anti-Muster:

- Sie wenden Idempotenz unterschiedslos an, auch wenn sie nicht benötigt wird.
- Sie führen eine zu komplexe Logik für die Implementierung von Idempotenz ein.
- Sie verwenden Zeitstempel als Schlüssel für Idempotenz. Dies kann zu Ungenauigkeiten führen, die auf eine Verschiebung der Uhrzeit oder darauf zurückzuführen sind, dass mehrere Clients dieselben Zeitstempel verwenden, um Änderungen vorzunehmen.
- Sie speichern vollständige Nutzdaten für Idempotenz. Bei diesem Ansatz speichern Sie vollständige Nutzdaten für jede Anfrage und überschreiben sie bei jeder neuen Anfrage. Dies kann Leistung und Skalierbarkeit beeinträchtigen.

- Sie generieren inkonsistente Service-Schlüssel. Ohne konsistente Schlüssel erkennen Services duplizierte Anfragen möglicherweise nicht, was zu unbeabsichtigten Ergebnissen führt.

Vorteile der Nutzung dieser bewährten Methode:

- **Höhere Skalierbarkeit:** Das System kann Wiederholungsversuche und doppelte Anfragen verarbeiten, ohne eine zusätzliche Logik oder eine komplexe Statusverwaltung anwenden zu müssen.
- **Verbesserte Zuverlässigkeit:** Mit Idempotenz können Services mehrere identische Anfragen konsistent bearbeiten, wodurch das Risiko unbeabsichtigter Nebenwirkungen oder doppelter Datensätze reduziert wird. Dies ist besonders wichtig in verteilten Systemen, in denen Netzwerkausfälle und Wiederholungsversuche häufig sind.
- **Verbesserte Datenkonsistenz:** Da dieselbe Anfrage zu derselben Antwort führt, trägt Idempotenz dazu bei, die Datenkonsistenz in verteilten Systemen aufrechtzuerhalten. Dies ist wichtig, um die Integrität von Transaktionen und Vorgängen aufrechtzuerhalten.
- **Fehlerbehandlung:** Idempotenz-Tokens vereinfachen die Fehlerbehandlung. Wenn ein Client aufgrund eines Problems keine Antwort erhält, kann er die Anfrage problemlos erneut mit demselben Idempotenz-Token senden.
- **Betriebliche Transparenz:** Idempotenz ermöglicht eine bessere Überwachung und Protokollierung. Services können Anfragen mit ihren Idempotenz-Tokens protokollieren, was das Nachverfolgen und Debuggen von Problemen erleichtert.
- **Vereinfachter API-Vertrag:** Dies kann den Vertrag zwischen den client- und serverseitigen Systemen vereinfachen und Sorgen hinsichtlich einer fehlerhaften Datenverarbeitung verringern.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

In einem verteilten System ist es einfach, eine Aktion höchstens einmal (der Client führt nur eine Anforderung aus) oder mindestens einmal (Anforderung wird so lange ausgeführt, bis der Erfolg bestätigt wird) durchzuführen. Es ist jedoch schwierig, die Aktion genau einmal ausführen zu lassen. Um dies zu erreichen, sollten Ihre Kunden für jede Anfrage ein Idempotenz-Token generieren und bereitstellen.

Durch die Verwendung von Idempotenz-Tokens kann ein Service zwischen neuen und wiederholten Anfragen unterscheiden. Wenn ein Service eine Anfrage mit einem Idempotenz-Token empfängt, prüft er, ob das Token bereits verwendet wurde. Wenn das Token bereits verwendet wurde, ruft der

Service die gespeicherte Antwort ab und gibt sie zurück. Wenn das Token neu ist, verarbeitet der Service die Anfrage, speichert die Antwort zusammen mit dem Token und gibt dann die Antwort zurück. Dieser Mechanismus macht alle Antworten idempotent, was die Zuverlässigkeit und Konsistenz des verteilten Systems erhöht.

Idempotenz ist auch ein wichtiges Verhalten ereignisgesteuerter Architekturen. Diese Architekturen werden in der Regel durch eine Nachrichtenwarteschlange wie Amazon SQS, Amazon MQ, Amazon Kinesis Streams oder Amazon Managed Streaming for Apache Kafka (MSK) unterstützt. Unter bestimmten Umständen kann eine Nachricht, die nur einmal veröffentlicht wurde, versehentlich mehr als einmal zugestellt werden. Wenn ein Herausgeber Idempotenz-Token generiert und in Nachrichten einfügt, wird damit angefordert, dass die Verarbeitung doppelt empfangener Nachrichten nicht zu einer wiederholten Aktion für dieselbe Nachricht führt. Verbraucher sollten jedes erhaltene Token nachverfolgen und Nachrichten ignorieren, die duplizierte Token enthalten.

Services und Verbraucher sollten das empfangene Idempotenz-Token auch an alle nachgelagerten Services weitergeben, die sie aufrufen. Jeder nachgelagerte Service in der Verarbeitungskette ist ebenfalls dafür verantwortlich, sicherzustellen, dass Idempotenz implementiert wird, um den Nebeneffekt einer mehrfachen Verarbeitung einer Nachricht zu vermeiden.

Implementierungsschritte

1. Identifizieren idempotenter Operationen

Ermitteln Sie, für welche Operationen Idempotenz erforderlich ist. Dazu gehören in der Regel die HTTP-Methoden POST, PUT und DELETE sowie Operationen zum Einfügen, Aktualisieren oder Löschen von Datenbanken. Operationen, bei denen der Status nicht verändert wird, z. B. schreibgeschützte Abfragen, erfordern normalerweise keine Idempotenz, es sei denn, es gibt Nebeneffekte.

2. Verwenden eindeutiger Bezeichner

Fügen Sie jeder idempotenten Operationsanforderung, die vom Absender gesendet wird, ein eindeutiges Token hinzu, entweder direkt in der Anfrage oder als Teil ihrer Metadaten (z. B. einen HTTP-Header). Auf diese Weise kann der Empfänger duplizierte Anforderungen oder Operationen erkennen und verarbeiten. Zu den üblicherweise für Token verwendeten Bezeichnern gehören [Universally Unique Identifiers \(UUIDs\)](#) und [K-Sortable Unique Identifiers \(KSUIDs\)](#).

3. Verfolgen und Verwalten des Status

Behalten Sie den Status jeder Operation oder Anforderung in Ihrem Workload bei. Dies kann erreicht werden, indem das Idempotenz-Token und der entsprechende Status (z. B. ausstehend,

abgeschlossen oder fehlgeschlagen) in einer Datenbank, einem Cache oder einem anderen persistenten Speicher gespeichert werden. Diese Statusinformationen ermöglichen dem Workload die Identifizierung und Verarbeitung duplizierter Anforderungen oder Operationen.

Sorgen Sie für Konsistenz und Atomizität, indem Sie bei Bedarf geeignete Mechanismen zur Kontrolle der Gleichzeitigkeit verwenden, z. B. Sperren, Transaktionen oder optimistische Gleichzeitigkeitskontrollen. Dazu gehören das Aufzeichnen des idempotenten Tokens und das Ausführen aller Mutationsoperationen, die mit der Bearbeitung der Anfrage verbunden sind. Auf diese Weise wird verhindert, dass es zu Race-Bedingungen kommt, und sichergestellt, dass idempotente Operationen korrekt ausgeführt werden.

Entfernen Sie regelmäßig alte Idempotenz-Token aus dem Datenspeicher, um Speicherplatz und Leistung zu verwalten. Wenn Ihr Speichersystem es unterstützt, sollten Sie die Verwendung von Ablaufzeitstempeln für Daten in Betracht ziehen (häufig auch als Time to Live- oder TTL-Werte bezeichnet). Die Wahrscheinlichkeit einer Wiederverwendung von Idempotenz-Token nimmt mit der Zeit ab.

Zu den gängigen AWS-Speicheroptionen, die typischerweise zum Speichern von Idempotenz-Token und des zugehörigen Status verwendet werden, gehören:

- **Amazon DynamoDB:** DynamoDB ist ein NoSQL-Datenbankservice, der eine Leistung mit niedriger Latenz und eine hohe Verfügbarkeit bietet und sich daher gut für die Speicherung von Daten im Zusammenhang mit Idempotenz eignet. Das Schlüssel-Wert- und Dokumentdatenmodell von DynamoDB ermöglicht das effiziente Speichern und Abrufen von Idempotenz-Token und der zugehörigen Statusinformationen. DynamoDB kann Idempotenz-Token auch automatisch ablaufen lassen, wenn Ihre Anwendung beim Einfügen einen TTL-Wert festlegt.
- **Amazon ElastiCache:** ElastiCache kann Idempotenz-Token mit hohem Durchsatz, geringer Latenz und niedrigen Kosten speichern. Sowohl ElastiCache (Redis) als auch ElastiCache (Memcached) können Idempotenz-Token auch automatisch ablaufen lassen, wenn Ihre Anwendung beim Einfügen einen TTL-Wert festlegt.
- **Amazon Relational Database Service (RDS):** Sie können Amazon RDS verwenden, um Idempotenz-Token und zugehörige Statusinformationen zu speichern, insbesondere, wenn Ihre Anwendung bereits eine relationale Datenbank für andere Zwecke verwendet.
- **Amazon Simple Storage Service (S3):** Amazon S3 ist ein hoch skalierbarer und langlebiger Objektspeicherservice, der zum Speichern von Idempotenz-Token und der zugehörigen Metadaten verwendet werden kann. Die Versionsverwaltungsfunktionen von S3 können besonders nützlich sein, um den Status idempotenter Operationen aufrechtzuerhalten.

Die Wahl des Speicherservices ist in der Regel von Faktoren wie der Menge der Daten im Zusammenhang mit Idempotenz, den erforderlichen Leistungsmerkmalen, der benötigten Speicherdauer und Verfügbarkeit sowie der Art und Weise abhängig, wie der Idempotenzmechanismus in die gesamte Workload-Architektur integriert ist.

4. Implementieren von idempotenten Operationen

Entwerfen Sie Ihre API- und Workload-Komponenten so, dass sie idempotent sind. Integrieren Sie Idempotenzprüfungen in Ihre Workload-Komponenten. Überprüfen Sie vor der Verarbeitung einer Anforderung oder Ausführung einer Operation, ob der eindeutige Bezeichner bereits verarbeitet wurde. Wenn dies der Fall ist, geben Sie das vorherige Ergebnis zurück, anstatt die Operation erneut auszuführen. Wenn ein Client beispielsweise eine Anforderung zur Erstellung eines Benutzers sendet, überprüfen Sie, ob bereits ein Benutzer mit demselben eindeutigen Bezeichner vorhanden ist. Wenn der Benutzer vorhanden ist, sollten die vorhandenen Benutzerinformationen zurückgegeben werden, statt einen neuen Benutzer zu erstellen. Ebenso sollte ein Verbraucher Nachrichten ignorieren, die duplizierte Idempotenz-Token enthalten.

Erstellen Sie umfassende Testsuiten zur Validierung der Idempotenz von Anforderungen. Diese sollten ein breites Spektrum von Szenarien abdecken, z. B. erfolgreiche, fehlgeschlagene und doppelte Anforderungen.

Wenn Ihr Workload AWS Lambda-Funktionen nutzt, sollten Sie Powertools für AWS Lambda in Betracht ziehen. Powertools für AWS Lambda ist ein Entwickler-Toolkit zur Implementierung bewährter Serverless-Methoden und zur Steigerung der Entwicklungsgeschwindigkeit bei der Arbeit mit AWS Lambda-Funktionen. Insbesondere enthält der Kit ein Hilfsprogramm, mit dem Sie Ihre Lambda-Funktionen in idempotente Operationen konvertieren können, die sicher wiederholt werden können.

5. Klare Kommunikation der Idempotenz

Dokumentieren Sie Ihre API- und Workload-Komponenten, um die idempotente Natur der Operationen deutlich zu machen. Dies hilft Clients, das erwartete Verhalten zu verstehen und zu wissen, wie sie zuverlässig mit Ihrem Workload interagieren können.

6. Überwachen und Prüfen

Implementieren Sie Überwachungs- und Prüfmechanismen, um Probleme im Zusammenhang mit der Idempotenz von Antworten aufzudecken, z. B. unerwartete Antwortvarianten oder eine übermäßige Verarbeitung duplizierter Anforderungen. Dies kann Ihnen helfen, Probleme oder unerwartete Verhaltensweisen in Ihren Workloads zu erkennen und zu untersuchen.

Ressourcen

Zugehörige bewährte Methoden:

- [REL05-BP03 Steuern und Einschränken von Wiederholungsaufrufen](#)
- [REL06-BP01 Überwachen aller Komponenten des Workloads \(Generierung\)](#)
- [REL06-BP03 Senden von Benachrichtigungen \(Verarbeitung und Benachrichtigung in Echtzeit\)](#)
- [REL08-BP02 Integrieren von Funktionstests in die Bereitstellung](#)

Zugehörige Dokumente:

- [Amazon Builders' Library: Sichere Wiederholungen mit idempotenten APIs](#)
- [Die Amazon Builders' Library: Herausforderungen für verteilte Systeme](#)
- [Die Amazon Builders' Library: Zuverlässigkeit, stetige Ausführung und eine gute Tasse Kaffee](#)
- [Amazon Elastic Container Service: Sicherstellen der Idempotenz](#)
- [Wie mache ich meine Lambda-Funktion idempotent?](#)
- [Sicherstellen der Idempotenz bei Amazon-EC2-API-Anforderungen](#)

Zugehörige Videos:

- [Entwickeln verteilter Anwendungen mit ereignisgesteuerter Architektur – AWS Online Tech Talks](#)
- [AWS re:Invent 2023 – Building next-generation applications with event-driven architecture](#)
- [AWS re:Invent 2023 – Advanced integration patterns & trade-offs for loosely coupled systems](#)
- [AWS re:Invent 2023 – Advanced event-driven patterns with Amazon EventBridge](#)
- [AWS re:Invent 2018 – Close Loops and Opening Minds: How to Take Control of Systems, Big and Small ARC337 \(includes loose coupling, constant work, static stability\)](#)
- [AWS re:Invent 2019 – Moving to event-driven architectures \(SVS308\)](#)

Zugehörige Tools:

- [Idempotenz mit AWS Lambda Powertools \(Java\)](#)
- [Idempotenz mit AWS Lambda Powertools \(Python\)](#)
- [AWS Lambda Powertools GitHub-Seite](#)

REL 5. Wie lassen sich Interaktionen in einem verteilten System so gestalten, dass Ausfälle abgemildert oder bewältigt werden?

Verteilte Systeme nutzen Kommunikationsnetzwerke, um Komponenten (wie Server oder Services) miteinander zu verbinden. Ihre Workload muss trotz Datenverlust oder höherer Latenz in diesen Netzwerken zuverlässig ausgeführt werden. Die Komponenten des verteilten Systems müssen so funktionieren, dass sie sich nicht negativ auf andere Komponenten oder die Workload auswirken. Diese bewährten Methoden sorgen dafür, dass Workloads Belastungen oder Fehlern standhalten, sich schneller davon erholen und die Auswirkungen solcher Beeinträchtigungen abgeschwächt werden. Das Ergebnis ist eine verbesserte mittlere Reparaturzeit (MTTR).

Best Practices

- [REL05-BP01 Implementieren einer ordnungsgemäßen Funktionsminderung, um harte Abhängigkeiten in weiche zu ändern](#)
- [REL05-BP02 Drosselung von Anfragen](#)
- [REL05-BP03 Steuerung und Begrenzung von Wiederholungsaufrufen](#)
- [REL05-BP04 Schnelles Scheitern und Begrenzen von Warteschlangen](#)
- [REL05-BP05 Client-Timeouts festlegen](#)
- [REL05-BP06 Systeme soweit möglich zustandslos machen](#)
- [REL05-BP07 Nothebel einbauen](#)

REL05-BP01 Implementieren einer ordnungsgemäßen Funktionsminderung, um harte Abhängigkeiten in weiche zu ändern

Anwendungskomponenten sollten weiterhin ihre Kernfunktion erfüllen, auch wenn Abhängigkeiten nicht mehr verfügbar sind. Sie liefern möglicherweise leicht veraltete Daten, alternative Daten oder sogar keine Daten. Dadurch wird sichergestellt, dass die Gesamtsystemfunktion nur minimal durch lokale Ausfälle beeinträchtigt wird, während gleichzeitig der zentrale Geschäftswert gewährleistet ist.

Gewünschtes Ergebnis: Wenn die Abhängigkeiten einer Komponente fehlerhaft sind, kann die Komponente selbst weiterhin funktionieren, wenn auch in eingeschränkter Weise. Komponentenausfälle sollten als normaler Geschäftsbetrieb betrachtet werden. Arbeitsabläufe sollten so konzipiert sein, dass solche Ausfälle nicht zu einem vollständigen Ausfall oder zumindest zu vorhersehbaren und wiederherstellbaren Zuständen führen.

Typische Anti-Muster:

- Die erforderlichen Kerngeschäftsfunktionen wurden nicht identifiziert. Es wird nicht getestet, ob die Komponenten auch bei Abhängigkeitsfehlern funktionsfähig sind.
- Es werden keine Daten zu Fehlern bereitgestellt oder wenn nur eine von mehreren Abhängigkeiten nicht verfügbar ist und Teilergebnisse dennoch zurückgegeben werden können.
- Es entsteht ein inkonsistenter Zustand, wenn eine Transaktion teilweise fehlschlägt.
- Es gibt keine alternative Möglichkeit, auf einen zentralen Parameterspeicher zuzugreifen.
- Lokale Zustände werden aufgrund einer fehlgeschlagenen Aktualisierung ungültig oder geleert, ohne die Konsequenzen zu berücksichtigen.

Vorteile der Nutzung dieser bewährten Methode: Eine schrittweise Degradation verbessert die Verfügbarkeit des gesamten Systems und gewährleistet die Funktionsfähigkeit der wichtigsten Funktionen auch bei Ausfällen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Die Implementierung einer schrittweisen Degradation trägt dazu bei, die Auswirkungen von Abhängigkeitsfehlern auf die Komponentenfunktion zu minimieren. Im Idealfall erkennt eine Komponente Abhängigkeitsfehler und umgeht sie so, dass sich dies nur minimal auf andere Komponenten oder Kunden auswirkt.

Eine Architektur, die auf eine schrittweise Degradation ausgerichtet ist, bedeutet, potenzielle Ausfallmodi beim Entwurf von Abhängigkeiten zu berücksichtigen. Sorgen Sie für jeden Ausfallmodus für eine Möglichkeit, aufrufenden Komponenten oder Kunden die meisten oder zumindest die wichtigsten Funktionen der Komponente bereitzustellen. Diese Überlegungen können zu zusätzlichen Anforderungen werden, die getestet und verifiziert werden können. Im Idealfall ist eine Komponente in der Lage, ihre Kernfunktion auf akzeptable Weise auszuführen, selbst wenn eine oder mehrere Abhängigkeiten ausfallen.

Dies ist sowohl eine geschäftliche als auch eine technische Diskussion. Alle Geschäftsanforderungen sind wichtig und sollten nach Möglichkeit erfüllt werden. Es ist jedoch immer noch sinnvoll, sich zu fragen, was passieren soll, wenn nicht alle erfüllt werden können. Ein System kann so konzipiert werden, dass es verfügbar und konsistent ist. Doch was davon ist wichtiger, wenn auf eines davon verzichtet werden muss? Bei der Zahlungsabwicklung könnte dies die Konsistenz sein. Bei einer Echtzeitanwendung ist es eher die Verfügbarkeit. Bei einer kundenseitigen Website kann die Antwort von den Kundenerwartungen abhängen.

Was das bedeutet, hängt von den Anforderungen der Komponente ab und davon, was als ihre Kernfunktion angesehen werden sollte. Beispiel:

- Eine E-Commerce-Website kann Daten aus verschiedenen Systemen wie personalisierte Empfehlungen, bestbewertete Produkte und den Status von Kundenbestellungen auf der Startseite anzeigen. Wenn ein Upstream-System ausfällt, ist es immer noch sinnvoll, alles andere anzuzeigen, anstatt einem Kunden eine Fehlerseite anzuzeigen.
- Eine Komponente, die Batch-Schreibvorgänge durchführt, kann einen Stapel trotzdem weiterverarbeiten, wenn eine der einzelnen Operationen fehlschlägt. Es sollte einfach sein, einen Wiederholungsmechanismus zu implementieren. Geben Sie dazu Informationen dazu zurück, welche Operationen erfolgreich, welche fehlgeschlagen und warum sie fehlgeschlagen sind. Oder stellen Sie fehlgeschlagene Anfragen in eine Warteschlange für unzustellbare Nachrichten, um asynchrone Wiederholungsversuche zu implementieren. Informationen über fehlgeschlagene Operationen sollten ebenfalls protokolliert werden.
- Ein System, das Transaktionen verarbeitet, muss überprüfen, ob entweder alle oder keine einzelnen Aktualisierungen ausgeführt werden. Bei verteilten Transaktionen kann das Saga-Muster verwendet werden, um vorherige Operationen rückgängig zu machen, falls ein späterer Vorgang derselben Transaktion fehlschlägt. Hier besteht die Kernfunktion darin, die Konsistenz aufrechtzuerhalten.
- Zeitkritische Systeme sollten in der Lage sein, mit Abhängigkeiten umzugehen, die nicht rechtzeitig reagieren. In diesen Fällen kann das Unterbrechermuster verwendet werden. Wenn bei Antworten aus einer Abhängigkeit eine Zeitüberschreitung auftritt, kann das System in einen geschlossenen Zustand wechseln, in dem keine weiteren Aufrufe getätigt werden.
- Eine Anwendung kann Parameter aus einem Parameterspeicher lesen. Es kann nützlich sein, Container-Images mit einem Satz von Standardparametern zu erstellen und diese zu verwenden, falls der Parameterspeicher nicht verfügbar ist.

Beachten Sie, dass die im Falle eines Komponentenausfalls eingeschlagenen Pfade getestet werden müssen und deutlich einfacher sein sollten als der primäre Pfad. Allgemein [sollten Fallback-Strategien vermieden werden](#).

Implementierungsschritte

Identifizieren Sie externe und interne Abhängigkeiten. Überlegen Sie, welche Arten von Fehlern bei ihnen auftreten können. Überlegen Sie, wie Sie die negativen Auswirkungen dieser Ausfälle auf vor- und nachgeschaltete Systeme und Kunden minimieren können.

Im Folgenden finden Sie eine Liste von Abhängigkeiten und wie Sie sie schrittweise degradieren können, wenn sie ausfallen:

1. Teilweiser Ausfall von Abhängigkeiten: Eine Komponente kann mehrere Anfragen an nachgelagerte Systeme stellen, entweder in Form mehrerer Anfragen an ein System oder in Form einer Anfrage an jeweils mehrere Systeme. Je nach Unternehmenskontext können unterschiedliche Vorgehensweisen angemessen sein (weitere Einzelheiten finden Sie in den vorherigen Beispielen in den Implementierungsleitfäden).
2. Ein nachgelagertes System kann Anfragen aufgrund der hohen Auslastung nicht verarbeiten: Wenn Anfragen an ein nachgelagertes System immer wieder fehlschlagen, ist es nicht sinnvoll, es erneut zu versuchen. Dies kann ein bereits überlastetes System zusätzlich belasten und die Wiederherstellung erschweren. Hier kann das Unterbrechermuster verwendet werden, das fehlgeschlagene Aufrufe an ein nachgelagertes System überwacht. Wenn eine große Anzahl von Aufrufen fehlschlägt, werden keine weiteren Anfragen mehr an das nachgelagerte System gesendet und nur gelegentlich Aufrufe durchgelassen, um zu testen, ob das nachgelagerte System wieder verfügbar ist.
3. Ein Parameterspeicher ist nicht verfügbar: Um einen Parameterspeicher umzuwandeln, können Soft Dependency Caching oder vernünftige Standardwerte verwendet werden, die in Container-Images oder Machine Images enthalten sind. Beachten Sie, dass diese Standardwerte auf dem neuesten Stand gehalten und in die Testsuiten aufgenommen werden müssen.
4. Ein Überwachungsservice oder eine andere nicht funktionale Abhängigkeit ist nicht verfügbar: Wenn eine Komponente zeitweise nicht in der Lage ist, Protokolle, Metriken oder Spuren an einen zentralen Überwachungsservice zu senden, ist es oft am besten, Geschäftsfunktionen weiterhin wie gewohnt auszuführen. Es ist oft nicht akzeptabel, Metriken über einen längeren Zeitraum stillschweigend nicht zu protokollieren oder weiterzuleiten. In einigen Anwendungsfällen können auch vollständige Auditeinträge erforderlich sein, um die Compliance-Anforderungen zu erfüllen.
5. Eine primäre Instance einer relationalen Datenbank ist möglicherweise nicht verfügbar: Amazon Relational Database Service kann, wie fast alle relationalen Datenbanken, nur eine primäre Writer-Instance haben. Dies führt zu einem einzigen Fehlerpunkt für Schreib-Workloads und erschwert die Skalierung. Dies kann teilweise gemildert werden, indem eine Multi-AZ-Konfiguration für hohe Verfügbarkeit oder Amazon Aurora Serverless für eine bessere Skalierung verwendet wird. Bei sehr hohen Verfügbarkeitsanforderungen kann es sinnvoll sein, sich überhaupt nicht auf den primären Writer zu verlassen. Für Abfragen, die nur lesen, können Lesereplikate verwendet werden, die Redundanz und die Möglichkeit bieten, nicht nur hoch-, sondern auch aufzuskalieren. Schreibvorgänge können gepuffert werden, zum Beispiel in einer Amazon Simple Queue Service-

Warteschlange, sodass Schreibenfragen von Kunden auch dann akzeptiert werden können, wenn das primäre Gerät vorübergehend nicht verfügbar ist.

Ressourcen

Zugehörige Dokumente:

- [Amazon API Gateway: Throttle API Requests for Better Throughput](#)
- [CircuitBreaker](#) (Zusammenfassung des Unterbrechers aus dem Buch „Release It!“)
- [Error Retries and Exponential Backoff in AWS](#)
- [Michael Nygard “Release It! Design and Deploy Production-Ready Software”](#)
- [Die Amazon Builders' Library: Vermeiden von Fallback in verteilten Systemen](#)
- [Die Amazon Builders' Library: Vermeiden von nicht mehr aufholbaren Warteschlangen-Rückständen](#)
- [Die Amazon Builders' Library: Herausforderungen und Strategien für das Caching](#)
- [Die Amazon Builders' Library: Timeouts, Wiederholungen und Backoff mit Jitter](#)

Zugehörige Videos:

- [Retry, backoff, and jitter: AWS re:Invent 2019: Introducing The Amazon Builders' Library \(DOP328\)](#)

REL05-BP02 Drosselung von Anfragen

Drosseln Sie Anfragen, um eine Ressourcenüberlastung aufgrund eines unerwarteten Nachfrageanstiegs zu verringern. Anfragen, die unter der Drosselungsrate liegen, werden verarbeitet, während Anfragen, die über dem definierten Limit liegen, abgelehnt werden. Es wird eine Meldung zurückgegeben, die besagt, dass die Anfrage gedrosselt wurde.

Gewünschtes Ergebnis: Stark ansteigendes Volumen, das entweder durch plötzliche Anstiege des Kundendatenverkehrs, Flooding-Angriffe oder Wiederholungsstürme verursacht wird, wird durch Anfragedrosselung abgeschwächt, sodass Workloads die normale Verarbeitung des unterstützten Anforderungsvolumens fortsetzen können.

Typische Anti-Muster:

- API-Endpunktdrosselungen sind nicht implementiert oder werden auf Standardwerten belassen, ohne die erwarteten Volumina zu berücksichtigen.

- API-Endpunkte werden nicht ausgelastet oder die Drosselungsgrenzwerte werden nicht getestet.
- Anforderungsraten werden ohne Berücksichtigung der Größe oder Komplexität der Anfrage gedrosselt.
- Es werden sowohl die maximalen Anforderungsraten als auch die maximale Anforderungsgröße getestet, aber nicht beides zusammen.
- Ressourcen werden nicht mit denselben Limits bereitgestellt, die beim Testen festgelegt wurden.
- Es wurden keine Nutzungspläne konfiguriert oder für A2A-API-Verbraucher in Betracht gezogen.
- Für Warteschlangenverbraucher, die horizontal skalieren, sind keine Einstellungen für maximale Parallelität konfiguriert.
- Eine Ratenbegrenzung pro IP-Adresse wurde nicht implementiert.

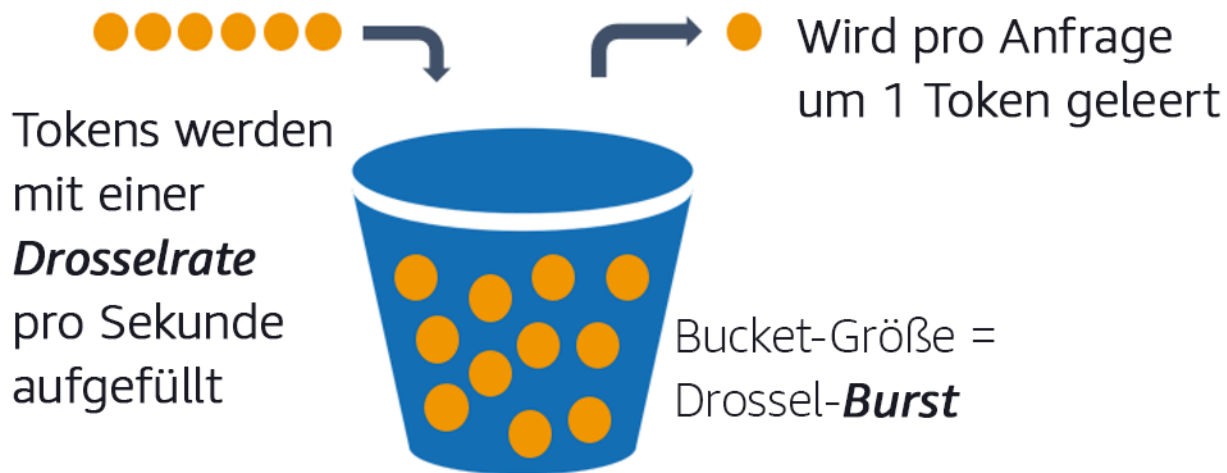
Vorteile der Nutzung dieser bewährten Methode: Workloads, die Drosselgrenzwerte festlegen, können normal arbeiten und akzeptierte Anfragen auch bei unerwarteten Volumenspitzen erfolgreich verarbeiten. Plötzliche oder anhaltende Spitzen von Anfragen an APIs und Warteschlangen werden gedrosselt und verbrauchen keine Ressourcen für die Anforderungsverarbeitung. Ratenbegrenzungen drosseln einzelne Anforderer, sodass ein hohes Datenverkehrsvolumen von einer einzelnen IP-Adresse oder einem API-Verbraucher keine Ressourcen verbraucht, die sich auf andere Verbraucher auswirken.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Services sollten so konzipiert sein, dass sie eine bekannte Kapazität von Anfragen verarbeiten. Diese Kapazität kann durch Auslastungstests ermittelt werden. Wenn die Anzahl der Anfragen die Grenzwerte überschreitet, signalisiert die entsprechende Antwort, dass eine Anfrage gedrosselt wurde. Dies ermöglicht es dem Verbraucher, den Fehler zu beheben und es später erneut zu versuchen.

Wenn für Ihren Service eine Drosselungsimplementierung erforderlich ist, sollten Sie die Implementierung des Token-Bucket-Algorithmus in Betracht ziehen, bei dem ein Token für eine Anfrage zählt. Tokens werden mit einer Drosselrate pro Sekunde aufgefüllt und asynchron um ein Token pro Anfrage geleert.



Der Token-Bucket-Algorithmus

[Amazon API Gateway](#) implementiert den Token-Bucket-Algorithmus entsprechend den Konto- und Regionslimits und kann pro Client mit Nutzungsplänen konfiguriert werden. Darüber hinaus können [Amazon Simple Queue Service \(Amazon SQS\)](#) und [Amazon Kinesis](#) Anfragen zwischenspeichern, um die Anforderungsrate auszugleichen, und höhere Drosselungsraten für Anfragen ermöglichen, die bearbeitet werden können. Schließlich können Sie die Ratenbegrenzung mit [AWS WAF](#) implementieren, um bestimmte API-Verbraucher zu drosseln, die ungewöhnlich hohe Lasten erzeugen.

Implementierungsschritte

Sie können API Gateway mit Drosselungslimits für Ihre APIs konfigurieren und 429 Too Many Requests-Fehler zurückgeben, wenn Grenzwerte überschritten werden. Sie können AWS WAF zusammen mit Ihren AWS AppSync- und API Gateway-Endpunkten verwenden, um die Ratenbegrenzung pro IP-Adresse zu aktivieren. Wenn Ihr System asynchrone Verarbeitung toleriert, können Sie außerdem Nachrichten in eine Warteschlange oder einen Stream stellen, um die Antworten an Service-Clients zu beschleunigen und so höhere Drosselungsraten zu erreichen.

Wenn Sie Amazon SQS als Ereignisquelle für AWS Lambda konfiguriert haben, können Sie mit asynchroner Verarbeitung [maximale Gleichzeitigkeit konfigurieren](#), um zu verhindern, dass hohe Ereignisraten die für andere Services in Ihrem Workload oder Konto benötigten Kontingente für gleichzeitige Ausführungen auf Kontoebene verbrauchen.

API Gateway bietet zwar eine verwaltete Implementierung des Token-Buckets, aber in Fällen, in denen Sie API Gateway nicht verwenden können, können Sie sprachspezifische Open-Source-

Implementierungen (siehe entsprechende Beispiele unter Ressourcen) des Token-Buckets für Ihre Services nutzen.

- Verstehen und konfigurieren Sie [API Gateway-Drosselungslimits](#) auf Kontoebene pro Region, API pro Phase und API-Schlüssel pro Nutzungsplanebene.
- Wenden Sie die [AWS WAF-Regeln zur Ratenbegrenzung](#) auf API Gateway- und AWS AppSync-Endpunkte an, um sich vor Flooding zu schützen und schädliche IPs zu sperren. Regeln zur Ratenbegrenzung können auch für AWS AppSync-API-Schlüssel für A2A-Verbraucher konfiguriert werden.
- Überlegen Sie, ob Sie für AWS AppSync-APIs mehr Drosselungskontrolle als Ratenbegrenzung benötigen, und konfigurieren Sie in diesem Fall ein API Gateway vor Ihrem AWS AppSync-Endpunkt.
- Wenn Amazon SQS-Warteschlangen als Auslöser für Lambda-Warteschlangenverbraucher eingerichtet werden, legen Sie die [maximale Gleichzeitigkeit](#) auf einen Wert fest, mit dem genug verarbeitet wird, um Ihre Service-Level-Ziele zu erreichen, aber keine Gleichzeitigkeitsbeschränkungen ausnutzt werden, die sich auf andere Lambda-Funktionen auswirken. Erwägen Sie, die reservierte Gleichzeitigkeit für andere Lambda-Funktionen in demselben Konto und derselben Region festzulegen, wenn Sie Warteschlangen mit Lambda verbrauchen.
- Verwenden Sie API Gateway mit nativen Serviceintegrationen in Amazon SQS oder Kinesis, um Anfragen zwischenzuspeichern.
- Wenn Sie API Gateway nicht verwenden können, nutzen Sie sprachspezifische Bibliotheken, um den Token-Bucket-Algorithmus für Ihren Workload zu implementieren. Sehen Sie sich den Abschnitt mit den Beispielen an und recherchieren Sie selbst, um eine geeignete Bibliothek zu finden.
- Testen Sie Grenzwerte, die Sie festlegen oder deren Erhöhung Sie zulassen möchten, und dokumentieren Sie die getesteten Grenzwerte.
- Erhöhen Sie die Grenzwerte nicht über das hinaus, was Sie beim Testen festgelegt haben. Wenn Sie einen Grenzwert erhöhen, stellen Sie sicher, dass die bereitgestellten Ressourcen bereits denen in Testszenarien entsprechen oder diese übertreffen, bevor Sie die Erhöhung anwenden.

Ressourcen

Zugehörige bewährte Methoden:

- [REL04-BP03 Arbeite ständig](#)

- [REL05-BP03 Steuerung und Begrenzung von Wiederholungsaufrufen](#)

Zugehörige Dokumente:

- [Amazon API Gateway: Throttle API Requests for Better Throughput](#)
- [AWS WAF: Rate-based rule statement](#)
- [Introducing maximum concurrency of AWS Lambda when using Amazon SQS as an event source](#)
- [AWS Lambda: Maximum Concurrency](#)

Zugehörige Beispiele:

- [The three most important AWS WAF rate-based rules](#)
- [Java Bucket4j](#)
- [Python-Token-Bucket](#)
- [Node-Token-Bucket](#)
- [.NET System Threading Rate Limiting](#)

Zugehörige Videos:

- [Implementing GraphQL API security best practices with AWS AppSync](#)

Zugehörige Tools:

- [Amazon API Gateway](#)
- [AWS AppSync](#)
- [Amazon SQS](#)
- [Amazon Kinesis](#)
- [AWS WAF](#)
- [Virtueller Warteraum auf AWS](#)

REL05-BP03 Steuerung und Begrenzung von Wiederholungsaufrufen

Verwenden Sie das exponentielle Backoff, um Anfragen in zunehmend längeren Intervallen zwischen den einzelnen Wiederholungsversuchen zu wiederholen. Führen Sie Jitter zwischen den

Wiederholungen ein, um die Wiederholungsintervalle zufällig zu bestimmen. Beschränken Sie die maximale Anzahl an Wiederholungen.

Gewünschtes Ergebnis: Zu den typischen Komponenten eines verteilten Softwaresystems gehören Server, Load Balancer, Datenbanken und Server. DNS Während des normalen Betriebs können diese Komponenten auf Anfragen mit temporären oder begrenzten Fehlern sowie mit Fehlern antworten, die unabhängig von Wiederholungsversuchen dauerhaft bleiben würden. Wenn Clients Anfragen an Services stellen, verbrauchen die Anfragen Ressourcen wie Speicher, Threads, Verbindungen, Ports oder andere begrenzte Ressourcen. Die Steuerung und Einschränkung von Wiederholungsversuchen ist eine Strategie zur Freigabe und Minimierung des Ressourcenverbrauchs, sodass beanspruchte Systemkomponenten nicht überlastet werden.

Wenn Client-Anfragen eine Zeitüberschreitung oder Fehlerantworten erhalten, sollten sie entscheiden, ob sie es erneut versuchen möchten oder nicht. Wenn sie es erneut versuchen, tun sie dies mit exponentiellem Backoff mit Jitter und einem maximalen Wiederholungswert. Dadurch werden Backend-Services und -Prozesse entlastet und erhalten Zeit, um sich selbst zu reparieren, was zu einer schnelleren Wiederherstellung und einer erfolgreichen Bearbeitung von Anfragen führt.

Typische Anti-Muster:

- Wiederholungsversuche werden ohne exponentielles Backoff, Jitter und maximale Wiederholungswerte implementiert. Backoff und Jitter helfen dabei, künstliche Datenverkehrsspitzen zu vermeiden, die durch ungewollt koordinierte Wiederholungsversuche in regelmäßigen Intervallen entstehen.
- Implementierung von Wiederholungsversuchen, ohne deren Auswirkungen zu testen oder davon auszugehen, dass Wiederholungsversuche bereits in Szenarien integriert sind, und SDK ohne Wiederholungsversuche zu testen.
- Veröffentlichte Fehlercodes aus Abhängigkeiten werden nicht richtig interpretiert, was dazu führt, dass bei allen Fehlern eine Wiederholung versucht wird, auch dann, wenn die Ursache auf eine fehlende Berechtigung, einen Konfigurationsfehler oder ein anderes Problem hindeutet, das vorhersehbar nicht ohne manuelles Eingreifen behoben werden kann.
- Beobachtbarkeits-Praktiken, einschließlich der Überwachung und Meldung von Warnmeldungen bei wiederholten Serviceausfällen, damit die zugrunde liegenden Probleme bekannt werden und behoben werden können, werden nicht beachtet.
- Es werden benutzerdefinierte Wiederholungsmechanismen entwickelt, wenn integrierte Wiederholungsfunktionen oder Wiederholungsfunktionen von Drittanbietern ausreichen.

- Es werden Wiederholungsversuche auf mehreren Ebenen eines Anwendungsstapels auf eine Weise ausgeführt, die Wiederholungsversuche verstärkt, was die Ressourcen durch einen Wiederholungssturm weiter verbraucht. Vergewissern Sie sich, dass Sie verstehen, wie sich diese Fehler auf Ihre Anwendung und die Abhängigkeiten auswirken, auf die Sie sich verlassen, und führen Sie dann Wiederholungsversuche nur auf einer Ebene durch.
- Nicht idempotente Serviceaufrufe werden erneut versucht, was zu unerwarteten Nebeneffekten wie doppelten Ergebnissen führt.

Vorteile der Nutzung dieser bewährten Methode: Wiederholungsversuche helfen Clients dabei, die gewünschten Ergebnisse zu erzielen, wenn Anfragen fehlschlagen, verbrauchen aber auch mehr Zeit auf dem Server, um die gewünschten erfolgreichen Antworten zu erhalten. Wenn Fehler selten oder vorübergehend auftreten, funktionieren Wiederholungsversuche gut. Wenn Fehler durch Ressourcenüberlastung verursacht werden, können Wiederholungsversuche die Situation verschlimmern. Durch das Hinzufügen eines exponentiellen Backoffs mit Jitter zu den Client-Wiederholungsversuchen können Server sich erholen, wenn Ausfälle durch Ressourcenüberlastung verursacht werden. Jitter verhindert, dass Anfragen zu Datenverkehrsspitzen führen, und Backoff verringert die Lasteskalation, die durch das Hinzufügen von Wiederholungsversuchen zur normalen Anforderungslast verursacht wird. Schließlich ist es wichtig, eine maximale Anzahl von Wiederholungsversuchen oder die verstrichene Zeit zu konfigurieren, um zu vermeiden, dass Rückstände entstehen, die zu metastabilen Ausfällen führen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Steuern und begrenzen Sie Wiederholungsaufrufe. Verwenden Sie ein exponentielles Backoff, um Aufrufe nach zunehmend längeren Intervallen zu wiederholen. Nutzen Sie Jitter, um die Wiederholungsintervalle zu randomisieren, und legen Sie ein Limit für die Zahl der Wiederholungen fest.

Einige AWS SDKs implementieren standardmäßig Wiederholungsversuche und exponentielles Backoff. Verwenden Sie diese integrierten AWS Implementierungen, sofern dies für Ihren Workload relevant ist. Implementieren Sie eine ähnliche Logik in Ihrem Workload, wenn Sie Services aufrufen, die idempotent sind und bei denen Wiederholungsversuche die Verfügbarkeit Ihrer Clients verbessern. Legen Sie entsprechend Ihrem Anwendungsfall Zeitüberschreitungen fest und geben Sie an, wann Wiederholversuche gestoppt werden sollen. Erstellen Sie Testszenarien für diese Wiederholungsfälle und führen Sie sie aus.

Implementierungsschritte

- Ermitteln Sie die optimale Ebene in Ihrem Anwendungsstack, um Wiederholungsversuche für die Services zu implementieren, auf die sich Ihre Anwendung stützt.
- Seien Sie sich bewusst SDKs, dass es für die Sprache Ihrer Wahl bewährte Wiederholungsstrategien mit exponentiellem Backoff und Jitter gibt, und ziehen Sie diese dem Schreiben Ihrer eigenen Wiederholungsimplementierungen vor.
- Stellen Sie sicher, dass [Services idempotent sind](#), bevor Sie Wiederholungen implementieren. Sobald Wiederholungsversuche implementiert wurden, stellen Sie sicher, dass sie sowohl getestet als auch regelmäßig in der Produktion ausgeführt werden.
- Verwenden Sie beim Aufrufen des AWS Dienstes die Option und APIs machen Sie sich mit den Konfigurationsoptionen für [AWS SDKs](#) Wiederholungen vertraut [AWS CLI](#). Finden Sie heraus, ob die Standardeinstellungen für Ihren Anwendungsfall geeignet sind, testen Sie sie und passen Sie sie nach Bedarf an.

Ressourcen

Zugehörige bewährte Methoden:

- [REL04-BP04 Festlegen von Mutationsoperationen als idempotent](#)
- [REL05-BP02 Drosselung von Anfragen](#)
- [REL05-BP04 Schnelles Scheitern und Begrenzen von Warteschlangen](#)
- [REL05-BP05 Client-Timeouts festlegen](#)
- [REL11-BP01 Überwachen aller Komponenten der Workload auf Fehler](#)

Zugehörige Dokumente:

- [Fehler bei Wiederholungen und exponentiellem Backoff in AWS](#)
- [Die Amazon Builders' Library: Timeouts, Wiederholungen und Backoff mit Jitter](#)
- [Exponential Backoff and Jitter](#)
- [Machen Sie Wiederholungen mit idempotent sicher APIs](#)

Zugehörige Beispiele:

- [Spring Retry](#)

- [Resilience4j Retry](#)

Zugehörige Videos:

- [Wiederholung, Backoff und Jitter: AWS re:Invent 2019: Einführung in die Amazon Builders' Library \(\) DOP328](#)

Zugehörige Tools:

- [AWS SDKs und Tools: Verhalten bei Wiederholungen](#)
- [AWS Command Line Interface: Wiederholungen AWS CLI](#)

REL05-BP04 Schnelles Scheitern und Begrenzen von Warteschlangen

Wenn ein Service nicht in der Lage ist, erfolgreich auf eine Anfrage zu antworten, sollte er schnell scheitern. Dies ermöglicht die Freigabe von mit einer Anfrage verbundenen Ressourcen und damit die Wiederherstellung eines Services, falls dieser nicht mehr über genügend Ressourcen verfügt. Schnelles Scheitern ist ein etabliertes Softwaredesignmuster, das genutzt werden kann, um hochzuverlässige Workloads in der Cloud aufzubauen. Warteschlangen sind ebenfalls ein etabliertes Integrationsmuster für Unternehmen. Sie sorgen für eine ausgeglichene Auslastung und ermöglichen es den Clients, Ressourcen freizugeben, wenn eine asynchrone Verarbeitung toleriert wird. Wenn ein Service unter normalen Bedingungen erfolgreich antworten kann, aber fehlschlägt, wenn die Anforderungsrate zu hoch ist, verwenden Sie eine Warteschlange, um Anfragen zwischenzuspeichern. Lassen Sie jedoch keine langen Warteschlangen zu. Sie können dazu führen, dass veraltete Anfragen verarbeitet werden, die ein Client bereits aufgegeben hat.

Gewünschtes Ergebnis: Wenn bei Systemen Ressourcenknappheit, Timeouts, Ausnahmen oder Grauausfälle auftreten, die Service-Level-Ziele unerreichbar machen, ermöglichen Strategien für schnelles scheitern eine schnellere Systemwiederherstellung. Systeme, die Traffic-Spitzen absorbieren müssen und asynchrone Verarbeitung ermöglichen, können die Zuverlässigkeit verbessern, indem sie es Clients ermöglichen, Anfragen schnell freizugeben, indem sie Warteschlangen verwenden, um Anfragen an Back-End-Services zu puffern. Beim Puffern von Anfragen in Warteschlangen werden Strategien zur Warteschlangenverwaltung implementiert, um nicht mehr aufzuholende Rückstände zu vermeiden.

Typische Anti-Muster:

- Implementierung von Nachrichtenwarteschlangen, aber keine Konfiguration von Warteschlangen für unzustellbare Nachrichten (DLQ) oder Alarmen für volle DLQs, um zu erkennen, wenn ein System ausfällt.
- Nichterfassung des Alters von Nachrichten in einer Warteschlange, einem Indikator für Latenz, um zu verstehen, wann Warteschlangenverbraucher mit der Verarbeitung nicht mehr hinterher kommen oder Fehler machen, was zu erneuten Versuchen führt.
- Kein Löschen von aufgestauten Nachrichten aus einer Warteschlange, wenn es keinen Sinn macht, diese Nachrichten zu verarbeiten, da kein Geschäftsbedarf mehr besteht.
- Die Konfiguration von First-in-First-Out (FIFO)-Warteschlangen, wenn Last-In-First-Out (LIFO)-Warteschlangen den Client-Anforderungen besser gerecht werden würden. Dies ist beispielsweise dann der Fall, wenn keine strenge Reihenfolge erforderlich ist und die Backlog-Verarbeitung alle neuen und zeitkritischen Anfragen verzögert, was dazu führt, dass alle Clients die Service-Levels nicht einhalten.
- Bereitstellung interner Warteschlangen für Clients, anstatt APIs verfügbar zu machen, die den Arbeitseingang verwalten und Anfragen in internen Warteschlangen platzieren.
- Wenn zu viele Arbeitsanforderungstypen in einer einzigen Warteschlange zusammengefasst werden, kann dies die Backlog-Bedingungen verschärfen, da der Ressourcenbedarf auf die verschiedenen Anforderungstypen verteilt wird.
- Verarbeitung komplexer und einfacher Anfragen in derselben Warteschlange, obwohl unterschiedliche Überwachungs-, Timeout- und Ressourcenzuweisungen erforderlich sind.
- Keine Validierung von Eingaben oder Nutzung von Aussagen, um Mechanismen für schnelles Scheitern in Software zu implementieren, die Ausnahmen an übergeordnete Komponenten weiterleiten, die Fehler problemlos verarbeiten können.
- Keine Entfernung fehlerhafter Ressourcen aus der Anforderungsweiterleitung, insbesondere bei Ausfällen ohne erkennbare Ursache mit sowohl erfolgreicher als auch fehlgeschlagener Verarbeitung aufgrund von Abstürzen und Neustarts, zeitweise auftretenden Abhängigkeitsfehlern, verringerter Kapazität oder Verlust von Netzwerkpaketen.

Vorteile der Nutzung dieser bewährten Methode: Systeme, die schnelles Scheitern nutzen, lassen sich leichter debuggen und korrigieren und weisen häufig Probleme im Code und in der Konfiguration auf, bevor Releases für die Produktion veröffentlicht werden. Systeme, die effektive Warteschlangenstrategien beinhalten, sind widerstandsfähiger und zuverlässiger bei Traffic-Spitzen und zeitweiligen Systemstörungen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Strategien für schnelles Scheitern können sowohl in Softwarelösungen als auch in der Infrastruktur konfiguriert werden. Warteschlangen scheitern nicht nur schnell, sondern sind auch eine einfache und dennoch leistungsstarke Architekturtechnik zur Entkopplung von Systemkomponenten für eine ausgeglichene Auslastung. [Amazon CloudWatch](#) bietet Funktionen zur Überwachung von Ausfällen und zur Warnung bei Ausfällen. Sobald erkannt wird, dass ein System ausfällt, können Strategien zur Schadensbegrenzung umgesetzt werden, darunter auch der Wechsel weg von knapp werdenden Ressourcen. Wenn in Systemen Warteschlangen mit [Amazon SQS](#) und anderen Warteschlangentechnologien implementiert werden, um eine ausgeglichene Auslastung zu gewährleisten, muss berücksichtigt werden, wie Warteschlangentrückstände sowie Fehler beim Nachrichtenabruf verwaltet werden können.

Implementierungsschritte

- Implementieren Sie programmatische Aussagen oder spezifische Metriken in Ihrer Software und verwenden Sie diese, um explizit Alarme bei Systemproblemen auszulösen. Amazon CloudWatch hilft Ihnen bei der Erstellung von Metriken und Alarmen auf der Grundlage des Anwendungsprotokollmusters und der SDK-Instrumentierung.
- Verwenden Sie CloudWatch-Metriken und Alarme, um knappe Ressourcen zu erkennen, die die Latenz bei der Verarbeitung erhöhen oder Anfragen wiederholt nicht bearbeiten können.
- Nutzen Sie asynchrone Verarbeitung, indem Sie APIs entwerfen, die Anfragen annehmen und an interne Warteschlangen anhängen. Verwenden Sie dazu Amazon SQS und senden Sie dann eine Erfolgsmeldung an den Nachrichten-Client, sodass der Client Ressourcen freigeben und mit anderen Arbeiten fortfahren kann, während die Verbraucher der Backend-Warteschlangen Anfragen verarbeiten.
- Messen und überwachen Sie die Latenz bei der Verarbeitung von Warteschlangen, indem Sie jedes Mal, wenn Sie eine Nachricht aus einer Warteschlange nehmen, eine CloudWatch-Metrik erstellen, indem Sie die aktuelle Uhrzeit mit dem Nachrichtenzeitstempel vergleichen.
- Wenn Fehler eine erfolgreiche Nachrichtenverarbeitung verhindern oder der Datenverkehr so stark ansteigt, dass er im Rahmen der Service Level Agreements nicht verarbeitet werden kann, wird älterer oder überschüssiger Datenverkehr in eine Überlaufwarteschlange ausgelagert. So können vorrangig neuere Aufträge verarbeitet werden. Ältere Aufträge werden verarbeitet, sobald Kapazitäten frei werden. Diese Technik ist eine Annäherung an die LIFO-Verarbeitung und ermöglicht eine normale Systemverarbeitung für alle neuen Aufträge.

- Verwenden Sie Warteschlangen für unzustellbare Nachrichten oder Redrive-Warteschlangen, um Nachrichten, die nicht verarbeitet werden können, aus dem Backlog an einen Ort zu verschieben, der später geprüft und verarbeitet werden kann.
- Versuchen Sie es entweder erneut oder, sofern dies tolerierbar ist, löschen Sie alte Nachrichten, indem Sie die tatsächliche Zeit mit dem Nachrichtenzeitstempel vergleichen und Nachrichten verwerfen, die für den anfragenden Client nicht mehr relevant sind.

Ressourcen

Zugehörige bewährte Methoden:

- [REL04-BP02 Lose gekoppelte Abhängigkeiten implementieren](#)
- [REL05-BP02 Drosselung von Anfragen](#)
- [REL05-BP03 Steuerung und Begrenzung von Wiederholungsaufrufen](#)
- [REL06-BP02 Definieren und Berechnen von Metriken \(Aggregation\)](#)
- [REL06-BP07 Überwachen der gesamten Nachverfolgung von Anfragen im System](#)

Zugehörige Dokumente:

- [Vermeiden von nicht mehr aufzuholenden Rückständen](#)
- [Fail Fast](#)
- [How can I prevent an increasing backlog of messages in my Amazon SQS queue?](#)
- [Elastic Load Balancing: Zonal Shift](#)
- [Amazon Application Recovery Controller: Routing-Steuerung für Datenverkehr-Failover](#)

Zugehörige Beispiele:

- [Enterprise Integration Patterns: Dead Letter Channel](#)

Zugehörige Videos:

- [AWS re:Invent 2022 - Operating highly available Multi-AZ applications](#)

Zugehörige Tools:

- [Amazon SQS](#)
- [Amazon MQ](#)
- [AWS IoT Core](#)
- [Amazon CloudWatch](#)

REL05-BP05 Client-Timeouts festlegen

Legen Sie angemessene Zeitüberschreitungen für Verbindungen und Anfragen fest, überprüfen Sie sie systematisch und verlassen Sie sich nicht auf Standardwerte, da sie nicht Workload-spezifisch sind.

Gewünschtes Ergebnis: Client-Zeitüberschreitungen sollten die Kosten für Client, Server und Workload berücksichtigen, die mit dem Warten auf Anfragen verbunden sind, deren Bearbeitung ungewöhnlich lange dauert. Da es nicht möglich ist, die genaue Ursache einer Zeitüberschreitung zu ermitteln, müssen Clients ihr Wissen über Services nutzen, um Erwartungen hinsichtlich wahrscheinlicher Ursachen und geeigneter Zeitüberschreitungen zu entwickeln.

Bei Client-Verbindungen kommt es aufgrund der konfigurierten Werte zu einer Zeitüberschreitung. Nach einer Zeitüberschreitung entscheidet der Client entweder, die Anfrage abubrechen und es erneut zu versuchen oder er öffnet einen [Unterbrecher](#). Durch diese Muster wird vermieden, dass Anfragen gestellt werden, die einen zugrunde liegenden Fehlerzustand verschlimmern könnten.

Typische Anti-Muster:

- Systemzeitüberschreitungen oder standardmäßige Zeitüberschreitungen werden nicht beachtet.
- Normale Abschlusszeit für Anfragen ist nicht bekannt.
- Mögliche Ursachen, warum die Bearbeitung von Anfragen ungewöhnlich lange dauert, oder die Kosten für die Client-, Service- oder Workload-Leistung, die während des Wartens darauf, dass diese Anfragen abgeschlossen werden, anfallen, sind nicht bekannt.
- Die Wahrscheinlichkeit, dass ein gestörtes Netzwerk dazu führt, dass eine Anfrage erst dann fehlschlägt, wenn die Zeitüberschreitung erreicht ist, und die Kosten für die Client- und Workload-Leistung, die entstehen, wenn keine kürzere Zeitüberschreitung gewählt wird, sind nicht bekannt.
- Zeitüberschreitungsszenarien sowohl für Verbindungen als auch für Anfragen werden nicht getestet.
- Zu hohe Zeitüberschreitungen können zu langen Wartezeiten führen und die Ressourcenauslastung erhöhen.

- Zu niedrige Zeitüberschreitungen führen zu künstlichen Fehlschlägen.
- Muster zur Behandlung von Zeitüberschreitungsfehlern bei Remote-Aufrufen wie Unterbrecher und Wiederholungsversuchen werden übersehen.
- Die Überwachung der Fehlerraten bei Serviceaufrufen, der Service-Level-Ziele für die Latenz und der Latenzausreißer wird nicht in Betracht gezogen. Diese Metriken können Aufschluss über aggressive oder tolerante Zeitüberschreitungen geben.

Vorteile der Nutzung dieser bewährten Methode: Zeitüberschreitungen für Remote-Aufrufe sind konfiguriert und die Systeme sind so konzipiert, dass sie Zeitüberschreitungen ordnungsgemäß behandeln, sodass Ressourcen geschont werden, wenn Remote-Aufrufe ungewöhnlich langsam reagieren und Zeitüberschreitungsfehler von Service-Clients ordnungsgemäß behandelt werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Legen Sie eine Zeitüberschreitung für Verbindungen sowie Anfragen für alle Serviceabhängigkeitsaufrufe und generell für prozessübergreifende Aufrufe fest. Viele Frameworks bieten integrierte Zeitüberschreitungsfunktionen. Seien Sie jedoch vorsichtig, da einige Standardwerte unendlich oder höher als für Ihre Serviceziele akzeptabel sind. Ein zu hoher Wert reduziert die Nützlichkeit der Zeitbeschränkung, da Ressourcen weiterhin verbraucht werden, während der Client auf das Einsetzen der Zeitbeschränkung wartet. Ein zu niedriger Wert kann zu erhöhtem Datenverkehr im Backend und zu erhöhter Latenz führen, da zu viele Anfragen wiederholt werden. In einigen Fällen kann dies zu vollständigen Ausfällen führen, da alle Anfragen wiederholt werden.

Beachten Sie bei der Festlegung von Zeitüberschreitungsstrategien Folgendes:

- Die Bearbeitung von Anfragen kann aufgrund ihres Inhalts, Beeinträchtigungen eines Zieldienstes oder eines Ausfalls einer Netzwerkpartition länger als normal dauern.
- Anfragen mit ungewöhnlich aufwändigem Inhalt könnten unnötige Server- und Client-Ressourcen verbrauchen. In diesem Fall können Ressourcen geschont werden, wenn für diese Anfragen eine Zeitüberschreitung konfiguriert wird und es nicht erneut versucht wird. Services sollten sich auch durch Drosselungen und serverseitige Zeitüberschreitungen vor ungewöhnlich aufwändigen Inhalten schützen.
- Anfragen, die aufgrund einer Servicebeeinträchtigung ungewöhnlich lange dauern, können mit einer Zeitüberschreitung abgebrochen und erneut versucht werden. Die Servicekosten für die Anfrage und den erneuten Versuch sollten berücksichtigt werden. Wenn die Ursache jedoch eine

lokale Beeinträchtigung ist, ist ein erneuter Versuch wahrscheinlich nicht teuer und reduziert den Ressourcenverbrauch des Clients. Die Zeitüberschreitung kann je nach Art der Beeinträchtigung auch Serverressourcen freisetzen.

- Anfragen, deren Bearbeitung lange dauert, weil die Anfrage oder Antwort nicht vom Netzwerk zugestellt wurde, können mit einer Zeitüberschreitung abgebrochen und erneut versucht werden. Da die Anfrage oder Antwort nicht zugestellt wurde, würde sie unabhängig von der Länge der Zeitüberschreitung fehlschlagen. Durch eine Zeitüberschreitung werden in diesem Fall keine Serverressourcen, aber Client-Ressourcen freigegeben und die Workload-Leistung wird verbessert.

Nutzen Sie etablierte Entwurfsmuster wie Wiederholungsversuche und Schutzschalter, um Timeouts elegant zu handhaben und ausfallschnelle Ansätze zu unterstützen. [AWS SDKs](#) und [AWS CLI](#) ermöglichen die Konfiguration von Verbindungs- und Anforderungs-Timeouts sowie von Wiederholungsversuchen mit exponentiellem Backoff und Jitter. [AWS Lambda](#) Funktionen unterstützen die Konfiguration von Timeouts. Damit können Sie Low-Code-Schutzschalter bauen [AWS Step Functions](#), die die Vorteile der vorgefertigten Integrationen mit Diensten und nutzen. AWS SDKs [AWS App Mesh](#) Envoy bietet Funktionen für Zeitüberschreitungen und Unterbrecher an.

Implementierungsschritte

- Konfigurieren Sie Zeitüberschreitungen für Remote-Serviceaufrufe und nutzen Sie die integrierten sprachspezifischen Zeitüberschreitungsfunktionen oder Open-Source-Bibliotheken für Zeitüberschreitungen.
- Wenn Ihr Workload Aufrufe mit einem tätig AWS SDK, finden Sie in der Dokumentation Informationen zur sprachspezifischen Timeout-Konfiguration.
 - [Python](#)
 - [PHP](#)
 - [.NET](#)
 - [Ruby](#)
 - [Java](#)
 - [Go](#)
 - [Node.js](#)
 - [C++](#)
- Wenn Sie AWS SDKs AWS CLI Or-Befehle in Ihrem Workload verwenden, konfigurieren Sie Standard-Timeout-Werte, indem Sie die AWS [Konfigurationsstandardwerte](#) für und festlegen. `connectTimeoutInMillis` `tlsNegotiationTimeoutInMillis`

- Wenden Sie [Befehlszeilenoptionen](#) `cli-read-timeout` an `cli-connect-timeout` und steuern Sie einmalige AWS CLI Befehle auf Dienste. AWS
- Überwachen Sie Remote-Serviceanfragen auf Zeitüberschreitungen und richten Sie Alarmer für anhaltende Fehler ein, sodass Sie proaktiv mit Fehlerszenarien umgehen können.
- Implementieren Sie [CloudWatch Metriken](#) und [CloudWatch Anomalieerkennung](#) zu Anruf Fehlerraten, Service-Level-Zielen für Latenz und Latenz ausreißer, um Einblicke in den Umgang mit übermäßig aggressiven oder toleranten Timeouts zu gewinnen.
- Konfigurieren Sie Zeitüberschreitungen für [Lambda-Funktionen](#).
- APIGateway-Clients müssen bei der Behandlung von Timeouts ihre eigenen Wiederholungsversuche implementieren. APIGateway unterstützt ein [Integrations-Timeout von 50 Millisekunden bis 29 Sekunden für Downstream-Integrationen und versucht es nicht erneut, wenn die Integration ein Timeout](#) anfordert.
- Implementieren Sie das [Unterbrecher](#)-Muster, um zu vermeiden, dass Remote-Aufrufe getätigt werden, wenn Zeitüberschreitungen auftreten. Öffnen Sie die Leitung, um fehlschlagende Aufrufe zu vermeiden, und schließen Sie die Leitung, wenn die Aufrufe normal reagieren.
- Für containerbasierte Workloads können Sie die Funktionen von [App Mesh Envoy](#) nutzen, um von den integrierten Zeitüberschreitungen und Unterbrechern zu profitieren.
- Wird verwendet AWS Step Functions , um Low-Code-Schutzschalter für Remote-Serviceanrufe zu erstellen, insbesondere wenn AWS native SDKs und unterstützte Step Functions Functions-Integrationen aufgerufen werden, um Ihre Arbeitslast zu vereinfachen.

Ressourcen

Zugehörige bewährte Methoden:

- [REL05-BP03 Steuerung und Begrenzung von Wiederholungsaufrufen](#)
- [REL05-BP04 Schnelles Scheitern und Begrenzen von Warteschlangen](#)
- [REL06-BP07 Überwachen der gesamten Nachverfolgung von Anfragen im System](#)

Zugehörige Dokumente:

- [AWS SDK: Wiederholungsversuche und Timeouts](#)
- [Die Amazon Builders' Library: Timeouts, Wiederholungen und Backoff mit Jitter](#)
- [Amazon API Gateway-Kontingente und wichtige Hinweise](#)

- [AWS Command Line Interface: Command line options](#)
- [AWS SDK for Java 2.x: API Timeouts konfigurieren](#)
- [AWS Botocore verwendet das Konfigurationsobjekt und die Konfigurationsreferenz](#)
- [AWS SDK für .NET: Retries and Timeouts](#)
- [AWS Lambda: Configuring Lambda function options](#)

Zugehörige Beispiele:

- [Verwenden des Circuit Breaker Patterns mit AWS Step Functions Amazon DynamoDB](#)
- [Martin Fowler: CircuitBreaker](#)

Zugehörige Tools:

- [AWS SDKs](#)
- [AWS Lambda](#)
- [Amazon SQS](#)
- [AWS Step Functions](#)
- [AWS Command Line Interface](#)

REL05-BP06 Systeme soweit möglich zustandslos machen

Systeme sollten entweder keinen Zustand erfordern oder ihn so auslagern, dass zwischen verschiedenen Client-Anfragen keine Abhängigkeit von lokal gespeicherten Daten auf der Festplatte und im Arbeitsspeicher besteht. Auf diese Weise können Server nach Belieben ersetzt werden, ohne dass dies Auswirkungen auf die Verfügbarkeit hat.

Wenn Benutzer oder Services mit einer Anwendung interagieren, führen sie häufig eine Reihe von Interaktionen aus, die eine Sitzung bilden. Bei einer Sitzung handelt es sich um eindeutige Daten für Benutzer, die zwischen Anfragen bestehen bleiben, während sie die Anwendung verwenden. Eine zustandslose Anwendung ist eine Anwendung, die keine Informationen zu früheren Interaktionen benötigt und keine Sitzungsinformationen speichert.

Sobald das System so konzipiert ist, dass es zustandslos ist, können Sie serverlose Rechendienste wie oder verwenden. AWS Lambda AWS Fargate

Neben dem Austausch von Servern besteht ein weiterer Vorteil statusfreier Anwendungen darin, dass sie horizontal skaliert werden können, da alle verfügbaren Rechenressourcen (wie EC2 Instanzen und AWS Lambda Funktionen) jede Anforderung bearbeiten können.

Vorteile der Nutzung dieser bewährten Methode: Systeme mit zustandslosem Design lassen sich besser an die horizontale Skalierung anpassen, sodass Kapazitäten je nach vorhandenem Datenverkehr und bestehender Nachfrage hinzugefügt oder entfernt werden können. Sie sind auch inhärent widerstandsfähig gegenüber Ausfällen und bieten Flexibilität und Agilität bei der Anwendungsentwicklung.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Erstellen Sie zustandslose Anwendungen. Zustandslose Anwendungen ermöglichen eine horizontale Skalierung und sind widerstandsfähig gegenüber Ausfällen einzelner Knoten. Analysieren Sie die Komponenten Ihrer Anwendung, die ihren Status innerhalb der Architektur beibehalten. Auf diese Weise können Sie die potenziellen Auswirkungen der Umstellung auf ein zustandsloses Design bewerten. Eine zustandslose Architektur entkoppelt Benutzerdaten und entlädt die Sitzungsdaten. Dies bietet die Flexibilität, jede Komponente unabhängig zu skalieren, um unterschiedlichen Workload-Anforderungen gerecht zu werden und die Ressourcenauslastung zu optimieren.

Implementierungsschritte

- Identifizieren und analysieren Sie die zustandsbehafteten Komponenten in Ihrer Anwendung.
- Entkoppeln Sie Daten, indem Sie Benutzerdaten von der Kernanwendungslogik trennen und verwalten.
 - [Amazon Cognito](#) kann Benutzerdaten mithilfe von Features wie [Identitätspools](#), [Benutzerpools](#) und [Amazon Cognito Sync](#) von Anwendungscode entkoppeln.
 - Sie können [AWS Secrets Manager](#) verwenden, um Benutzerdaten zu entkoppeln, indem Sie Secrets an einem sicheren, zentralen Ort speichern. Das bedeutet, dass der Anwendungscode keine Secrets speichern muss, was seine Sicherheit erhöht.
 - Erwägen Sie die Verwendung von [Amazon S3](#), um große, unstrukturierte Daten wie Bilder und Dokumente zu speichern. Ihre Anwendung kann diese Daten bei Bedarf abrufen, sodass sie nicht im Arbeitsspeicher gespeichert werden müssen.
 - Verwenden Sie [Amazon DynamoDB](#), um Informationen wie Benutzerprofile zu speichern. Ihre Anwendung kann diese Daten nahezu in Echtzeit abfragen.
- Verlagern Sie Sitzungsdaten in eine Datenbank, einen Cache oder externe Dateien.

- [Amazon ElastiCache](#), Amazon DynamoDB, [Amazon Elastic File System](#) (AmazonEFS) und [Amazon MemoryDB](#) sind Beispiele für AWS Dienste, die Sie zum Auslagern von Sitzungsdaten verwenden können.
- Entwerfen Sie eine zustandslose Architektur, nachdem Sie festgelegt haben, welche Zustands- und Benutzerdaten in Ihrer bevorzugten Speicherlösung abgelegt werden müssen.

Ressourcen

Zugehörige bewährte Methoden:

- [REL11-BP03 Automatisieren Sie die Heilung auf allen Ebenen](#)

Zugehörige Dokumente:

- [Die Amazon Builders' Library: Vermeiden von Fallback in verteilten Systemen](#)
- [Die Amazon Builders' Library: Vermeiden von nicht mehr aufholbaren Warteschlangen-Rückständen](#)
- [Die Amazon Builders' Library: Herausforderungen und Strategien für das Caching](#)
- [Bewährte Methoden für Stateless Web Tier auf AWS](#)

REL05-BP07 Nothebel einbauen

Nothebel sind schnelle Prozesse, die die Auswirkungen auf die Verfügbarkeit Ihres Workloads mindern können.

Nothebel bewirken, dass das Verhalten von Komponenten oder Abhängigkeiten mithilfe bekannter und getesteter Mechanismen deaktiviert, gedrosselt oder geändert wird. Dadurch können Beeinträchtigungen des Workloads, die durch die Erschöpfung von Ressourcen aufgrund unerwarteter Nachfragesteigerungen verursacht werden, gemildert und die Auswirkungen von Ausfällen bei nicht kritischen Komponenten innerhalb Ihres Workloads reduziert werden.

Gewünschtes Ergebnis: Durch die Implementierung von Nothebeln können Sie bewährte Prozesse einrichten, um die Verfügbarkeit kritischer Komponenten in Ihrem Workload aufrechtzuerhalten. Der Workload sollte sich problemlos reduzieren lassen und auch während der Aktivierung eines Nothebels weiterhin seine geschäftskritischen Funktionen ausführen. Weitere Informationen zur graziösen Degradation finden Sie unter [REL05-BP01 Implementieren Sie eine graziöse Degradation, um entsprechende harte Abhängigkeiten in weiche Abhängigkeiten umzuwandeln](#).

Typische Anti-Muster:

- Der Ausfall von nicht kritischen Abhängigkeiten wirkt sich auf die Verfügbarkeit Ihres Kern-Workloads aus.
- Das Verhalten kritischer Komponenten wird während der Beeinträchtigung unkritischer Komponenten nicht getestet oder überprüft.
- Es sind keine klaren und deterministischen Kriterien für die Aktivierung oder Deaktivierung eines Nothebels definiert.

Vorteile der Nutzung dieser bewährten Methode: Die Implementierung von Nothebeln kann die Verfügbarkeit der kritischen Komponenten Ihres Workloads verbessern, indem Ihre Resolver mit bewährten Prozessen ausgestattet werden, um auf unerwartete Nachfragespitzen oder Ausfälle von nicht kritischen Abhängigkeiten zu reagieren.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

- Ermitteln Sie die kritischen Komponenten in Ihrem Workload.
- Entwerfen und gestalten Sie die kritischen Komponenten Ihres Workloads so, dass sie Ausfällen von nicht kritischen Komponenten standhalten.
- Führen Sie Tests durch, um das Verhalten Ihrer kritischen Komponenten beim Ausfall von nicht kritischen Komponenten zu überprüfen.
- Definieren und überwachen Sie relevante Metriken oder Auslöser für die Einleitung von Nothebeln.
- Definieren Sie die Verfahren (manuell oder automatisiert), die Bestandteil des Nothebels sind.

Implementierungsschritte

- Ermitteln Sie die kritischen Komponenten in Ihrem Workload.
 - Jede technische Komponente Ihres Workloads sollte der entsprechenden Geschäftsfunktion zugeordnet und als kritisch oder nicht kritisch eingestuft werden. Beispiele für wichtige und unkritische Funktionen bei Amazon finden Sie unter [Any Day Can Be Prime Day: How Amazon.com Search Uses Chaos Engineering to Handle Over 84K Requests Per Second](#).
- Hierbei handelt es sich sowohl um eine technische als auch um eine geschäftliche Entscheidung, die je nach Organisation und Workload unterschiedlich ausfallen kann.

- Entwerfen und gestalten Sie die kritischen Komponenten Ihres Workloads so, dass sie Ausfällen von nicht kritischen Komponenten standhalten.
- Berücksichtigen Sie bei der Abhängigkeitsanalyse alle potenziellen Fehlermodi und stellen Sie sicher, dass Ihre Notfallmechanismen die kritischen Funktionen an nachgelagerte Komponenten weitergeben.
- Führen Sie Tests durch, um das Verhalten Ihrer kritischen Komponenten bei der Aktivierung Ihrer Nothebel zu überprüfen.
- Vermeiden Sie bimodales Verhalten. Weitere Informationen finden Sie unter [REL11-BP05 Verwenden Sie statische](#) Stabilität, um bimodales Verhalten zu verhindern.
- Definieren und überwachen Sie relevante Metriken und lassen Sie gegebenenfalls einen Alarm auslösen, um einen Nothebel einzuleiten.
- Die richtigen Metriken zur Überwachung zu finden, hängt von Ihrem Workload ab. Einige Beispielmetriken sind die Latenzzeit oder die Anzahl der fehlgeschlagenen Anfragen an eine Abhängigkeit.
- Definieren Sie die manuellen oder automatisierten Verfahren, die Bestandteil des Nothebels sind.
- Dazu können Mechanismen wie [Lastabwurf](#), [Drosselung von Anfragen](#) oder die Implementierung einer [ordnungsgemäßen Funktionsminderung](#) gehören.

Ressourcen

Zugehörige bewährte Methoden:

- [REL05-BP01 Implementieren Sie eine schrittweise Degradation, um anwendbare harte Abhängigkeiten in weiche Abhängigkeiten umzuwandeln](#)
- [REL05-BP02 Drosseln Sie Anfragen](#)
- [REL11-BP05 Verwenden Sie statische Stabilität, um bimodales Verhalten zu verhindern](#)

Zugehörige Dokumente:

- [Automating safe, hands-off deployments](#)
- [Any Day Can Be Prime Day: How Amazon.com Search Uses Chaos Engineering to Handle Over 84K Requests Per Second](#)

Zugehörige Videos:

- [AWS re:Invent 2020: Zuverlässigkeit, Konsistenz und Vertrauen durch Unveränderlichkeit](#)

Änderungsmanagement

Fragen

- [REL 6. Wie werden Ihre Workload-Ressourcen überwacht?](#)
- [REL 7. Wie wird die Workload so gestaltet, dass sie sich an Veränderungen der Nachfrage anpasst?](#)
- [REL 8. Wie werden Veränderungen implementiert?](#)

REL 6. Wie werden Ihre Workload-Ressourcen überwacht?

Protokolle und Metriken sind leistungsstarke Tools, mit denen Sie sich einen Überblick über den Zustand Ihrer Workload verschaffen können. Sie können Ihre Workload so konfigurieren, dass Protokolle und Metriken überwacht und Benachrichtigungen gesendet werden, wenn Schwellenwerte überschritten werden oder wichtige Ereignisse auftreten. Dank der Überwachung kann die Workload erkennen, wenn Schwellenwerte für eine niedrige Leistung unterschritten werden oder Ausfälle auftreten, sodass als Reaktion drauf eine automatische Wiederherstellung erfolgen kann.

Best Practices

- [REL06-BP01 Überwachen aller Komponenten der Workload \(Generierung\)](#)
- [REL06-BP02 Definieren und Berechnen von Metriken \(Aggregation\)](#)
- [REL06-BP03 Senden von Benachrichtigungen \(Verarbeitung und Benachrichtigung in Echtzeit\)](#)
- [REL06-BP04 Automatisieren von Antworten \(Verarbeitung und Benachrichtigung in Echtzeit\)](#)
- [REL06-BP05 Analysieren von Protokollen](#)
- [REL06-BP06 Regelmäßiges Durchführen von Prüfungen von Umfang und Metriken](#)
- [REL06-BP07 Überwachen der gesamten Nachverfolgung von Anfragen im System](#)

REL06-BP01 Überwachen aller Komponenten der Workload (Generierung)

Überwachen Sie die Komponenten der Workload mit Amazon CloudWatch oder Tools von Drittanbietern. Überwachen Sie AWS-Services mit dem AWS Health Dashboard.

Alle Komponenten Ihres Workloads sollten überwacht werden, einschließlich Frontend, Geschäftslogik und Speicherstufen. Definieren Sie Schlüsselmetriken, beschreiben Sie, wie Sie

diese gegebenenfalls aus Protokollen extrahieren, und legen Sie Schwellenwerte für das Auslösen entsprechender Alarmereignisse fest. Stellen Sie sicher, dass die Metriken für die wichtigen Leistungskennzahlen (KPIs) Ihrer Workload relevant sind und verwenden Sie Metriken und Protokolle, um frühe Warnzeichen einer Serviceverschlechterung zu identifizieren. Beispielsweise kann eine mit Geschäftsergebnissen zusammenhängende Metrik wie etwa die Anzahl der pro Minute erfolgreich verarbeiteten Bestellungen schneller auf Workload-Probleme hinweisen als eine technische Metrik wie etwa die CPU-Auslastung. Verwenden Sie das AWS Health Dashboard für eine personalisierte Ansicht der Leistung und Verfügbarkeit der AWS-Services, die Ihren AWS-Ressourcen.

Die Überwachung in der Cloud bietet neue Möglichkeiten. Die meisten Cloudanbieter haben anpassbare Hooks entwickelt und können Einblicke liefern, mit denen Sie mehrere Ebenen Ihrer Workload überwachen können. AWS-Services wie Amazon CloudWatch wenden statistische und Machine-Learning-Algorithmen an, um Metriken von Systemen und Anwendungen kontinuierlich zu analysieren, normale Basiswerte zu erkennen und Oberflächenanomalien anhand eines minimalen Benutzereingriffs aufzudecken. Anomalieerkennungsalgorithmen berücksichtigen saisonale und trendbasierte Änderungen von Metriken.

AWS stellt zahlreiche Überwachungs- und Protokollinformationen bereit, die genutzt werden können, um workload-spezifische Metriken und Bedarfsänderungsprozesse zu definieren und Machine-Learning-Verfahren unabhängig von der ML-Erfahrung einzuführen.

Zudem können Sie auch all Ihre externen Endpunkte überwachen, um sicherzustellen, dass diese von Ihrer Basisimplementierung unabhängig sind. Diese aktive Überwachung kann anhand von synthetischen Transaktionen erfolgen (auch Benutzer-Canaries genannt, jedoch nicht zu verwechseln mit Canary-Bereitstellungen). Diese führen regelmäßig eine Reihe gängiger Aufgaben aus, die mit Aktionen übereinstimmen, die von Clients der Workload durchgeführt werden. Diese Aufgaben sollten nicht zu lang sein und Sie sollten darauf achten, Ihre Workload beim Testen nicht zu überlasten. Mit Amazon CloudWatch Synthetics können Sie [synthetische Canaries erstellen](#), um Ihre Endpunkte und APIs zu überwachen. Sie können die synthetischen Canary-Client-Knoten auch mit der AWS X-Ray-Konsole kombinieren, um zu bestimmen, bei welchen synthetischen Canaries im ausgewählten Zeitraum Probleme mit Fehlern, Störungen oder Drosselungsraten auftreten.

Gewünschtes Ergebnis:

Erfassen und Nutzen kritischer Metriken aus allen Komponenten der Workload, um die Workload-Zuverlässigkeit und eine optimale Benutzererfahrung sicherzustellen. Wenn Sie erkennen, dass mit einem Workload keine Geschäftsergebnisse erzielt werden, können Sie schnell einen Systemausfall deklarieren und das System nach einem Vorfall wiederzustellen.

Typische Anti-Muster:

- Es werden nur externe Schnittstellen zum Workload überwacht.
- Es werden keine workload-spezifischen Metriken erzeugt und Sie verlassen sich nur auf Metriken, die Ihnen von den AWS-Services, die Ihre Workload verwendet, bereitgestellt werden.
- Es werden nur technische Metriken in Ihrer Workload verwendet und es werden keinerlei Metriken im Zusammenhang mit nicht-technischen KPIs, zu denen die Workload beiträgt, überwacht.
- Sie verlassen sich auf den Produktionsdatenverkehr und einfache Zustandsprüfungen für die Überwachung und Bewertung des Workload-Status.

Vorteile der Nutzung dieser bewährten Methode: Durch die Überwachung aller Ebenen Ihrer Workload können Sie Probleme in den darin enthaltenen Komponenten schneller vorhersehen und beheben.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

1. Aktivieren Sie die Protokollierung, wann immer verfügbar. Von allen Workload-Komponenten sollten Überwachungsdaten erzielt werden. Aktivieren Sie eine zusätzliche Protokollierung, wie etwa S3 Access Logs, und ermöglichen Sie es Ihrer Workload, die workload-spezifischen Daten zu protokollieren. Erfassen Sie Metriken für die Durchschnittswerte zu CPU, Netzwerk-E/A und Laufwerk-E/A von Services wie Amazon ECS, Amazon EKS, Amazon EC2, Elastic Load Balancing, AWS Auto Scaling und Amazon EMR Unter [AWS Services That Publish CloudWatch Metrics](#) finden Sie eine Liste an AWS-Services, die Metriken in CloudWatch veröffentlichen.
2. Sehen Sie sich alle Standardmetriken an, um mehr über mögliche Datenerfassungslücken zu erfahren. Jeder Service generiert Standardmetriken. Durch die Erfassung von Standardmetriken erhalten Sie ein besseres Verständnis über die Abhängigkeiten zwischen Workload-Komponenten und darüber, wie die Komponentenzuverlässigkeit und -leistung die Workload beeinträchtigen. Sie können auch [Ihre eigenen Metriken](#) in CloudWatch unter Verwendung der AWS CLI oder einer API erstellen und veröffentlichen.
3. Bewerten Sie alle Metriken, um zu entscheiden, für welche eine Warnmeldung für jeden AWS-Service in Ihrer Workload eingerichtet werden soll. Sie können eine Metriken-Untergruppe auswählen, die eine höhere Auswirkung auf die Workload-Zuverlässigkeit hat. Wenn Sie sich auf kritische Metriken und Schwellenwerte konzentrieren, können Sie die Anzahl an [Warnmeldungen](#) genauer definieren und so Falschmeldungen reduzieren.

4. Definieren Sie Warnungen und den Wiederherstellungsprozess für Ihre Workload nach dem Auslösen der Warnmeldung. Das Definieren von Warnmeldungen ermöglicht es Ihnen, schnell zu benachrichtigen, zu eskalieren und die für die Wiederherstellung nach einem Vorfall erforderlichen Schritte durchzuführen, um so Ihren festgelegten Recovery Time Objective (RTO) zu erfüllen. Sie können [Amazon CloudWatch-Alarme](#) für das Aufrufen von automatisierten Workflows und die Initiierung von Wiederherstellungsverfahren basierend auf definierten Schwellenwerten verwenden.
5. Erfahren Sie mehr über die Verwendung von synthetischen Transaktionen für das Erfassen relevanter Daten zum Workload-Status. Die synthetische Überwachung folgt denselben Routen und führt dieselben Aktionen aus wie ein Kunde. Dadurch haben Sie die Möglichkeit, die Kundenerfahrung kontinuierlich zu überprüfen, selbst, wenn Sie keinen Kundendatenverkehr auf Ihren Workloads haben. Durch die Verwendung von [synthetischen Transaktionen](#) können Sie Probleme erkennen, bevor Ihre Kunden dies tun.

Ressourcen

Zugehörige bewährte Methoden:

- [REL11-BP03 Automatisieren der Reparatur auf allen Ebenen](#)

Zugehörige Dokumente:

- [Getting started with your AWS Health Dashboard – Your account health](#)
- [AWS Services That Publish CloudWatch Metrics](#)
- [Access Logs for Your Network Load Balancer](#)
- [Access logs for your application load balancer](#)
- [Accessing Amazon CloudWatch Logs for AWS Lambda](#)
- [Amazon-S3-Server-Zugriffsprotokollierung](#)
- [Enable Access Logs for Your Classic Load Balancer](#)
- [Exporting log data to Amazon S3](#)
- [Install the CloudWatch agent on an Amazon EC2 instance](#)
- [Publishing Custom Metrics](#)
- [Using Amazon CloudWatch Dashboards](#)
- [Verwenden von Amazon-CloudWatch-Metriken](#)

- [Using Canaries \(Amazon CloudWatch Synthetics\)](#)
- [What are Amazon CloudWatch Logs?](#)

Benutzerhandbücher:

- [Creating a trail](#)
- [Monitoring memory and disk metrics for Amazon EC2 Linux instances](#)
- [Using CloudWatch Logs with container instances](#)
- [VPC Flow Logs](#)
- [What is Amazon DevOps Guru?](#)
- [Was ist AWS X-Ray?](#)

Verwandte Blogs:

- [Debugging with Amazon CloudWatch Synthetics and AWS X-Ray](#)

Zugehörige Beispiele:

- [The Amazon Builders' Library: Instrumentieren verteilter Systeme für Einblicke in die Betriebsabläufe](#)
- [Workshop zur Beobachtbarkeit](#)

REL06-BP02 Definieren und Berechnen von Metriken (Aggregation)

Erfassen Sie Metriken und Protokolle aus Ihren Workload-Komponenten und berechnen Sie daraus relevante aggregierte Metriken. Diese Metriken bieten eine umfassende und tiefgehende Beobachtbarkeit über Ihre Workloads und können Ihre Resilienzlage erheblich verbessern.

Beobachtbarkeit bedeutet mehr als nur das Erfassen von Metriken aus Workload-Komponenten und die Möglichkeit, diese einzusehen und entsprechende Warnmeldungen zu erstellen. Es geht vielmehr darum, ein ganzheitliches Verständnis des Verhaltens Ihrer Workloads zu erhalten. Diese Verhaltensinformationen stammen aus allen Komponenten Ihrer Workloads, einschließlich der Cloud-Services, von denen sie abhängen, gut ausgearbeiteten Protokolle und Metriken. Diese Daten geben Ihnen einen Überblick über das Verhalten Ihrer Workloads insgesamt sowie ein detailliertes Verständnis der Interaktion jeder Komponente mit jeder Arbeitseinheit.

Gewünschtes Ergebnis:

- Sie erfassen Protokolle Ihrer Workload-Komponenten und AWS-Serviceabhängigkeiten und veröffentlichen sie an einem zentralen Ort, wo sie leicht abgerufen und verarbeitet werden können.
- Ihre Protokolle enthalten äußerst präzise Zeitstempel.
- Ihre Protokolle enthalten relevante Informationen über den Verarbeitungskontext, wie z. B. eine Ablaufverfolgungs-ID, eine Benutzer- oder Kontokennung und eine Remote-IP-Adresse.
- Sie erstellen aus Ihren Protokollen aggregierte Metriken, die das Verhalten Ihrer Workloads aus einer übergeordneten Perspektive darstellen.
- Sie können Ihre aggregierten Protokolle abfragen, um tiefe und relevante Einblicke in Ihre Workloads zu gewinnen und tatsächliche und potenzielle Probleme zu identifizieren.

Typische Anti-Muster:

- Sie erfassen keine relevanten Protokolle oder Metriken von den Computing-Instanzen, auf denen Ihre Workloads ausgeführt werden, oder von den Cloud-Services, die sie verwenden.
- Sie übersehen die Erfassung von Protokollen und Metriken, die Ihre Leistungskennzahlen (KPIs) betreffen.
- Sie analysieren die auf Workloads bezogene Telemetrie isoliert, ohne Aggregation und Korrelation.
- Sie lassen zu, dass Metriken und Protokolle zu schnell ablaufen, was Trendanalysen und die Identifizierung wiederkehrender Probleme erschwert.

Vorteile der Nutzung dieser bewährten Methoden: Sie können mehr Anomalien erkennen und Ereignisse und Metriken zwischen verschiedenen Komponenten Ihrer Workloads korrelieren. Sie können anhand Ihrer Workload-Komponenten Erkenntnisse gewinnen, die auf Informationen in Protokollen basieren, die häufig nicht nur in Form von Metriken verfügbar sind. Sie können Fehlerursachen schneller ermitteln, indem Sie Ihre Protokolle in großem Umfang abfragen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Identifizieren Sie die Quellen von Telemetriedaten, die für Ihre Workloads und deren Komponenten relevant sind. Diese Daten stammen nicht nur aus Komponenten, die Metriken veröffentlichen, wie Ihrem Betriebssystem (OS) und Anwendungslaufzeiten wie Java, sondern auch aus Anwendungs- und Cloud-Serviceprotokollen. Beispielsweise protokollieren Webserver in der Regel jede Anfrage mit detaillierten Informationen wie Zeitstempel, Verarbeitungslatenz, Benutzer-ID, Remote-IP-Adresse, Pfad und Abfragezeichenfolge. Der Detaillierungsgrad dieser Protokolle hilft Ihnen dabei, detaillierte

Abfragen durchzuführen und Metriken zu generieren, die sonst möglicherweise nicht verfügbar gewesen wären.

Erfassen Sie die Metriken und Protokolle mithilfe geeigneter Tools und Prozesse. Protokolle, die von Anwendungen generiert werden, die auf einer Amazon-EC2-Instance ausgeführt werden, können von einem Agenten wie dem [Amazon CloudWatch Agent](#) erfasst und in einem zentralen Speicherservice wie [Amazon CloudWatch Logs](#) veröffentlicht werden. AWS-verwaltete Computing-Services wie [AWS Lambda](#) und [Amazon Elastic Container Service](#) veröffentlichen Protokolle automatisch in CloudWatch Logs für Sie. Aktivieren Sie die Protokollerfassung für AWS-Speicher- und Verarbeitungsservices, die von Ihren Workloads wie [Amazon CloudFront](#), [Amazon S3](#), [Elastic Load Balancing](#) und [Amazon API Gateway](#) verwendet werden.

Reichern Sie Ihre Telemetriedaten mit [Dimensionen](#) an, anhand derer Sie Verhaltensmuster klarer erkennen und korrelierte Probleme anhand von Gruppen verwandter Komponenten isolieren können. Nach dem Hinzufügen können Sie das Verhalten der Komponenten detaillierter beobachten, korrelierte Fehler erkennen und geeignete Abhilfemaßnahmen ergreifen. Beispiele für nützliche Dimensionen sind Availability Zone, EC2-Instance-ID und Container-Task- oder Pod-ID.

Sobald Sie die Metriken und Protokolle erfasst haben, können Sie Abfragen erstellen und daraus aggregierte Metriken generieren, die nützliche Einblicke in normales und anomales Verhalten bieten. Sie können beispielsweise [Amazon CloudWatch Logs Insights](#) verwenden, um benutzerdefinierte Metriken aus Ihren Anwendungsprotokollen abzuleiten, [Amazon CloudWatch Metrics Insights](#), um Ihre Metriken in großem Umfang abzufragen, [Amazon CloudWatch Container Insights](#) zum Sammeln, Aggregieren und Zusammenfassen von Metriken und Protokollen aus Ihren containerisierten Anwendungen und Microservices oder [Amazon CloudWatch Lambda Insights](#), wenn Sie AWS Lambda-Funktionen verwenden. Um eine aggregierte Fehlerratenmetrik zu erstellen, können Sie einen Zähler jedes Mal erhöhen, wenn eine Fehlerantwort oder Meldung in Ihren Komponentenprotokollen gefunden wird, oder Sie können den Gesamtwert einer vorhandenen Fehlerratenmetrik berechnen. Sie können diese Daten verwenden, um Histogramme zu generieren, die das Verhalten von Vorgängen zeigen, z. B. die Anfragen oder Prozesse mit der schlechtesten Leistung. Mithilfe von Lösungen wie der [Anomalieerkennung](#) von CloudWatch Logs können Sie diese Daten auch in Echtzeit nach anomalen Mustern durchsuchen. Diese Erkenntnisse können in Dashboards platziert werden, sodass sie Ihren Anforderungen und Präferenzen entsprechend organisiert werden können.

Mithilfe von Protokollabfragen können Sie besser verstehen, wie bestimmte Anfragen von Ihren Workload-Komponenten bearbeitet wurden, und Anforderungsmuster oder andere Kontexte aufdecken, die sich auf die Resilienz Ihres Workloads auswirken. Es kann nützlich sein, Abfragen auf

der Grundlage Ihres Wissens über das Verhalten Ihrer Anwendungen und anderer Komponenten im Voraus zu recherchieren und vorzubereiten, so dass Sie sie bei Bedarf einfacher ausführen können. Beispielsweise können Sie mit [CloudWatch Logs Insights](#) Protokolldaten in Amazon CloudWatch Logs interaktiv durchsuchen und analysieren. Sie können auch [Amazon Athena](#) verwenden, um Protokolle aus mehreren Quellen, einschließlich [zahlreicher AWS-Services](#), im Petabyte-Bereich abzufragen.

Wenn Sie eine Richtlinie zur Aufbewahrung von Protokollen definieren, sollten Sie den Wert historischer Protokolle berücksichtigen. Historische Protokolle können dabei helfen, langfristige Nutzungs- und Verhaltensmuster, Regressionen und Leistungsverbesserungen Ihrer Workloads zu identifizieren. Dauerhaft gelöschte Protokolle können später nicht analysiert werden. Der Wert historischer Protokolle nimmt jedoch über längere Zeiträume hinweg tendenziell ab. Wählen Sie eine Richtlinie, die Ihren Anforderungen angemessen Rechnung trägt und alle gesetzlichen oder vertraglichen Anforderungen erfüllt, denen Sie möglicherweise unterliegen.

Implementierungsschritte

1. Wählen Sie die Erfassungs-, Speicher-, Analyse- und Anzeigemechanismen für Ihre Beobachtbarkeitsdaten.
2. Installieren und konfigurieren Sie Metrik- und Protokollkollektoren für die entsprechenden Komponenten Ihrer Workloads (z. B. für Amazon-EC2-Instances und in [Sidecar-Containern](#)). Konfigurieren Sie diese Kollektoren so, dass sie automatisch neu gestartet werden, wenn sie unerwartet beendet werden. Aktivieren Sie die Festplatten- oder Speicherpufferung für die Kollektoren, sodass temporäre Veröffentlichungsfehler Ihre Anwendungen nicht beeinträchtigen oder zu Datenverlusten führen.
3. Aktivieren Sie die Protokollierung für AWS-Services, die Sie im Rahmen Ihrer Workloads verwenden, und leiten Sie diese Protokolle an den ausgewählten Speicherservice weiter, wenn notwendig. Ausführliche Anweisungen finden Sie in den Benutzer- oder Entwicklerhandbüchern der jeweiligen Services.
4. Definieren Sie die für Ihre Workloads relevanten betrieblichen Metriken, die auf Ihren Telemetriedaten basieren. Diese können auf direkten Metriken basieren, die von Ihren Workload-Komponenten ausgegeben werden, zu denen auch Metriken im Zusammenhang mit Geschäftskennzahlen oder die Ergebnisse aggregierter Berechnungen wie Summen, Raten, Perzentile oder Histogramme gehören können. Berechnen Sie diese Metriken mit Ihrem Protokollanalysator und platzieren Sie sie gegebenenfalls in Dashboards.
5. Bereiten Sie geeignete Protokollabfragen vor, um Workload-Komponenten, Anfragen oder das Transaktionsverhalten nach Bedarf zu analysieren.

6. Definieren und aktivieren Sie eine Protokollaufbewahrungsrichtlinie für Ihre Komponentenprotokolle. Löschen Sie regelmäßig Protokolle, wenn sie älter sind, als es die Richtlinie zulässt.

Ressourcen

Zugehörige bewährte Methoden:

- [REL06-BP01 Überwachen aller Komponenten des Workloads \(Generierung\)](#)
- [REL06-BP03 Senden von Benachrichtigungen \(Verarbeitung und Benachrichtigung in Echtzeit\)](#)
- [REL06-BP04 Automatisieren von Antworten \(Verarbeitung und Benachrichtigung in Echtzeit\)](#)
- [REL06-BP05 Analysieren von Protokollen](#)
- [REL06-BP06 Regelmäßiges Durchführen von Prüfungen von Umfang und Metriken](#)
- [REL06-BP07 Überwachen der gesamten Nachverfolgung von Anfragen im System](#)

Zugehörige Dokumentation:

- [Funktionsweise von Amazon CloudWatch](#)
- [Amazon Managed Prometheus](#)
- [Amazon Managed Grafana](#)
- [Analysieren von Protokolldaten mit CloudWatch Logs Insights](#)
- [Amazon CloudWatch Lambda Insights](#)
- [Amazon CloudWatch Container Insights](#)
- [Abfragen Ihrer Metriken mit CloudWatch Metrics Insights](#)
- [AWS Distro for OpenTelemetry](#)
- [Amazon CloudWatch Logs Insights Sample Queries](#)
- [Debuggen mit Amazon CloudWatch Synthetics und AWS X-Ray](#)
- [Suchen und Filtern von Protokolldaten](#)
- [Senden von Protokollen direkt an Amazon S3](#)
- [Die Amazon Builders' Library: Verteilte Systeme instrumentieren, um betriebliche Transparenz zu erzielen](#)

Zugehörige Workshops:

- [Workshop zur Beobachtbarkeit](#)

Zugehörige Tools:

- [AWS Distro for OpenTelemetry \(GitHub\)](#)

REL06-BP03 Senden von Benachrichtigungen (Verarbeitung und Benachrichtigung in Echtzeit)

Wenn Organisationen potenzielle Probleme erkennen, senden sie Benachrichtigungen und Warnungen in Echtzeit an das entsprechende Personal und die entsprechenden Systeme, um schnell und effektiv auf diese Probleme reagieren zu können.

Gewünschtes Ergebnis: Durch die Konfiguration relevanter Alarmer auf der Grundlage von Service- und Anwendungsmetriken ist eine schnelle Reaktion auf operative Ereignisse möglich. Bei Überschreitung der Alarmschwellen werden das entsprechende Personal und die entsprechenden Systeme benachrichtigt, damit sie die zugrunde liegenden Probleme beseitigen können.

Typische Anti-Muster:

- Sie konfigurieren Alarmer mit einem übermäßig hohen Schwellenwert, was dazu führt, dass wichtige Benachrichtigungen nicht gesendet werden können.
- Sie konfigurieren Alarmer mit einem zu niedrigen Schwellenwert, was dazu führt, dass bei wichtigen Warnungen aufgrund des Lärms übermäßiger Benachrichtigungen keine Aktion erfolgt.
- Sie aktualisieren keine Alarmer und ihre Schwellenwerte, wenn sich die Nutzung ändert.
- Bei Alarmen, die am besten durch automatische Aktionen behoben werden, führt das Senden der Benachrichtigung an das Personal, anstatt die automatische Aktion zu generieren, dazu, dass übermäßig viele Benachrichtigungen gesendet werden.

Vorteile der Nutzung dieser bewährten Methode: Das Senden von Benachrichtigungen und Warnungen in Echtzeit an das entsprechende Personal und die entsprechenden Systeme ermöglicht eine frühzeitige Erkennung von Problemen und eine schnelle Reaktion auf betriebliche Vorfälle.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Workloads sollten mit der Verarbeitung und Benachrichtigung in Echtzeit ausgestattet sein, um die Erkennbarkeit von Problemen zu verbessern, die sich auf die Verfügbarkeit der Anwendung

auswirken und als Auslöser für automatische Reaktionen dienen könnten. Organisationen können die Verarbeitung und Benachrichtigung in Echtzeit durchführen, indem sie Warnungen mit definierten Metriken erstellen, um Benachrichtigungen zu erhalten, wenn wichtige Ereignisse eintreten oder eine Metrik einen Schwellenwert überschreitet.

[Amazon CloudWatch](#) ermöglicht es Ihnen, [Metrik-Alarme](#) und zusammengesetzte Alarme mithilfe von CloudWatch-Alarmen zu erstellen, die auf statischen Schwellenwerten, der Erkennung von Unregelmäßigkeiten und anderen Kriterien basieren. Weitere Informationen zu den Alarmtypen, die Sie mit CloudWatch konfigurieren können, finden Sie im [Abschnitt über Alarme in der CloudWatch-Dokumentation](#).

Sie können benutzerdefinierte Ansichten von Metriken und Warnungen Ihrer AWS-Ressourcen für Ihre Teams erstellen, indem Sie [CloudWatch-Dashboards](#) nutzen. Die anpassbaren Startseiten in der CloudWatch-Konsole ermöglichen es Ihnen, Ihre Ressourcen in einer einzigen Ansicht über mehrere Regionen hinweg zu überwachen.

Alarme können mindestens eine Aktion ausführen, z. B. das Senden einer Benachrichtigung an ein [Amazon SNS-Thema](#), das eine [Amazon EC2](#)-Aktion oder eine [Amazon EC2 Auto Scaling](#)-Aktion durchgeführt, oder das Erstellen eines [OpsItem](#)-Elements oder [Vorfalls](#) in AWS Systems Manager.

Amazon CloudWatch verwendet [Amazon SNS](#) zum Senden von Benachrichtigungen, wenn sich der Status des Alarms ändert, und ermöglicht so die Nachrichtenzustellung von den Publishern (Produzenten) an die Subscriber (Verbraucher). Weitere Informationen zum Einrichten von Amazon SNS-Benachrichtigungen finden Sie unter [Configuring Amazon SNS](#).

CloudWatch sendet [EventBridge-Ereignisse](#), wenn ein CloudWatch-Alarm erstellt, aktualisiert oder gelöscht wird oder sich sein Status ändert. Sie können EventBridge mit diesen Ereignissen verwenden, um Regeln zu erstellen, die Aktionen ausführen, z. B. Sie benachrichtigen, wenn sich der Status eines Alarms ändert, oder automatisch Ereignisse in Ihrem Konto mit [Systems Manager-Automatisierung](#) auslösen.

Bleiben Sie auf dem Laufenden mit [AWS Health](#). AWS Health ist die maßgebliche Informationsquelle über den Zustand Ihrer AWS Cloud-Ressourcen. Verwenden Sie AWS Health, um über bestätigte Serviceereignisse benachrichtigt zu werden, so dass Sie schnell Maßnahmen ergreifen können, um etwaige Auswirkungen zu minimieren. Erstellen Sie maßgeschneiderte AWS Health-Ereignisbenachrichtigungen für E-Mail- und Chat-Kanäle über [AWS-Benutzerbenachrichtigungen](#), und integrieren Sie sie programmgesteuert in [Ihre Überwachungs- und Warnmeldungs tools über Amazon EventBridge](#). Wenn Sie AWS Organizations verwenden, können Sie AWS Health-Ereignisse kontenübergreifend zusammenfassen.

Wann sollten Sie EventBridge im Vergleich zu Amazon SNS verwenden?

Sowohl EventBridge als auch Amazon SNS können zur Entwicklung ereignisgesteuerter Anwendungen verwendet werden. Ihre Wahl hängt von Ihren spezifischen Anforderungen ab.

Amazon EventBridge wird empfohlen, wenn Sie eine Anwendung erstellen möchten, die auf Ereignisse aus Ihren eigenen Anwendungen, SaaS-Anwendungen und AWS-Services reagiert. EventBridge ist der einzige ereignisbasierte Service, der direkt in SaaS-Partner von Drittanbietern integriert werden kann. EventBridge nimmt außerdem automatisch Ereignisse von über 200 AWS-Services auf, ohne dass Entwickler Ressourcen in ihrem Konto erstellen müssen.

EventBridge verwendet eine definierte JSON-basierte Struktur für Ereignisse und hilft Ihnen bei der Erstellung von Regeln, die auf den gesamten Ereignistext angewendet werden, um Ereignisse auszuwählen, die an ein [Ziel](#) weitergeleitet werden sollen. EventBridge unterstützt derzeit über 20 AWS-Services als Ziele, darunter [AWS Lambda](#), [Amazon SQS](#), Amazon SNS, [Amazon Kinesis Data Streams](#) und [Amazon Data Firehose](#).

Amazon SNS wird für Anwendungen empfohlen, die eine hohe Verteilung benötigen (Tausende oder Millionen von Endpunkten). Ein gängiges Muster, das wir beobachten, ist, dass Kunden Amazon SNS als Ziel für ihre Regel verwenden, um die Ereignisse zu filtern, die sie benötigen, und dann an mehrere Endpunkte zu verteilen.

Nachrichten sind unstrukturiert und können in jedem Format vorliegen. Amazon SNS unterstützt die Weiterleitung von Nachrichten an sechs verschiedene Zieltypen, darunter Lambda, Amazon SQS, HTTP/S-Endpunkte, SMS, mobile Push-Benachrichtigungen und E-Mail. Die [typische Latenz von Amazon SNS liegt unter 30 Millisekunden](#). Eine Vielzahl von AWS-Services sendet Amazon SNS-Nachrichten, indem sie den Service entsprechend konfigurieren (mehr als 30, einschließlich Amazon EC2, [Amazon S3](#), und [Amazon RDS](#)).

Implementierungsschritte

1. Erstellen Sie einen Alarm mithilfe von [Amazon CloudWatch-Alarmen](#).
 - a. Ein metrischer Alarm überwacht eine einzelne CloudWatch-Metrik oder einen Ausdruck, der von CloudWatch-Metriken abhängig ist. Der Alarm initiiert eine oder mehrere Aktionen auf der Grundlage des Werts der Metrik oder des Ausdrucks im Vergleich zu einem Schwellenwert über eine Reihe von Zeitintervallen. Die Aktion kann darin bestehen, eine Benachrichtigung an ein [Amazon SNS-Thema](#) zu senden, das eine [Amazon EC2](#)-Aktion oder eine [Amazon EC2 Auto Scaling](#)-Aktion durchführt, oder ein [OpsItem](#)-Element oder einen [Vorfall](#) in AWS Systems Manager zu erstellen.

- b. Ein zusammengesetzter Alarm besteht aus einem Regelausdruck, der die Alarmbedingungen anderer von Ihnen erstellter Alarme berücksichtigt. Der zusammengesetzte Alarm wechselt nur dann in den Alarmstatus, wenn alle Regelbedingungen erfüllt sind. Die im Regelausdruck eines zusammengesetzten Alarms angegebenen Alarme können metrische Alarme und zusätzliche zusammengesetzte Alarme enthalten. Zusammengesetzte Alarme können Amazon SNS-Benachrichtigungen senden, wenn sich ihr Status ändert, und sie können Systems Manager [OpsItems](#)-Elemente oder [Vorfälle](#) auslösen, wenn sie in den Alarmzustand wechseln, aber sie können weder Amazon EC2- noch Auto Scaling-Aktionen ausführen.
2. Richten Sie [Amazon SNS-Benachrichtigungen](#) ein. Wenn Sie einen CloudWatch-Alarm erstellen, können Sie ein Amazon SNS-Thema hinzufügen, um eine Benachrichtigung zu senden, wenn sich der Status des Alarms ändert.
3. [Erstellen Sie Regeln in EventBridge](#), die bestimmten CloudWatch-Alarmen entsprechen. Jede Regel unterstützt mehrere Ziele, einschließlich Lambda-Funktionen. Jede Regel unterstützt mehrere Ziele, einschließlich Lambda-Funktionen. Sie können beispielsweise einen Alarm definieren, der initiiert wird, wenn der verfügbare Festplattenspeicher knapp wird, wodurch über eine EventBridge-Regel eine Lambda-Funktion ausgelöst wird, um den Speicherplatz zu bereinigen. Weitere Informationen zu EventBridge-Zielen finden Sie unter [EventBridge-Ziele](#).

Ressourcen

Zugehörige bewährte Methoden für Well-Architected:

- [REL06-BP01 Überwachen aller Komponenten der Workload \(Generierung\)](#)
- [REL06-BP02 Definieren und Berechnen von Metriken \(Aggregation\)](#)
- [REL12-BP01 Untersuchen von Fehlern mit Playbooks](#)

Zugehörige Dokumente:

- [Amazon CloudWatch](#)
- [CloudWatch Logs Insights](#)
- [Verwenden von Amazon-CloudWatch-Alarmen](#)
- [Using Amazon CloudWatch dashboards](#)
- [Verwenden von Amazon-CloudWatch-Metriken](#)
- [Setting up Amazon SNS notifications](#)

- [CloudWatch anomaly detection](#)
- [CloudWatch Logs data protection](#)
- [Amazon EventBridge](#)
- [Amazon Simple Notification Service](#)

Zugehörige Videos:

- [reinvent 2022 observability videos](#)
- [AWS re:Invent 2022 - Observability best practices at Amazon](#)

Zugehörige Beispiele:

- [Workshop zur Beobachtbarkeit](#)
- [Amazon EventBridge to AWS Lambda with feedback control by Amazon CloudWatch Alarms](#)

REL06-BP04 Automatisieren von Antworten (Verarbeitung und Benachrichtigung in Echtzeit)

Automatisieren Sie bei Erkennung von Ereignissen die erforderlichen Maßnahmen, wie etwa den Austausch fehlerhafter Komponenten.

Die automatische Echtzeitverarbeitung von Alarmen ist implementiert, sodass die Systeme bei Auslösung von Alarmen schnell korrigierend eingreifen und versuchen können, Ausfälle oder Beeinträchtigungen des Services zu verhindern. Zu den automatisierten Reaktionen auf Alarme könnten der Austausch ausgefallener Komponenten, die Anpassung der Rechenkapazität, die Umleitung des Datenverkehrs auf fehlerfreie Hosts, Availability Zones oder andere Regionen sowie die Benachrichtigung der Betreiber gehören.

Gewünschtes Ergebnis: Echtzeitalarme werden ermittelt und die automatische Verarbeitung von Alarmen wird eingerichtet, um die entsprechenden Maßnahmen zur Einhaltung von Service-Level-Zielen und Service Level Agreements (SLAs) einzuleiten. Die Automatisierung kann von der Selbstreparatur einzelner Komponenten bis hin zum Failover eines ganzen Standorts reichen. Die Automatisierung kann von der Selbstreparatur einzelner Komponenten bis hin zum Failover eines ganzen Standorts reichen.

Typische Anti-Muster:

- Fehlen einer genauen Bestandsaufnahme oder eines Katalogs der wichtigsten Echtzeitalarme

- Keine automatischen Reaktionen auf kritische Alarme (z. B. automatische Skalierung, wenn die Rechenkapazität fast erschöpft ist)
- Widersprüchliche Alarmreaktionen
- Fehlen von Standard-Betriebsabläufen (SOPs), an die sich die Bediener halten müssen, wenn sie Alarmmeldungen erhalten
- Keine Überwachung von Konfigurationsänderungen, da unentdeckte Konfigurationsänderungen zu Ausfallzeiten bei Workloads führen können
- Keine Strategie, um unbeabsichtigte Konfigurationsänderungen rückgängig zu machen

Vorteile der Nutzung dieser bewährten Methode: Die Automatisierung der Alarmverarbeitung kann die Ausfallsicherheit des Systems verbessern. Das System ergreift automatisch Korrekturmaßnahmen und reduziert so manuelle Tätigkeiten, bei denen es zu einem menschlichen, fehleranfälligen Eingreifen kommen kann. Der Workload-Betrieb erfüllt die Verfügbarkeitsziele und reduziert Serviceunterbrechungen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Zur wirksamen Verwaltung von Alarmen und zur Automatisierung ihrer Beantwortung kategorisieren Sie die Alarme nach ihrer Kritikalität und Auswirkung, dokumentieren die Reaktionsverfahren und planen die Reaktionen, bevor Sie die Aufgaben einordnen.

Ermitteln Sie Aufgaben, die bestimmte Aktionen erfordern (oft in Runbooks detailliert beschrieben), und untersuchen Sie alle Runbooks und Playbooks, um festzustellen, welche Aufgaben automatisiert werden können. Lassen sich Aktionen definieren, können sie oft auch automatisiert werden. Wenn Aktionen nicht automatisiert werden können, dokumentieren Sie die manuellen Schritte in einer SOP und schulen Sie die Mitarbeiter darin. Hinterfragen Sie kontinuierlich manuelle Prozesse und suchen Sie nach Möglichkeiten zur Automatisierung, um einen Plan für die Automatisierung von Alarmreaktionen zu erstellen und zu verwalten.

Implementierungsschritte

1. Erstellen eines Inventars von Alarmen: Um eine Liste aller Alarme zu erhalten, können Sie die [AWS CLI](#) mit dem [Amazon CloudWatch](#)-Befehl [describe-alarms](#) verwenden. Je nachdem, wie viele Alarme Sie eingerichtet haben, müssen Sie möglicherweise eine Paginierung verwenden, um eine Untergruppe von Alarmen für jeden Anruf aufzurufen. Alternativ können Sie das AWS-SDK verwenden, um die Alarme über [einen API-Aufruf](#) aufzurufen.

2. Dokumentieren aller Alarmaktionen: Aktualisieren Sie ein Runbook mit allen Alarmen und ihren Aktionen, unabhängig davon, ob sie manuell oder automatisiert sind. [AWS Systems Manager](#) bietet vordefinierte Runbooks. Ausführliche Informationen zum Anzeigen von Runbook-Inhalten finden Sie unter [Working with runbooks](#). Ausführliche Informationen zum Anzeigen von Runbook-Inhalten finden Sie unter [View runbook content](#).
3. Einrichten und Verwalten von Alarmaktionen: Für jeden der Alarme, die eine Aktion erfordern, geben Sie die [automatische Aktion mithilfe des CloudWatch-SDK an](#). So können Sie beispielsweise den Zustand Ihrer Amazon EC2-Instances automatisch auf Grundlage eines CloudWatch-Alarms ändern, indem Sie Aktionen für einen Alarm erstellen und aktivieren oder Aktionen für einen Alarm deaktivieren.

Sie können [Amazon EventBridge](#) auch verwenden, um automatisch auf Systemereignisse zu reagieren, z. B. auf Probleme mit der Anwendungsverfügbarkeit oder auf Ressourcenänderungen. Sie können Regeln erstellen, um anzugeben, an welchen Ereignissen Sie interessiert sind, und welche Aktionen auszuführen sind, wenn ein Ereignis mit einer Regel übereinstimmt. Zu den Aktionen, die automatisch ausgelöst werden können, gehören der Aufruf einer [AWS Lambda](#)-Funktion, der Aufruf von [Amazon EC2](#) Run Command, die Weiterleitung des Ereignisses an [Amazon Kinesis Data Streams](#) und [Automatisieren von Amazon EC2 mithilfe von EventBridge](#).

4. Standard-Betriebsabläufe (SOPs): Basierend auf den Komponenten Ihrer Anwendung empfiehlt [AWS Resilience Hub](#) mehrere [SOP-Vorlagen](#). Sie können diese SOPs verwenden, um alle Prozesse zu dokumentieren, die ein Bediener im Falle eines Alarms befolgen sollte. Sie können auch eine [SOP](#) auf Grundlage von Resilience Hub-Empfehlungen erstellen, für die Sie eine Resilience Hub-Anwendung mit einer zugehörigen Resilienzrichtlinie sowie eine historische Resilienzbewertung für diese Anwendung benötigen. Die Empfehlungen für Ihre SOP ergeben sich aus der Resilienzbewertung.

Resilience Hub arbeitet mit Systems Manager zusammen, um die einzelnen Schritte Ihrer SOPs zu automatisieren. Dazu erhalten Sie eine Reihe von [SSM-Dokumenten](#), die Sie als Grundlage für diese SOPs verwenden können. So kann Resilience Hub zum Beispiel eine SOP für das Hinzufügen von Speicherplatz auf Grundlage eines bestehenden SSM-Automatisierungsdokuments empfehlen.

5. Durchführen automatisierter Aktionen mit Amazon DevOps Guru: Sie können [Amazon DevOps Guru](#) verwenden, um Anwendungsressourcen automatisch auf anomales Verhalten zu überwachen und gezielte Empfehlungen für eine schnellere Problemerkennung und -behebung zu geben. Mit DevOps Guru können Sie Ströme von Betriebsdaten aus verschiedenen Quellen wie Amazon CloudWatch-Metriken, [AWS Config](#), [AWS CloudFormation](#) und [AWS X-Ray](#) nahezu in Echtzeit überwachen. Sie können DevOps Guru auch verwenden, um automatisch [OpsItems](#)

in OpsCenter zu erstellen und Ereignisse an [EventBridge](#) zu senden, um eine zusätzliche [Automatisierung zu erreichen](#).

Ressourcen

Zugehörige bewährte Methoden:

- [REL06-BP01 Überwachen aller Komponenten der Workload \(Generierung\)](#)
- [REL06-BP02 Definieren und Berechnen von Metriken \(Aggregierung\)](#)
- [REL06-BP03 Senden von Benachrichtigungen \(Verarbeitung und Benachrichtigung in Echtzeit\)](#)
- [REL08-BP01 Verwenden von Runbooks für Standardaktivitäten wie die Bereitstellung](#)

Zugehörige Dokumente:

- [AWS Systems Manager Automation](#)
- [Creating an EventBridge Rule That Triggers on an Event from an AWS Resource](#)
- [Workshop zur Beobachtbarkeit](#)
- [Die Amazon Builders' Library: Verteilte Systeme instrumentieren, um betriebliche Transparenz zu erzielen](#)
- [What is Amazon DevOps Guru?](#)
- [Working with Automation Documents \(Playbooks\)](#)

Zugehörige Videos:

- [AWS re:Invent 2022 - Observability best practices at Amazon](#)
- [AWS re:Invent 2020: Automate anything with AWS Systems Manager](#)
- [Introduction to AWS Resilience Hub](#)
- [Create Custom Ticket Systems for Amazon DevOps Guru Notifications](#)
- [Enable Multi-Account Insight Aggregation with Amazon DevOps Guru](#)

Zugehörige Beispiele:

- [Workshop zu Amazon CloudWatch und Systems Manager](#)

REL06-BP05 Analysieren von Protokollen

Erfassen Sie Protokolldateien und Metrikverläufe und analysieren Sie diese, um allgemeine Trends zu erkennen und Workload-Einblicke zu erhalten.

Amazon CloudWatch Logs Insights unterstützt eine [einfache, aber leistungsstarke Abfragesprache](#), mit der Sie Protokolldaten analysieren können. Amazon CloudWatch Logs unterstützt auch Abonnements, mit denen Daten nahtlos zu Amazon S3 fließen können, wo Sie sie nutzen oder Amazon Athena verwenden können, um die Daten abzufragen. Abfragen für eine große Auswahl von Formaten werden ebenfalls unterstützt. Unter [Supported SerDes and Data Formats](#) im Amazon Athena-Benutzerhandbuch finden Sie weitere Informationen dazu. Für die Analyse riesiger Protokolldateisätze können Sie einen Amazon EMR-Cluster ausführen, um Analysen im Petabyte-Bereich auszuführen.

Es gibt es eine Reihe von Werkzeugen von AWS-Partnern und externen Anbietern, die Aggregation, Verarbeitung, Speicherung und Analyse ermöglichen. Dazu gehören u. a. die Tools New Relic, Splunk, Loggly, Logstash, CloudHealth und Nagios. Die Generierung außerhalb von System- und Anwendungsprotokollen weicht jedoch bei jedem Cloud-Anbieter und häufig sogar bei den einzelnen Services ab.

Ein häufig übersehener Teil des Überwachungsprozesses ist die Datenverwaltung. Sie müssen Aufbewahrungsanforderungen für die Überwachung von Daten definieren und anschließend entsprechende Lebenszyklusrichtlinien anwenden. Amazon S3 unterstützt die Lebenszyklusverwaltung auf der Ebene von S3-Buckets. Diese Lebenszyklusverwaltung kann auf unterschiedliche Weise auf verschiedene Pfade im Bucket angewendet werden. Gegen Ende des Lebenszyklus können Sie die Daten zur Langzeitspeicherung an Amazon Glacier übertragen und die Speicherung nach Ablauf des Aufbewahrungszeitraums beenden. Die S3 Intelligent-Tiering-Speicherkategorie wurde entwickelt, um die Kosten zu optimieren. Daten werden automatisch in die kostengünstigste Zugriffsstufe verschoben, ohne Auswirkungen auf die Leistung oder höheren Betriebsaufwand.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

- Mit CloudWatch Logs Insights können Sie Protokolldaten in Amazon CloudWatch Logs interaktiv durchsuchen und analysieren.
 - [Analyzing Log Data with CloudWatch Logs Insights](#)
 - [Amazon CloudWatch Logs Insights Sample Queries](#)

- Senden Sie mit Amazon CloudWatch Logs Protokolle an Amazon S3. Dort können Sie die Daten mit Amazon Athena abfragen.
- [Wie verwende ich Amazon Athena, um meine Amazon-S3-Serverzugriffsprotokolle zu analysieren?](#)
 - Erstellen Sie eine S3-Lebenszyklusrichtlinie für Ihren Bucket mit den Serverzugriffsprotokollen. Konfigurieren Sie die Richtlinie so, dass Protokolldateien regelmäßig entfernt werden. Dadurch wird die Menge der Daten reduziert, die Athena in einer Abfrage analysiert.
 - [How Do I Create a Lifecycle Policy for an S3 Bucket?](#)

Ressourcen

Zugehörige Dokumente:

- [Amazon CloudWatch Logs Insights Sample Queries](#)
- [Analyzing Log Data with CloudWatch Logs Insights](#)
- [Debugging with Amazon CloudWatch Synthetics and AWS X-Ray](#)
- [How Do I Create a Lifecycle Policy for an S3 Bucket?](#)
- [Wie verwende ich Amazon Athena, um meine Amazon-S3-Serverzugriffsprotokolle zu analysieren?](#)
- [Workshop zur Beobachtbarkeit](#)
- [Die Amazon Builders' Library: Verteilte Systeme instrumentieren, um betriebliche Transparenz zu erzielen](#)

REL06-BP06 Regelmäßiges Durchführen von Prüfungen von Umfang und Metriken

Prüfen Sie häufig, wie die Workload-Überwachung implementiert ist, und aktualisieren Sie sie, wenn sich Ihre Workloads und ihre Architektur weiterentwickeln. Regelmäßige Prüfungen Ihrer Überwachung tragen dazu bei, das Risiko zu verringern, dass Fehlerindikatoren übersehen werden. Sie helfen außerdem Ihrem Workload, die Verfügbarkeitsziele zu erreichen.

Eine effektive Überwachung ist in wichtigen Geschäftsmetriken verankert, die entsprechend neuen Geschäftsprioritäten geändert werden. Ihr Monitoring-Überprüfungsprozess sollte den Schwerpunkt auf Service-Level-Indikatoren (SLIs) legen und Erkenntnisse aus Infrastruktur, Anwendungen, Clients und Benutzern einbeziehen.

Gewünschtes Ergebnis: Sie verfügen über eine effektive Überwachungsstrategie, die regelmäßig überprüft und in regelmäßigen Abständen sowie nach allen wichtigen Ereignissen oder Änderungen

aktualisiert wird. Sie stellen sicher, dass die wichtigsten Indikatoren für den Zustand Ihrer Anwendungen relevant bleiben, wenn sich Ihre Workloads und Ihre Geschäftsanforderungen weiterentwickeln.

Typische Anti-Muster:

- Sie erfassen nur Standardmetriken.
- Sie richten eine Überwachungsstrategie ein, überprüfen sie aber nie.
- Bei der Bereitstellung größerer Änderungen wird die Überwachung nicht berücksichtigt.
- Sie vertrauen veralteten Metriken, um den Zustand eines Workloads zu bestimmen.
- Ihre operativen Teams werden aufgrund veralteter Metriken und Schwellenwerte mit Fehlalarmen überlastet.
- Ihnen fehlt die Beobachtbarkeit von Anwendungskomponenten, die nicht überwacht werden.
- Sie konzentrieren sich bei der Überwachung nur auf technische Metriken auf untergeordneter Ebene und schließen geschäftliche Metriken aus.

Vorteile der Nutzung dieser bewährten Methode: Wenn Sie die Überwachung regelmäßig überprüfen, können Sie potenzielle Probleme antizipieren und sicherstellen, dass Sie diese erkennen. Außerdem können Sie so blinde Flecken aufdecken, die Sie bei früheren Überprüfungen möglicherweise übersehen haben, was Ihre Fähigkeit, Probleme zu erkennen, weiter verbessert.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Überprüfen Sie Metriken und Umfang der Überwachung im Rahmen Ihrer [Operational Readiness Review, ORR](#)). Führen Sie regelmäßige Prüfungen der operativen Bereitschaft nach einem konsistenten Zeitplan durch, um festzustellen, ob zwischen Ihren aktuellen Workloads und der von Ihnen konfigurierten Überwachung Lücken bestehen. Der Aufbau einer Struktur mit regelmäßigen Überprüfungen der operativen Leistung und einem Wissensaustausch verbessert Ihre Fähigkeit, höhere Leistungen bei Ihren operativen Teams zu erzielen. Prüfen Sie, ob die vorhandenen Schwellenwerte für Warnmeldungen immer noch ausreichend sind, und prüfen Sie, ob operative Teams falsch-positive Warnmeldungen erhalten oder Aspekte der Anwendung, die überwacht werden sollten, nicht überwachen.

Das [Framework für die Resilienzanalyse](#) bietet nützliche Hinweise, die Ihnen bei der Steuerung des Prozesses helfen können. Der Schwerpunkt des Frameworks liegt auf der Identifizierung potenzieller Arten von Ausfällen und der präventiven und korrigierenden Maßnahmen, mit denen Sie

ihre Auswirkungen abmildern können. Dieses Wissen kann Ihnen helfen, die richtigen Metriken und Ereignisse zu identifizieren, die Sie überwachen und bei denen Sie gewarnt werden sollten.

Implementierungsschritte

1. Planen und prüfen Sie die Workload-Dashboards regelmäßig. Was die Gründlichkeit der Untersuchungen angeht, sind unterschiedliche Intervalle denkbar.
2. Spüren Sie Trends in den Metriken auf. Vergleichen Sie die Metrikwerte mit Werten aus der Vergangenheit, um Trends zu erkennen, die darauf hinweisen könnten, dass etwas untersucht werden muss. Beispiele hierfür sind: zunehmende Latenz, Nachlassen der primären Geschäftsfunktion und zunehmende Anzahl von Reaktionen auf Fehler.
3. Suchen Sie in Ihren Metriken nach Ausreißern und Anomalien, die durch Durchschnitts- oder Medianwerte maskiert sein können. Sehen Sie sich die höchsten und niedrigsten Werte in einem bestimmten Zeitraum an und untersuchen Sie die Ursachen für Beobachtungen, die weit außerhalb der normalen Grenzen liegen. Beseitigen Sie nach und nach die Ursachen und legen Sie dabei einen immer engeren Maßstab für die erwarteten Metriken an, um auf die verbesserte Konsistenz der Workload-Leistung zu reagieren.
4. Spüren Sie plötzliche Änderungen im Verhalten auf. Eine plötzliche Veränderung in der Menge oder Richtung einer Metrik kann auf eine Änderung in der Anwendung hindeuten. Sie kann aber auch ein Hinweis auf externe Faktoren sein, für deren Verfolgung sie möglicherweise weitere Metriken hinzufügen müssen.
5. Prüfen Sie, ob die aktuelle Überwachungsstrategie für die Anwendung weiterhin relevant ist. Beurteilen Sie auf der Grundlage einer Analyse früherer Vorfälle (oder des Frameworks für die Resilienzanalyse), ob es weitere Aspekte der Anwendung gibt, die in den Überwachungsumfang aufgenommen werden sollten.
6. Überprüfen Sie Ihre RUM-Metriken (Real User Monitoring), um festzustellen, ob es Lücken bei der Abdeckung der Anwendungsfunktionen gibt.
7. Prüfen Sie Ihren Änderungsmanagementprozess. Aktualisieren Sie Ihre Verfahren bei Bedarf um einen Schritt zur Überwachung und Analyse, der durchgeführt werden sollte, bevor Sie eine Änderung genehmigen.
8. Implementieren Sie die Überprüfung der Überwachung als Teil Ihrer Prozesse zur Überprüfung der operativen Bereitschaft und zur Korrektur von Fehlern.

Ressourcen

Zugehörige bewährte Methoden:

- [REL06-BP01 Überwachen aller Komponenten des Workloads \(Generierung\)](#)
- [REL06-BP02 Definieren und Berechnen von Metriken \(Aggregation\)](#)
- [REL06-BP07 Überwachen der gesamten Nachverfolgung von Anfragen im System](#)
- [REL12-BP02 Durchführen von Analysen nach Vorfällen](#)
- [REL12-BP06 Regelmäßiges Durchführen von Gamedays](#)

Zugehörige Dokumente:

- [Warum sollten Sie eine Fehlerkorrektur \(Correction of Error, CoE\) entwickeln](#)
- [Amazon CloudWatch Dashboards verwenden](#)
- [Erstellen von Dashboards für operative Sichtbarkeit](#)
- [Erweiterte Multi-AZ-Resilienzmuster – Graue Ausfälle](#)
- [Amazon CloudWatch Logs Insights Sample Queries](#)
- [Debuggen mit Amazon CloudWatch Synthetics und AWS X-Ray](#)
- [Workshop zur Beobachtbarkeit](#)
- [Die Amazon Builders' Library: Verteilte Systeme instrumentieren, um betriebliche Transparenz zu erzielen](#)
- [Using Amazon CloudWatch Dashboards](#)
- [AWS Bewährte Methoden zur Beobachtbarkeit für](#)
- [Framework für Resilienzanalyse](#)
- [Framework für die Resilienzanalyse – Beobachtbarkeit](#)
- [Überprüfen der operativen Bereitschaft – ORR](#)

REL06-BP07 Überwachen der gesamten Nachverfolgung von Anfragen im System

Verfolgen Sie Anfragen während der Bearbeitung durch die Servicekomponenten, damit Produktteams Probleme einfacher analysieren und beheben und die Leistung verbessern können.

Gewünschtes Ergebnis: Workloads mit umfassender Nachverfolgung über alle Komponenten hinweg lassen sich leicht debuggen und verbessern so die [durchschnittliche Zeit für die Behebung](#) (MTTR) von Fehlern und Latenz durch eine vereinfachte Ursachenerkennung. Die durchgängige Nachverfolgung reduziert die Zeit, die benötigt wird, um betroffene Komponenten zu erkennen und die Ursachen von Fehlern oder Latenzen genau zu ermitteln.

Typische Anti-Muster:

- Nachverfolgung wird für einige Komponenten verwendet, aber nicht für alle. Ohne Nachverfolgung in AWS Lambda können Teams beispielsweise die durch Kaltstarts bei hohen Workloads verursachte Latenz nicht genau nachvollziehen.
- Synthetische Canaries oder Real-User Monitoring (RUM) sind nicht für Nachverfolgung konfiguriert. Ohne Canaries oder RUM wird die Telemetrie der Client-Interaktion in der Spurenanalyse ausgelassen, was zu einem unvollständigen Leistungsprofil führt.
- Hybride Workloads umfassen sowohl cloudnative Nachverfolgungs-Tools als auch Tools von Drittanbietern, es wurden jedoch keine Schritte unternommen, um eine einzige Nachverfolgungs-Lösung auszuwählen und vollständig zu integrieren. Basierend auf der gewählten Nachverfolgungs-Lösung sollten cloudnative Nachverfolgungs-SDKs verwendet werden, um Komponenten zu instrumentieren, die nicht cloudnativ sind. Oder Tools von Drittanbietern sollten so konfiguriert werden, dass sie cloudnative Nachverfolgungstelemetrie aufnehmen.

Vorteile der Nutzung dieser bewährten Methode: Wenn Entwicklungsteams über Probleme informiert werden, können sie sich ein vollständiges Bild der Interaktionen zwischen den Systemkomponenten machen, einschließlich der Beziehung zwischen Komponenten, Protokollierung, Leistung und Ausfällen. Da die Nachverfolgung die visuelle Identifizierung der Ursachen erleichtert, können diese schneller untersucht werden. Teams, die die Interaktionen der Komponenten im Detail verstehen, treffen bessere und schnellere Entscheidungen bei der Lösung von Problemen. Entscheidungen, z. B. wann ein Notfallwiederherstellung (DR)-Failover eingeleitet werden sollte oder wo Strategien zur Selbstreparatur am besten implementiert werden sollten, können durch die Analyse von Systemprotokollen verbessert werden, was letztlich die Kundenzufriedenheit mit Ihren Services erhöht.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Teams, die verteilte Anwendungen betreiben, können mithilfe von Nachverfolgungs-Tools eine Korrelationskennung einrichten, Spuren von Anfragen erfassen und Service-Maps für verbundene Komponenten erstellen. Alle Anwendungskomponenten sollten in den Anforderungsspuren enthalten sein, einschließlich Service-Clients, Middleware-Gateways und Event Busse, Rechenkomponenten und Speicher, einschließlich Schlüssel-Wert-Speicher und -Datenbanken. Integrieren Sie synthetische Canaries und Real-User Monitoring in Ihre Konfiguration für die gesamte

Nachverfolgung, um die Interaktionen und Latenz von Remote-Clients zu messen, sodass Sie die Leistung Ihres Systems anhand Ihrer Service Level Agreements und Ziele genau bewerten können.

Nutzen Sie Instrumentierungsservices wie [AWS X-Ray](#) und [Amazon CloudWatch-Anwendungsüberwachung](#), um einen vollständigen Überblick über die Anfragen zu erhalten, die in Ihrer Anwendung verarbeitet werden. X-Ray erfasst Anwendungstelemetrie und ermöglicht es Ihnen, diese nach Payloads, Funktionen, Spuren, Services und APIs zu visualisieren und zu filtern. Sie kann für Systemkomponenten aktiviert werden, bei denen kein Code oder Low-Code verwendet wird. Die CloudWatch-Anwendungsüberwachung umfasst ServiceLens, um Ihre Spuren in Metriken, Protokollen und Alarmen zu integrieren. Die CloudWatch-Anwendungsüberwachung umfasst auch synthetische Funktionen zur Überwachung Ihrer Endpunkte und APIs sowie Real-User Monitoring zur Instrumentierung Ihrer Webanwendungsclients.

Implementierungsschritte

- Verwenden Sie AWS X-Ray auf allen unterstützten nativen Services wie [Amazon S3](#), [AWS Lambda](#) und [Amazon API Gateway](#). Diese AWS-Services ermöglichen X-Ray mit Konfigurationsschaltern unter Verwendung von Infrastruktur als Code, AWS SDKs oder der AWS-Managementkonsole.
- Instrumenten Anwendungen [AWS Distro for Open Telemetry und X-Ray](#) oder Erfassungs-Agenten von Drittanbietern.
- Im [AWS X-Ray-Entwicklerhandbuch](#) finden Sie weitere Informationen für die programmiersprachenspezifische Implementierung. In diesen Dokumentationsabschnitten wird detailliert beschrieben, wie HTTP-Anfragen, SQL-Abfragen und andere Prozesse, die für Ihre Anwendungsprogrammiersprache spezifisch sind, instrumentiert werden.
- Verwenden Sie X-Ray-Nachverfolgung für [Amazon CloudWatch synthetische Canaries](#) und [Amazon CloudWatch RUM](#), um den Anforderungspfad von Ihrem Endbenutzer-Client durch Ihre AWS-Infrastruktur zu analysieren.
- Konfigurieren Sie CloudWatch-Metriken und -Alarme auf der Grundlage des Ressourcenzustands und der Canary-Telemetrie, sodass Teams schnell über Probleme informiert werden und dann mit ServiceLens Spuren und Servicemaps eingehend untersuchen können.
- Aktivieren Sie die X-Ray-Integration für Nachverfolgungs-Tools von Drittanbietern wie [Datadog](#), [New Relic](#) oder [Dynatrace](#), wenn Sie Tools von Drittanbietern als primäre Nachverfolgungslösung verwenden.

Ressourcen

Zugehörige bewährte Methoden:

- [REL06-BP01 Überwachen aller Komponenten der Workload \(Generierung\)](#)
- [REL11-BP01 Überwachen aller Komponenten der Workload auf Fehler](#)

Zugehörige Dokumente:

- [Was ist AWS X-Ray?](#)
- [Amazon CloudWatch-Anwendungsüberwachung](#)
- [Debugging with Amazon CloudWatch Synthetics and AWS X-Ray](#)
- [Die Amazon Builders' Library: Verteilte Systeme instrumentieren, um betriebliche Transparenz zu erzielen](#)
- [Integrating AWS X-Ray with other AWS services](#)
- [AWS Distro for OpenTelemetry und AWS X-Ray](#)
- [Amazon CloudWatch: Using synthetic monitoring](#)
- [Amazon CloudWatch: Use CloudWatch RUM](#)
- [Set up Amazon CloudWatch synthetics canary and Amazon CloudWatch alarm](#)
- [Verfügbarkeit und mehr: Verdeutlichung und Verbesserung der Ausfallsicherheit bei verteilten Systemen in AWS](#)

Zugehörige Beispiele:

- [Workshop zur Beobachtbarkeit](#)

Zugehörige Videos:

- [AWS re:Invent 2.022 - How to monitor applications across multiple accounts](#)
- [How to Monitor your AWS Applications](#)

Zugehörige Tools:

- [AWS X-Ray](#)
- [Amazon CloudWatch](#)

- [Amazon Route 53](#)

REL 7. Wie wird die Workload so gestaltet, dass sie sich an Veränderungen der Nachfrage anpasst?

Eine skalierbare Workload bietet Elastizität, sodass Ressourcen automatisch hinzugefügt oder entfernt werden können, damit sie dem aktuellen Bedarf zu einem bestimmten Zeitpunkt genau entsprechen.

Best Practices

- [REL07-BP01 Automatisiertes Abrufen oder Skalieren von Ressourcen](#)
- [REL07-BP02 Abrufen von Ressourcen bei Erkennen einer Beeinträchtigung einer Workload](#)
- [REL07-BP03 Abrufen von Ressourcen bei Feststellung, dass für eine Workload mehr Ressourcen benötigt werden](#)
- [REL07-BP04 Belastungstest Ihr Workload](#)

REL07-BP01 Automatisiertes Abrufen oder Skalieren von Ressourcen

Ein Eckpfeiler der Zuverlässigkeit in der Cloud ist die programmatische Definition, Bereitstellung und Verwaltung Ihrer Infrastruktur und Ihrer Ressourcen. Die Automatisierung hilft Ihnen dabei, die Ressourcenbereitstellung zu optimieren, konsistente und sichere Bereitstellungen zu ermöglichen und Ressourcen in Ihrer gesamten Infrastruktur zu skalieren.

Gewünschtes Ergebnis: Sie verwalten Ihre Infrastructure as Code (IaC). Sie definieren und verwalten Ihren Infrastrukturcode in Versionsverwaltungssystemen (VCS, Version Control Systems). Sie delegieren die Bereitstellung von AWS-Ressourcen an automatisierte Mechanismen und nutzen verwaltete Services wie Application Load Balancer (ALB), Network Load Balancer (NLB) und Auto-Scaling-Gruppen. Sie stellen Ihre Ressourcen mithilfe von CI/CD-Pipelines (Continuous Integration/Continuous Delivery) bereit, sodass Codeänderungen automatisch Ressourcenaktualisierungen einleiten, einschließlich Aktualisierungen Ihrer Auto-Scaling-Konfigurationen.

Typische Anti-Muster:

- Sie stellen Ressourcen manuell über die Befehlszeile oder in der AWS-Managementkonsole bereit (auch als ClickOps bezeichnet).
- Sie koppeln Ihre Anwendungskomponenten oder Ressourcen eng miteinander und schaffen dadurch unflexible Architekturen.

- Sie implementieren starre Skalierungsrichtlinien, die sich nicht an geänderte Geschäftsanforderungen, Datenverkehrsmuster oder neue Ressourcentypen anpassen.
- Sie schätzen die Kapazität manuell ein, um den erwarteten Bedarf zu decken.

Vorteile der Einführung dieser bewährten Methode: Mit Infrastructure as Code (IaC) kann Infrastruktur programmgesteuert definiert werden. Auf diese Weise können Sie Infrastrukturänderungen über denselben Softwareentwicklungszyklus verwalten wie Anwendungsänderungen. Dies fördert die Konsistenz und Wiederholbarkeit und verringert das Risiko manueller, fehleranfälliger Aufgaben. Sie können den Prozess der Bereitstellung und Aktualisierung von Ressourcen weiter optimieren, indem Sie IaC mit automatisierten Bereitstellungspipelines implementieren. Sie können Infrastrukturupdates zuverlässig und effizient bereitstellen, ohne dass manuelle Eingriffe erforderlich sind. Diese Flexibilität ist besonders wichtig, wenn Ressourcen skaliert werden, um schwankenden Anforderungen gerecht zu werden.

Sie erstellen aus Ihren Protokollen aggregierte Metriken, die das Verhalten Ihrer Workloads aus einer übergeordneten Perspektive darstellen. Durch die Überwachung wichtiger Metriken und die Anwendung vordefinierter Skalierungsrichtlinien kann Auto Scaling Ressourcen nach Bedarf automatisch bereitstellen oder die Bereitstellung aufheben, was die Leistung und Kosteneffizienz verbessert. Dadurch wird das Risiko manueller Fehler oder Verzögerungen bei der Reaktion auf Änderungen der Anwendungs- oder Workload-Anforderungen reduziert.

Die Kombination aus IaC, automatisierten Bereitstellungspipelines und Auto Scaling hilft Unternehmen dabei, ihre Umgebungen zuverlässig bereitzustellen, zu aktualisieren und zu skalieren. Diese Automatisierung ist für die Aufrechterhaltung einer reaktionsschnellen, belastbaren und effizient verwalteten Cloud-Infrastruktur unerlässlich.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Um die Automatisierung mit CI/CD-Pipelines und Infrastructure as Code (IaC) für Ihre AWS-Architektur einzurichten, wählen Sie ein Versionskontrollsystem wie Git, um Ihre IaC-Vorlagen und die Konfiguration zu speichern. Diese Vorlagen können mit Tools wie [AWS CloudFormation](#) geschrieben werden. Definieren Sie zunächst Ihre Infrastrukturkomponenten (z. B. AWSVPCs, Auto-Scaling-Gruppen in Amazon EC2 und Amazon-RDS-Datenbanken) innerhalb dieser Vorlagen.

Integrieren Sie dann diese IaC-Vorlagen in eine CI/CD-Pipeline, um den Bereitstellungsprozess zu automatisieren. [AWS CodePipeline](#) bietet eine nahtlose, AWS-native Lösung, Sie können aber auch andere CI/CD-Lösungen von Drittanbietern verwenden. Erstellen Sie eine Pipeline, die bei

Änderungen an Ihrem Versionsverwaltungs-Repository aktiviert wird. Konfigurieren Sie die Pipeline so, dass sie Stufen enthält, die Ihre IaC-Vorlagen erweitern und validieren, die Infrastruktur in einer Staging-Umgebung bereitstellen, automatisierte Tests ausführen und schließlich für die Produktion bereitstellen. Integrieren Sie bei Bedarf Genehmigungsschritte, um die Kontrolle über Änderungen zu behalten. Diese automatisierte Pipeline beschleunigt nicht nur die Bereitstellung, sondern sorgt auch für Konsistenz und Zuverlässigkeit in allen Umgebungen.

Konfigurieren Sie Auto Scaling für Ressourcen wie Amazon-EC2-Instances, Amazon-ECS-Aufgaben und Datenbankreplikate in Ihrer IaC, um bei Bedarf ein automatisches Aufskalieren und Abskalieren zu ermöglichen. Auf diese Weise verbessert sich die Verfügbarkeit und Leistung der Anwendungen und die Kosten werden optimiert, da die Ressourcen dynamisch an den Bedarf angepasst werden. Eine Liste der unterstützten Ressourcen finden Sie unter [Amazon EC2 Auto Scaling](#) und [AWS Auto Scaling](#).

Implementierungsschritte

1. Erstellen und verwenden Sie ein Quellcode-Repository, um den Code zu speichern, der Ihre Infrastrukturkonfiguration steuert. Übernehmen Sie die Änderungen in dieses Repository, um alle laufenden Änderungen widerzuspiegeln, die Sie vornehmen möchten.
2. Wählen Sie eine Infrastructure-as-Code-Lösung wie AWS CloudFormation, um Ihre Infrastruktur auf dem neuesten Stand zu halten und Abweichungen (Drift) gegenüber dem geplanten Zustand zu erkennen.
3. Integrieren Sie Ihre IaC-Plattform mit Ihrer CI/CD-Pipeline, um Bereitstellungen zu automatisieren.
4. Ermitteln und erfassen Sie die geeigneten Metriken für eine automatische Skalierung von Ressourcen.
5. Konfigurieren Sie die automatische Skalierung von Ressourcen mithilfe von Richtlinien für das Auf- und Abskalieren, die für Ihre Workload-Komponenten geeignet sind. Erwägen Sie die Verwendung einer geplanten Skalierung bei vorhersehbaren Nutzungsmustern.
6. Überwachen Sie Bereitstellungen, um Fehler und Regressionen zu erkennen. Implementieren Sie Rollback-Mechanismen innerhalb Ihrer CI/CD-Plattform, um Änderungen bei Bedarf rückgängig zu machen.

Ressourcen

Zugehörige Dokumente:

- [AWS Auto Scaling: Funktionsweise von Skalierungsplänen](#)

- [AWS Marketplace: Für Auto Scaling geeignete Produkte](#)
- [Automatische Verwaltung der Durchsatzkapazität mit DynamoDB-Auto-Scaling](#)
- [Nutzen eines Load Balancer mit einer Auto-Scaling-Gruppe](#)
- [Was ist AWS Global Accelerator?](#)
- [Was ist Amazon EC2 Auto Scaling?](#)
- [Was ist AWS Auto Scaling?](#)
- [Was ist Amazon CloudFront?](#)
- [Was ist Amazon Route 53?](#)
- [Was ist Elastic Load Balancing?](#)
- [Was ist ein Network Load Balancer?](#)
- [Was ist ein Application Load Balancer?](#)
- [Integrieren von Jenkins mit AWS CodeBuild und AWS CodeDeploy](#)
- [Erstellen einer vierstufigen Pipeline mit AWS CodePipeline](#)

Zugehörige Videos:

- [Zurück zu den Grundlagen: Stellen Sie Ihren Code in Amazon EC2 bereit](#)
- [AWS bietet Ihnen Unterstützung | Infrastructure-as-Code-Lösung mithilfe von AWS CloudFormation-Vorlagen starten](#)
- [Optimieren des Prozesses zur Software-Veröffentlichung mit AWS CodePipeline](#)
- [Überwachen von AWS-Ressourcen mithilfe von Amazon-CloudWatch-Dashboards](#)
- [Erstellen von konto- und regionsübergreifenden CloudWatch-Dashboards | Amazon Web Services](#)

REL07-BP02 Abrufen von Ressourcen bei Erkennen einer Beeinträchtigung einer Workload

Skalieren Sie Ressourcen bei Bedarf reaktiv, wenn die Verfügbarkeit beeinträchtigt ist, um die Verfügbarkeit der Workload wiederherzustellen.

Sie müssen zunächst Zustandsprüfungen und die Kriterien für diese Prüfungen konfigurieren, um anzugeben, wann die Verfügbarkeit durch fehlende Ressourcen beeinträchtigt wird. Benachrichtigen Sie anschließend entweder die zuständigen Mitarbeiter, um die Ressource manuell zu skalieren, oder starten Sie die Automatisierung, um sie automatisch zu skalieren.

Die Skalierung kann manuell an Ihre Workload angepasst werden, z. B. indem Sie die Anzahl der EC2-Instances in einer Auto Scaling-Gruppe ändern oder den Durchsatz einer DynamoDB-Tabelle über die AWS-Managementkonsole oder AWS CLI. Wann immer es möglich ist, sollte jedoch Automatisierung eingesetzt werden (siehe Automatisiertes Abrufen oder Skalieren von Ressourcen).

Gewünschtes Ergebnis: Skalierungsaktivitäten (entweder automatisch oder manuell) werden eingeleitet, um die Verfügbarkeit wiederherzustellen, sobald ein Ausfall oder eine Verschlechterung der Kundenerfahrung festgestellt wird.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Implementieren Sie Beobachtbarkeit und Überwachung für alle Komponenten Ihres Workloads, um die Kundenerfahrung zu überwachen und Fehler zu erkennen. Definieren Sie die manuellen oder automatischen Verfahren zur Skalierung der erforderlichen Ressourcen. Weitere Informationen finden Sie unter [REL11-BP01 Überwachen aller Komponenten der Workload auf Fehler](#).

Implementierungsschritte

- Definieren Sie die manuellen oder automatisierten Verfahren, mit denen die erforderlichen Ressourcen skaliert werden.
 - Die Skalierungsverfahren hängen davon ab, wie die verschiedenen Komponenten innerhalb Ihres Workloads gestaltet sind.
 - Die Skalierungsverfahren variieren auch je nach der zugrunde liegenden Technologie, die verwendet wird.
 - Komponenten, die AWS Auto Scaling verwenden, können Skalierungspläne nutzen, um eine Reihe von Anweisungen für die Skalierung Ihrer Ressourcen zu konfigurieren. Wenn Sie mit AWS CloudFormation arbeiten oder AWS-Ressourcen Tags hinzufügen, können Sie pro Anwendung Skalierungspläne für verschiedene Ressourcengruppen einrichten. Auto Scaling bietet Empfehlungen für Skalierungsstrategien, die auf die einzelnen Ressourcen zugeschnitten sind. Nachdem Sie einen Skalierungsplan erstellt haben, kombiniert Auto Scaling zur Unterstützung Ihrer Skalierungsstrategie Methoden für die dynamische und prädiktive Skalierung. Weitere Informationen finden Sie unter [Funktionsweise von Skalierungsplänen](#).
- Mit Amazon EC2 Auto Scaling können Sie sicherstellen, dass Ihnen die richtige Anzahl von Amazon EC2-Instances zur Verfügung steht, um die Anwendungslast zu bewältigen. Sie erstellen Sammlungen von EC2 Instances, die als Auto Scaling-Gruppen bezeichnet

werden. In jeder Auto Scaling-Gruppe können Sie die Mindestanzahl von Instances angeben. Amazon EC2 Auto Scaling stellt dann sicher, dass die Gruppe diese Größe nie unter- oder überschreitet. Weitere Informationen finden Sie unter [Was ist Amazon EC2 Auto Scaling?](#)

- Amazon DynamoDB-Auto-Scaling verwendet den -Application-Auto-Scaling-Service, um die bereitgestellte Durchsatzkapazität in Ihrem Namen als Reaktion auf tatsächliche Datenverkehrsmuster dynamisch anzupassen. Auf diese Weise kann eine Tabelle oder ein globaler sekundärer Index die bereitgestellte Lese- und Schreibkapazität zum Verarbeiten eines plötzlichen Datenverkehrsanstiegs ohne Drosselung erhöhen. Weitere Informationen finden Sie unter [Automatische Verwaltung der Durchsatzkapazität mit DynamoDB-Auto-Scaling](#).

Ressourcen

Zugehörige bewährte Methoden:

- [REL07-BP01 Automatisiertes Abrufen oder Skalieren von Ressourcen](#)
- [REL11-BP01 Überwachen aller Komponenten der Workload auf Fehler](#)

Zugehörige Dokumente:

- [AWS Auto Scaling: Funktionsweise von Skalierungsplänen](#)
- [Automatische Verwaltung der Durchsatzkapazität mit DynamoDB-Auto-Scaling](#)
- [Was ist Amazon EC2 Auto Scaling?](#)

REL07-BP03 Abrufen von Ressourcen bei Feststellung, dass für eine Workload mehr Ressourcen benötigt werden

Eine der wertvollsten Funktionen von Cloud Computing ist die Fähigkeit, Ressourcen dynamisch bereitzustellen.

In herkömmlichen On-Premises-Computerumgebungen müssen Sie im Voraus genügend Kapazität identifizieren und bereitstellen, um Zeiten mit höchster Auslastung bewältigen zu können. Dies ist ein Problem, da es teuer ist und die Verfügbarkeit gefährdet sein kann, wenn Sie die Spitzenkapazitätsanforderungen des Workloads unterschätzen.

In der Cloud ist dies nicht notwendig. Hier können Sie die Rechen-, Datenbank- und sonstigen Ressourcenkapazitäten nach Bedarf bereitstellen, um den aktuellen und prognostizierten Bedarf

zu decken. Automatisierte Lösungen wie Amazon EC2 Auto Scaling und Application Auto Scaling können Ressourcen basierend auf von Ihnen angegebenen Metriken online für Sie bereitstellen. Dies kann den Skalierungsprozess einfacher und vorhersehbarer machen. Ihre Workloads sind jetzt erheblich zuverlässiger, da sichergestellt wird, dass Ihnen jederzeit genügend Ressourcen zur Verfügung stehen.

Gewünschtes Ergebnis: Sie konfigurieren die automatische Skalierung von Rechenleistung und anderen Ressourcen, um den Bedarf zu decken. Sie sorgen in Ihren Skalierungsrichtlinien für ausreichend Spielraum, damit Datenverkehrsspitzen aufgefangen werden können, während zusätzliche Ressourcen online geschaltet werden.

Typische Anti-Muster:

- Sie stellen eine feste Anzahl von skalierbaren Ressourcen bereit.
- Sie wählen eine Skalierungsmetrik, die nicht dem tatsächlichen Bedarf entspricht.
- Sie bieten in Ihren Skalierungsplänen nicht genügend Spielraum, um Nachfragespitzen Rechnung zu tragen.
- Ihre Skalierungsrichtlinien erhöhen die Kapazität zu spät, dadurch kommt es zu einer Erschöpfung der Kapazitäten und zu einem schlechten Service, während zusätzliche Ressourcen online geschaltet werden.
- Sie konfigurieren die minimale und maximale Anzahl an Ressourcen nicht korrekt, was zu Skalierungsfehlern führt.

Vorteile der Einführung dieser bewährten Methode: Ausreichend Ressourcen zur Deckung des aktuellen Bedarfs sind entscheidend, um eine hohe Verfügbarkeit Ihres Workloads zu gewährleisten und Ihre definierten Servicelevel-Ziele (SLOs) einzuhalten. Durch die automatische Skalierung können Sie die richtige Menge an Rechen-, Datenbank- und sonstigen Ressourcen bereitstellen, die Ihr Workload benötigt, um den aktuellen und prognostizierten Bedarf zu decken. Sie müssen den Spitzenkapazitätsbedarf nicht ermitteln und Ressourcen nicht statisch zuweisen, um dem Bedarf gerecht zu werden. Sie haben vielmehr die Möglichkeit, bei steigendem Bedarf mehr Ressourcen zuzuweisen und bei einem Rückgang des Bedarfs Ressourcen zu deaktivieren, um die Kosten zu senken.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Stellen Sie zunächst fest, ob die Workload-Komponente für die automatische Skalierung geeignet ist. Diese Komponenten werden als horizontal skalierbar bezeichnet, da sie dieselben Ressourcen

bereitstellen und sich identisch verhalten. Beispiele für horizontal skalierbare Komponenten sind EC2-Instances, die gleich konfiguriert sind, Aufgaben von [Amazon Elastic Container Service \(ECS\)](#) und Pods, die auf [Amazon Elastic Kubernetes Service \(EKS\)](#) ausgeführt werden. Diese Rechenressourcen befinden sich in der Regel hinter einem Load Balancer und werden als Replikate bezeichnet.

Weitere replizierte Ressourcen können Datenbanklesereplikate, [Amazon-DynamoDB-Tabellen](#) und [Amazon ElastiCache](#) (Redis OSS)-Cluster sein. Eine vollständige Liste der unterstützten Ressourcen finden Sie unter [AWS-Services, die Sie mit Application Auto Scaling verwenden können](#).

Bei containerbasierten Architekturen müssen Sie möglicherweise auf zwei verschiedene Arten skalieren. Zunächst müssen Sie möglicherweise die Container skalieren, die horizontal skalierbare Services bereitstellen. Zweitens müssen Sie möglicherweise die Rechenressourcen skalieren, um Platz für neue Container zu schaffen. Für jede Ebene gibt es unterschiedliche automatische Skalierungsmechanismen. Um ECS-Aufgaben zu skalieren, können Sie [Application Auto Scaling](#) verwenden. Um Kubernetes-Pods zu skalieren, können Sie [Horizontal Pod Autoscaler \(HPA\)](#) oder [Kubernetes Event-Driven Autoscaling \(KEDA\)](#) verwenden. Zum Skalieren der Rechenressourcen können Sie [Kapazitätsanbieter](#) für ECS oder, für Kubernetes, [Karpenter](#) oder [Cluster Autoscaler](#) verwenden.

Wählen Sie als Nächstes aus, wie Sie die automatische Skalierung durchführen möchten. Es gibt drei Hauptoptionen: metrikbasierte Skalierung, geplante Skalierung und prädiktive Skalierung.

Metrikbasierte Skalierung

Bei der metrikbasierten Skalierung werden Ressourcen auf der Grundlage des Werts einer oder mehrerer Skalierungsmetriken bereitgestellt. Eine Skalierungsmetrik entspricht dem Bedarf Ihres Workloads. Eine gute Methode, um geeignete Skalierungsmetriken zu ermitteln, besteht darin, Lasttests in einer Nicht-Produktionsumgebung durchzuführen. Halten Sie während der Lasttests die Anzahl skalierbarer Ressourcen konstant und erhöhen Sie die Nachfrage langsam (z. B. Durchsatz, Parallelität oder simulierte Benutzer). Suchen Sie dann nach Metriken, die sich bei steigender Nachfrage erhöhen (oder verringern) und sich umgekehrt bei sinkender Nachfrage verringern (oder erhöhen). Zu den typischen Skalierungsmetriken gehören die CPU-Auslastung, die Tiefe der Arbeitswarteschlange (z. B. einer [Amazon SQS](#)-Warteschlange), die Anzahl der aktiven Benutzer und der Netzwerkdurchsatz.

 Note

AWS hat beobachtet, dass bei den meisten Anwendungen die Speicherauslastung in der Aufwärmphase der Anwendung zunimmt und dann einen konstanten Wert erreicht. Wenn die Nachfrage sinkt, bleibt die Speicherauslastung in der Regel erhöht und geht nicht parallel zurück. Da die Speicherauslastung in beiden Richtungen nicht dem Bedarf entspricht, also nicht mit dem Bedarf steigt und fällt, sollten Sie es sich genau überlegen, bevor Sie diese Metrik für die automatische Skalierung auswählen.

Die metrikbasierte Skalierung ist eine latente Operation. Es kann mehrere Minuten dauern, bis die Nutzungsmetriken auf Auto-Scaling-Mechanismen übertragen werden. Diese Mechanismen warten in der Regel auf ein deutliches Signal für eine erhöhte Nachfrage, bevor sie reagieren. Wenn der Autoscaler dann neue Ressourcen erstellt, kann es zusätzliche Zeit dauern, bis diese voll funktionsfähig sind. Aus diesem Grund ist es wichtig, dass Sie Ihre Skalierungsmetrikziele nicht zu nahe an der Vollausslastung festlegen (z. B. 90 % CPU-Auslastung). Ansonsten besteht die Gefahr, dass die vorhandene Ressourcenkapazität ausgeschöpft wird, bevor zusätzliche Kapazität verfügbar ist. Um eine optimale Verfügbarkeit zu erreichen, können die Ziele für die Ressourcennutzung typischerweise zwischen 50 und 70 % liegen, je nach Bedarfsmuster und Zeitaufwand für die Bereitstellung zusätzlicher Ressourcen.

Geplante Skalierung

Bei der geplanten Skalierung werden Ressourcen basierend auf dem Kalender oder der Tageszeit bereitgestellt oder entfernt. Sie wird häufig für Workloads mit vorhersehbarem Bedarf verwendet, wenn es beispielsweise zu den Geschäftszeiten an Wochentagen oder bei Verkaufsveranstaltungen zu einer Spitzenauslastung kommt. Sowohl [Amazon EC2 Auto Scaling](#) als auch [Application Auto Scaling](#) unterstützen die geplante Skalierung. Der [Cron Scaler](#) von KEDA unterstützt die geplante Skalierung von Kubernetes-Pods.

Prädiktive Skalierung

Die prädiktive Skalierung verwendet Machine Learning, um Ressourcen automatisch basierend auf der erwarteten Nachfrage zu skalieren. Bei der prädiktiven Skalierung wird der historische Wert einer von Ihnen angegebenen Nutzungsmetrik analysiert und ihr zukünftiger Wert kontinuierlich prognostiziert. Der prognostizierte Wert wird dann verwendet, um die Ressource nach oben oder unten zu skalieren. [Amazon EC2 Auto Scaling](#) kann eine prädiktive Skalierung durchführen.

Implementierungsschritte

1. Ermitteln Sie, ob die Workload-Komponente für die automatische Skalierung geeignet ist.
2. Finden Sie heraus, welche Art von Skalierungsmechanismus für den Workload am besten geeignet ist: metrikbasierte Skalierung, geplante Skalierung oder prädiktive Skalierung.
3. Wählen Sie den geeigneten automatischen Skalierungsmechanismus für die Komponente aus. Verwenden Sie Amazon EC2 Auto Scaling für Amazon-EC2-Instances. Verwenden Sie Application Auto Scaling für andere AWS-Services. Für Kubernetes-Pods (wie z. B. solche, die in einem Amazon-EKS-Cluster ausgeführt werden) können Sie Horizontal Pod Autoscaler (HPA) oder Kubernetes Event-Driven Autoscaling (KEDA) in Betracht ziehen. Für Kubernetes- oder EKS-Knoten sollten Sie Karpenter und Cluster Auto Scaler (CAS) in Betracht ziehen.
4. Führen Sie für die metrikbasierte oder geplante Skalierung Lasttests durch, um die geeigneten Skalierungsmetriken und Zielwerte für Ihren Workload zu ermitteln. Ermitteln Sie für die geplante Skalierung die benötigte Ressourcenanzahl zu den von Ihnen ausgewählten Daten und Uhrzeiten. Ermitteln Sie die maximale Anzahl an Ressourcen, die benötigt werden, um die erwartete Verkehrsspitze zu bewältigen.
5. Konfigurieren Sie den Autoscaler auf der Grundlage der oben gesammelten Informationen. Weitergehende Informationen finden Sie in der Dokumentation zum Auto-Scaling-Service. Stellen Sie sicher, dass die Ober- und Untergrenze für die Skalierung korrekt konfiguriert sind.
6. Vergewissern Sie sich, dass die Skalierungskonfiguration wie erwartet funktioniert. Führen Sie Lasttests in einer Nicht-Produktionsumgebung durch, beobachten Sie, wie das System reagiert und nehmen Sie gegebenenfalls Anpassungen vor. Wenn Sie Auto Scaling in der Produktion aktivieren, konfigurieren Sie entsprechende Alarme, damit Sie über unerwartetes Verhalten informiert werden.

Ressourcen

Zugehörige Dokumente:

- [Was ist Amazon EC2 Auto Scaling?](#)
- [AWS – Präskriptive Leitlinien: Anwendungen für Lasttests](#)
- [AWS Marketplace: Für Auto Scaling geeignete Produkte](#)
- [Automatische Verwaltung der Durchsatzkapazität mit DynamoDB-Auto-Scaling](#)
- [Predictive Scaling for EC2, Powered by Machine Learning](#)
- [Scheduled Scaling for Amazon EC2 Auto Scaling](#)

- [Berichte über das Gesetz von Little](#)

REL07-BP04 Belastungstest Ihr Workload

Messen Sie anhand von Lasttests, ob die Skalierung den Workload-Anforderungen gerecht wird.

Es ist wichtig, regelmäßige Lasttests durchzuführen. Auslastungstests sollten die Belastungsgrenze ermitteln und die Leistung Ihres Workloads testen. AWS macht es einfach, temporäre Testumgebungen einzurichten, die den Umfang Ihrer Produktionslast modellieren. Sie können in der Cloud bei Bedarf eine Testumgebung in Produktionsgröße einrichten, Ihre Tests abschließen und die Ressourcen dann wieder stilllegen. Weil Sie für die Testumgebung nur dann zahlen, wenn sie genutzt wird, können Sie Ihre Live-Umgebung zu einem Bruchteil der Kosten testen, die Sie an einem On-Premises-Standort hätten.

Lasttests in der Produktion sollten auch im Rahmen von Ernstfallübungen durchgeführt werden, bei denen das Produktionssystem in einem Zeitraum mit geringer Kundennutzung stark belastet wird. Alle Mitarbeiter sollten an dieser Übung beteiligt sein, die Ergebnisse gemeinsam interpretieren und auftretende Probleme beheben.

Typische Anti-Muster:

- Es werden Lasttests für Bereitstellungen durchgeführt, die nicht mit der Konfiguration der Produktionsumgebung übereinstimmen.
- Lasttests werden nur für einzelne Teile, nicht aber für den gesamten Workload durchgeführt.
- Es werden Lasttests mit einer Teilmenge von Anfragen durchgeführt, aber nicht mit einer repräsentativen Gruppe tatsächlicher Anfragen.
- Es werden Lasttests mit einem kleinen Sicherheitsfaktor durchgeführt, der über der erwarteten Last liegt.

Vorteile der Nutzung dieser bewährten Methode: Sie wissen, welche Komponenten in der Architektur unter Last ausfallen, und können die zu überwachenden Metriken festlegen, die rechtzeitig auf die Annäherung an die Belastungsgrenze hinweisen, damit Sie das Problem beheben und entsprechende Auswirkungen vermeiden können.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

- Bestimmen Sie anhand von Lasttests, welcher Aspekt der Workload angeben soll, dass Kapazität hinzugefügt oder entfernt werden muss. Bei Lasttests sollte ein repräsentativer Datenverkehr zum Einsatz kommen, der dem in der Produktion ähnelt. Erhöhen Sie unter Beobachtung der instrumentierten Metriken die Last, um zu bestimmen, welche Metrik angibt, wann Ressourcen hinzugefügt oder entfernt werden müssen.
- [Distributed Load Testing auf AWS: Simulieren Sie Tausende verbundener Benutzer](#)
 - Ermitteln Sie die Zusammensetzung von Anfragen. Möglicherweise haben Sie unterschiedliche Zusammensetzungen von Anfragen. Daher sollten Sie sich bei der Ermittlung der Zusammensetzung des Datenverkehrs verschiedene Zeiträume ansehen.
 - Implementieren Sie einen Lasttreiber. Zum Implementieren eines Lasttreibers können Sie Software mit eigenem Code, Open-Source-Software oder kommerzielle Software verwenden.
 - Führen Sie Lasttests zunächst mit geringer Kapazität durch. Schon bei der Erhöhung der Last für eine Einheit mit geringerer Kapazität, etwa einer einzelnen Instance oder einem einzelnen Container, stellen Sie unmittelbare Auswirkungen fest.
 - Führen Sie Lasttests mit größerer Kapazität durch. Bei einer verteilten Last sehen die Auswirkungen anders aus. Daher müssen Sie bei Tests Bedingungen herstellen, die der Produktionsumgebung möglichst nahekommen.

Ressourcen

Zugehörige Dokumente:

- [Distributed Load Testing aktiviert AWS: simulieren Sie Tausende verbundener Benutzer](#)
- [Anwendungen für Lasttests](#)

Zugehörige Videos:

- [AWS Summit ANZ 2023: Mit AWS Distributed Load Testing können Sie mit Zuversicht schneller vorankommen](#)

REL 8. Wie werden Veränderungen implementiert?

Kontrollierte Änderungen sind erforderlich, um neue Funktionen bereitzustellen und um sicherzustellen, dass die Workloads und die Betriebsumgebung bekannte Software ausführen und

auf vorhersagbare Weise durch Patches aktualisiert oder ersetzt werden können. Wenn solche Änderungen nicht kontrolliert sind, ist es schwierig, die Auswirkungen der Änderungen vorherzusagen oder Probleme zu lösen, die sich aus ihnen ergeben.

Best Practices

- [REL08-BP01 Verwenden von Runbooks für Standardaktivitäten wie die Bereitstellung](#)
- [REL08-BP02 Integrieren von Funktionstests in die Bereitstellung](#)
- [REL08-BP03 Integrieren von Ausfallsicherheitstests in die Bereitstellung](#)
- [REL08-BP04 Bereitstellung mit einer unveränderlichen Infrastruktur](#)
- [REL08-BP05 Automatisieren von Änderungen](#)

REL08-BP01 Verwenden von Runbooks für Standardaktivitäten wie die Bereitstellung

Runbooks sind vordefinierte Verfahren, die ein bestimmtes Ergebnis verfolgen. Verwenden Sie Runbooks, um Standardaktivitäten manuell oder automatisch durchzuführen. Beispiele für solche Aktivitäten sind etwa die Bereitstellung und das Patchen einer Workload oder das Vornehmen von DNS-Änderungen.

Sie können z. B. Prozesse einrichten, [um bei Bereitstellungen die Rollback-Sicherheit zu gewährleisten](#). Wenn Sie eine Bereitstellung ohne Unterbrechung für Ihre Kunden zurücksetzen können, steigert das die Zuverlässigkeit Ihres Service.

Für Runbook-Verfahren sollten Sie mit einem gültigen, effektiven manuellen Prozess beginnen, diesen in Code implementieren und ggf. die automatische Ausführung auslösen.

Selbst bei anspruchsvollen Workloads mit umfassender Automatisierung sind Runbooks nützlich, um [Ernstfallübungen auszuführen](#) oder strenge Berichterstellungs- und Auditing-Anforderungen zu erfüllen.

Playbooks werden als Reaktion auf bestimmte Vorfälle verwendet und mit Runbooks sollen bestimmte Ergebnisse erzielt werden. Häufig werden Runbooks für Routineaktivitäten genutzt, während Playbooks für die Reaktion auf außerplanmäßige Ereignisse verwendet werden.

Typische Anti-Muster:

- Durchführen ungeplanter Änderungen an der Konfiguration in der Produktion.
- Überspringen von Schritten in Ihrem Plan, um eine schnellere Bereitstellung durchzuführen, was zu einer fehlgeschlagenen Bereitstellung führt.

- Vornehmen von Änderungen, ohne die Umkehrung der Änderung zu testen.

Vorteile der Nutzung dieser bewährten Methode: Die effektive Änderungsplanung erhöht Ihre Fähigkeit, die Änderung erfolgreich auszuführen, da Sie sich aller betroffenen Systeme bewusst sind. Die Validierung Ihrer Änderungen in Testumgebungen erhöht Ihre Sicherheit.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

- Unterstützen Sie konsistente und schnelle Reaktionen auf gut bekannte Ereignisse, indem Sie Verfahren in Runbooks dokumentieren.
- Verwenden Sie zur Definition Ihrer Infrastruktur den Grundsatz „Infrastructure as Code“. Wenn Sie Ihre Infrastruktur mit AWS CloudFormation oder dem vertrauenswürdigen Tool eines Drittanbieters definieren, können Sie Änderungen mithilfe einer Versionskontrollsoftware versionieren und nachverfolgen.
 - Nutzen Sie zur Definition Ihrer Infrastruktur AWS CloudFormation (oder das vertrauenswürdige Tool eines Drittanbieters).
 - [Was ist AWS CloudFormation?](#)
- Erstellen Sie unter Anwendung guter Grundsätze für das Softwaredesign Vorlagen, die getrennt und entkoppelt sind.
 - Ermitteln Sie die für die Implementierung erforderlichen Berechtigungen, Vorlagen und zuständigen Parteien.
 - [Zugriffssteuerung mit AWS Identity and Access Management](#)
 - Verwenden Sie ein gehostetes Quellcodeverwaltungssystem, das auf einer verbreiteten Technologie wie Git basiert, um Ihren Quellcode und Ihre Infrastruktur-as-Code (IaC)-Konfiguration zu speichern.

Ressourcen

Zugehörige Dokumente:

- [APN-Partner: Partner, die Sie beim Erstellen automatisierter Bereitstellungslösungen unterstützen können](#)
- [AWS Marketplace: Produkte zur Automatisierung Ihrer Bereitstellungen](#)
- [Was ist AWS CloudFormation?](#)

Zugehörige Beispiele:

- [Automating operations with Playbooks and Runbooks](#)

REL08-BP02 Integrieren von Funktionstests in die Bereitstellung

Verwenden Sie Techniken wie Modultests und Integrationstests, die die erforderliche Funktionalität validieren.

Bei Modultests testen Sie die kleinste funktionale Codeeinheit, um ihr Verhalten zu überprüfen. Mit Integrationstests soll überprüft werden, ob jede Anwendungsfunktion den Softwareanforderungen entspricht. Während sich Modultests auf die isolierte Prüfung eines Teils einer Anwendung konzentrieren, berücksichtigen Integrationstests auch Nebenwirkungen (z. B. den Effekt, dass Daten durch eine Mutationsoperation verändert werden). In beiden Fällen sollten die Tests in eine Bereitstellungs-Pipeline integriert werden. Wenn die Erfolgskriterien nicht erfüllt sind, wird die Pipeline angehalten oder es wird ein Rollback durchgeführt. Diese Tests werden in einer Vorproduktionsumgebung ausgeführt, die vor der Produktion in der Pipeline bereitgestellt wird.

Im Idealfall sollten diese Tests automatisch als Teil von Build- und Bereitstellungsaktionen ausgeführt werden. Mit AWS CodePipeline übertragen Entwickler beispielsweise Änderungen an ein Quell-Repository, in dem CodePipeline die Änderungen automatisch erkennt. Die Anwendung wird erstellt und es werden Modultests ausgeführt. Nachdem die Modultests bestanden wurden, wird der erstellte Code für weitere Tests auf Staging-Servern bereitgestellt. Auf dem Staging-Server führt CodePipeline weitere Tests aus, z. B. Integrations- oder Belastungstests. Nach dem erfolgreichen Abschluss dieser Tests stellt CodePipeline den getesteten und genehmigten Code für Produktions-Instances bereit.

Gewünschtes Ergebnis: Sie nutzen die Automatisierung zur Durchführung von Modul- und Integrationstests, um zu überprüfen, ob sich Ihr Code erwartungsgemäß verhält. Diese Tests sind in den Bereitstellungsprozess integriert, bei einem fehlgeschlagenen Test wird die Bereitstellung abgebrochen.

Typische Anti-Muster:

- Sie ignorieren oder umgehen Testfehler und Testpläne während des Bereitstellungsprozesses, um die Bereitstellung zu beschleunigen.
- Sie führen Tests außerhalb der Bereitstellungs-pipeline manuell durch.
- Sie überspringen Testschritte in Ihrer Automatisierung durch manuelle Notfall-Workflows.
- Sie führen automatisierte Tests in einer Umgebung durch, die der Produktionsumgebung nicht sehr ähnlich ist.

- Sie erstellen eine Testsuite, die nicht flexibel genug ist und sich nur schwer warten, aktualisieren oder skalieren lässt, wenn sich die Anwendung weiterentwickelt.

Vorteile der Einführung dieser bewährten Methode: Durch automatisierte Tests während des Bereitstellungsprozesses werden Probleme frühzeitig erkannt. Dadurch verringert sich das Risiko einer Produktionsfreigabe mit Fehlern oder unerwartetem Verhalten. Modultests überprüfen, ob sich der Code wie gewünscht verhält und API-Verträge eingehalten werden. Integrationstests bestätigen, dass das System den angegebenen Anforderungen entsprechend funktioniert. Diese Arten von Tests werden verwendet, um die beabsichtigte Funktionsreihenfolge von Komponenten wie Benutzeroberflächen, APIs, Datenbanken und Quellcode zu überprüfen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Führen Sie einen Ansatz für eine testgesteuerte Entwicklung (TDD, Test-driven Development) für das Schreiben von Software ein, bei dem Sie Testfälle entwickeln, um Ihren Code zu spezifizieren und zu validieren. Erstellen Sie zunächst Testfälle für jede Funktion. Wenn der Test fehlschlägt, schreiben Sie neuen Code, um den Test zu bestehen. Dieser Ansatz hilft Ihnen, das erwartete Ergebnis jeder Funktion zu überprüfen. Führen Sie Modultests aus und überprüfen Sie, ob diese bestanden werden, bevor Sie Code in ein Quellcode-Repository übertragen.

Implementieren Sie sowohl Modul- als auch Integrationstests im Rahmen der Erstellungs-, Test- und Bereitstellungsphasen der CI/CD-Pipeline. Automatisieren Sie die Tests und leiten Sie automatisch Tests ein, wenn eine neue Version der Anwendung bereit für die Bereitstellung ist. Wenn die Erfolgskriterien nicht erfüllt sind, wird die Pipeline angehalten oder rückgängig gemacht.

Wenn es sich bei der Anwendung um eine Web-App oder eine mobile App handelt, führen Sie automatisierte Integrationstests auf mehreren Desktop-Browsern oder echten Geräten durch. Dieser Ansatz ist besonders nützlich, um die Kompatibilität und Funktionalität mobiler Apps auf einer Vielzahl von Geräten zu überprüfen.

Implementierungsschritte

1. Schreiben Sie Modultests, bevor Sie funktionalen Code schreiben (testgesteuerte Entwicklung oder TDD). Legen Sie Coderichtlinien fest, sodass das Schreiben und Ausführen von Modultests zu den nicht-funktionalen Codierungsanforderungen gehört.

2. Erstellen Sie eine Suite automatisierter Integrationstests, die die identifizierten testbaren Funktionen abdecken. Diese Tests sollten Benutzerinteraktionen simulieren und die erwarteten Ergebnisse validieren.
3. Erstellen Sie die erforderliche Testumgebung für die Ausführung der Integrationstests. Hierzu können Staging- oder Vorproduktionsumgebungen gehören, die der Produktionsumgebung sehr ähnlich sind.
4. Richten Sie Ihre Quell-, Erstellungs-, Test- und Bereitstellungsphasen über die AWS CodePipeline-Konsole oder die AWS Command Line Interface (CLI) ein.
5. Stellen Sie die Anwendung bereit, sobald der Code erstellt und getestet wurde. AWS CodeDeploy kann die Anwendung in Ihren Staging- (Test-) und Produktionsumgebungen bereitstellen. Zu diesen Umgebungen können Amazon-EC2-Instances, AWS Lambda-Funktionen oder On-Premises-Server gehören. Für die Bereitstellung der Anwendung in allen Umgebungen sollte derselbe Bereitstellungsmechanismus verwendet werden.
6. Überwachen Sie den Fortschritt Ihrer Pipeline und den Status jeder Phase. Verwenden Sie Qualitätsprüfungen, um die Pipeline auf der Grundlage des Teststatus zu blockieren. Außerdem können Sie Benachrichtigungen über jeden Ausfall einer Pipeline-Phase oder den Abschluss einer Pipeline erhalten.
7. Überwachen Sie kontinuierlich die Testergebnisse und suchen Sie nach Mustern, Regressionen oder Bereichen, die mehr Aufmerksamkeit erfordern. Verwenden Sie diese Informationen, um die Testsuite zu verbessern, Bereiche der Anwendung zu identifizieren, in denen robustere Tests erforderlich sind, und den Bereitstellungsprozess zu optimieren.

Ressourcen

Zugehörige bewährte Methoden:

- [REL07-BP04 Durchführen von Lasttests für den Workload](#)
- [REL08-BP03 Integrieren von Ausfallsicherheitstests in die Bereitstellung](#)
- [REL12-BP04 Testen der Ausfallsicherheit mit Chaos-Engineering](#)

Zugehörige Dokumente:

- [Präskriptive AWS-Leitlinien: Testautomatisierung](#)
- [Kontinuierliche Bereitstellung und kontinuierliche Integration](#)
- [Anzeigen für Funktionstests](#)

- [Überwachen von Pipelines](#)
- [Verwenden von AWS CodePipeline mit AWS CodeBuild zum Testen von Code und Ausführen von Builds](#)
- [AWS Device Farm](#)

REL08-BP03 Integrieren von Ausfallsicherheitstests in die Bereitstellung

Integrieren Sie Resilienztests, indem Sie bewusst Fehler in Ihr System einleiten, um dessen Leistungsfähigkeit im Falle von Störszenarien zu messen. Resilienztests unterscheiden sich von Geräte- und Funktionstests, die normalerweise in Bereitstellungszyklen integriert werden, da sie sich auf die Identifizierung unerwarteter Ausfälle in Ihrem System konzentrieren. Es ist zwar sicher, in der Vorproduktion mit der Integration von Resilienztests zu beginnen, aber setzen Sie sich das Ziel, diese Tests im Rahmen Ihrer [GameDays](#) in der Produktion zu implementieren.

Gewünschtes Ergebnis: Resilienztests tragen dazu bei, Vertrauen in die Fähigkeit des Systems aufzubauen, Beeinträchtigungen in der Produktion standzuhalten. Experimente identifizieren Schwachstellen, die zu Ausfällen führen könnten. Auf diese Weise können Sie Ihr System verbessern, um Ausfälle und Beeinträchtigungen automatisch und effizient zu beheben.

Typische Anti-Muster:

- Mangelnde Beobachtbarkeit und Überwachung in Bereitstellungsprozessen
- Verlass auf Menschen, um Systemausfälle zu beheben
- Analysemechanismen von schlechter Qualität
- Fokus auf bekannte Probleme in einem System und das Fehlen von Experimenten, um unbekannte Probleme zu identifizieren
- Identifizieren von Fehlern, aber keine Lösung
- Keine Dokumentation der Erkenntnisse und keine Runbooks

Vorteile der Nutzung dieser bewährten Methode: Resilienztests, die in Ihre Bereitstellungen integriert sind, helfen dabei, unbekannte Probleme im System zu identifizieren, die andernfalls unbemerkt bleiben und zu Produktionsausfällen führen können. Die Identifizierung dieser unbekannten Probleme in einem System hilft Ihnen, Ergebnisse zu dokumentieren, Tests in Ihren CI/CD-Prozess zu integrieren und Runbooks zu erstellen, die die Schadensbegrenzung durch effiziente, wiederholbare Mechanismen vereinfachen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Die gängigsten Formen von Resilienztests, die in die Bereitstellungen Ihres Systems integriert werden können, sind die Notfallwiederherstellung und Chaos-Engineering.

- Fügen Sie Aktualisierungen Ihrer Notfallwiederherstellungspläne und Standardarbeitsanweisungen (SOPs) bei jeder wichtigen Bereitstellung hinzu.
- Integrieren Sie Zuverlässigkeitstests in Ihre automatisierten Bereitstellungspipelines. Services wie [AWS Resilience Hub](#) können [in Ihre CI/CD-Pipeline integriert werden](#), um kontinuierliche Resilienzbewertungen zu erstellen, die im Rahmen jeder Bereitstellung automatisch bewertet werden.
- Definieren Sie Ihre Anwendungen in AWS Resilience Hub. Resilienzanalysen generieren Codefragmente, die Sie bei der Erstellung von Wiederherstellungsprozeduren als AWS Systems Manager-Dokumente für Ihre Anwendungen unterstützen und eine Liste mit empfohlenen Amazon CloudWatch-Monitoren und -Alarmen enthalten.
- Sobald Ihre DR-Pläne und SOPs aktualisiert sind, führen Sie Notfallwiederherstellungstests durch, um sicherzustellen, dass sie wirksam sind. Mithilfe von Notfallwiederherstellungstests können Sie feststellen, ob Sie Ihr System nach einem Ereignis wiederherstellen und zum normalen Betrieb zurückkehren können. Sie können verschiedene Notfallwiederherstellungsstrategien simulieren und feststellen, ob Ihre Planung ausreicht, um Ihre Verfügbarkeitsanforderungen zu erfüllen. Zu den gängigen Notfallwiederherstellungsstrategien gehören Backup und Wiederherstellung, Pilot Light, Cold Standby, Warm Standby, Hot Standby und Aktiv-Aktiv. Sie alle unterscheiden sich in Kosten und Komplexität. Vor dem Notfallwiederherstellungstest empfehlen wir, dass Sie Ihr Recovery Time Objective (RTO) und Ihr Recovery Point Objective (RPO) definieren, um die Wahl der zu simulierenden Strategie zu vereinfachen. AWS bietet Notfallwiederherstellungstools wie [AWS Elastic Disaster Recovery](#), die Ihnen unter anderem den Einstieg in Ihre Planung und Tests erleichtern.
- Experimente im Bereich des Chaos-Engineering führen zu Störungen im System, wie z. B. Netzwerk- und Serviceausfällen. Durch die Simulation mit kontrollierten Ausfällen können Sie die Sicherheitsschwachstellen Ihres Systems erkennen und gleichzeitig die Auswirkungen der eingeführten Fehler eindämmen. Führen Sie wie bei den anderen Strategien kontrollierte Ausfallsimulationen in Umgebungen außerhalb der Produktion durch, indem Sie beispielsweise Services wie [AWS Fault Injection Service](#) nutzen, um vor dem Einsatz in der Produktion Vertrauen zu gewinnen.

Ressourcen

Zugehörige Dokumente:

- [Experiment with failure using resilience testing to build recovery preparedness](#)
- [Continually assessing application resilience with AWS Resilience Hub and AWS CodePipeline](#)
- [Disaster recovery \(DR\) architecture on AWS, part 1: Strategies for recovery in the cloud](#)
- [Verify the resilience of your workloads using Chaos Engineering](#)
- [Principles of Chaos Engineering](#)
- [Chaos Engineering Workshop](#)

Zugehörige Videos:

- [AWS re:Invent 2020: Testing Resilience using Chaos Engineering](#)
- [Improve Application Resilience with AWS Fault Injection Service](#)
- [Prepare & Protect Your Applications From Disruption With AWS Resilience Hub](#)

REL08-BP04 Bereitstellung mit einer unveränderlichen Infrastruktur

Eine unveränderliche Infrastruktur sieht vor, dass Updates, Sicherheits-Patches oder Konfigurationsänderungen nicht direkt in Produktions-Workloads durchgeführt werden. Wenn eine Änderung erforderlich ist, wird die Architektur auf einer neuen Infrastruktur eingerichtet und für die Produktion bereitgestellt.

Verfolgen Sie eine Strategie zur Bereitstellung einer unveränderlichen Infrastruktur, um die Zuverlässigkeit, Konsistenz und Reproduzierbarkeit Ihrer Workload-Bereitstellungen zu erhöhen.

Gewünschtes Ergebnis: Bei einer unveränderlichen Infrastruktur sind keine [direkten Änderungen](#) an den Infrastrukturressourcen innerhalb eines Workloads erlaubt. Wenn eine Änderung erforderlich ist, wird stattdessen ein neuer Satz aktualisierter Infrastrukturressourcen, der alle erforderlichen Änderungen enthält, parallel zu Ihren vorhandenen Ressourcen bereitgestellt. Diese Bereitstellung wird automatisch validiert und bei Erfolg wird der Datenverkehr schrittweise auf die neuen Ressourcen verlagert.

Diese Bereitstellungsstrategie gilt unter anderem für Softwareupdates, Sicherheits-Patches, Infrastrukturänderungen, Konfigurationsupdates und Anwendungsupdates.

Typische Anti-Muster:

- Implementieren von Änderungen an laufenden Infrastruktur-Ressourcen vor Ort.

Vorteile der Nutzung dieser bewährten Methode:

- Erhöhte Konsistenz zwischen verschiedenen Umgebungen: Da es keine Unterschiede bei den Infrastrukturressourcen zwischen den Umgebungen gibt, wird die Konsistenz erhöht und das Testen vereinfacht.
- Verringerung der Konfigurationsabweichungen: Durch das Ersetzen von Infrastrukturressourcen durch eine bekannte und versionskontrollierte Konfiguration wird die Infrastruktur in einen bekannten, getesteten und vertrauenswürdigen Zustand versetzt, wodurch Konfigurationsabweichungen vermieden werden.
- Zuverlässige atomare Bereitstellungen: Entweder werden die Verteilungen erfolgreich abgeschlossen oder es ändert sich nichts, was die Konsistenz und Zuverlässigkeit des Verteilungsprozesses erhöht.
- Vereinfachte Bereitstellungen: Bereitstellungen werden vereinfacht, da sie keine Upgrades unterstützen müssen. Upgrades sind einfach neue Bereitstellungen.
- Sicherere Bereitstellungen mit schnellen Rollback- und Wiederherstellungsprozessen: Bereitstellungen sind sicherer, da die vorherige funktionierende Version nicht geändert wird. Sie können einen Rollback zur vorherigen Version durchführen, wenn Fehler erkannt werden.
- Verbesserte Sicherheitslage: Indem Sie keine Änderungen an der Infrastruktur zulassen, können Fernzugriffsmechanismen (wie SSH) deaktiviert werden. Dadurch wird der Angriffsvektor reduziert und die Sicherheitslage Ihrer Organisation verbessert.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Automatisierung

Bei der Definition einer Strategie zur Bereitstellung einer unveränderlichen Infrastruktur empfiehlt es sich, die [Automatisierung](#) so weit wie möglich zu nutzen, um die Reproduzierbarkeit zu erhöhen und das Potenzial für menschliche Fehler zu minimieren. Weitere Informationen finden Sie unter [REL08-BP05 Automatisierung von Änderungen](#) und [Automating safe, hands-off deployments](#).

Mit [Infrastructure as code \(IaC\)](#) werden Schritte zur Bereitstellung, Orchestrierung und Implementierung der Infrastruktur auf programmatische, beschreibende und deklarative Weise

definiert und in einem Quellkontrollsystem gespeichert. Die Nutzung von Infrastructure as Code vereinfacht die Automatisierung der Infrastrukturbereitstellung und trägt zur Unveränderbarkeit der Infrastruktur bei.

Bereitstellungsmuster

Wenn eine Änderung des Workloads erforderlich ist, schreibt die Strategie der unveränderlichen Infrastrukturbereitstellung vor, dass ein neuer Satz von Infrastrukturressourcen bereitgestellt wird, einschließlich aller erforderlichen Änderungen. Es ist wichtig, dass diese neuen Ressourcen nach einem Muster eingeführt werden, das die Auswirkungen auf die Benutzer minimiert. Für diese Bereitstellung gibt es zwei Hauptstrategien:

[Canary-Bereitstellung](#): Hierbei wird eine kleine Anzahl Ihrer Kunden auf die neue Version umgestellt, die in der Regel auf einer einzelnen Service-Instance (dem Canary) ausgeführt wird. Anschließend überprüfen Sie sämtliche Verhaltensänderungen oder Fehler, die generiert werden. Sie können Datenverkehr aus der Canary-Umgebung entfernen, wenn kritische Probleme auftreten, und die Benutzer auf die vorherige Version zurücksetzen. Wenn die Bereitstellung erfolgreich verläuft, können Sie das gewünschte Tempo beibehalten und die Änderungen auf Fehler überwachen, bis der Bereitstellungsvorgang vollständig abgeschlossen ist. Sie können AWS CodeDeploy mit einer [Bereitstellungskonfiguration konfigurieren](#), die eine Canary-Bereitstellung ermöglicht.

[Blue/Green-Bereitstellung](#): Verhält sich ähnlich wie die Canary-Bereitstellung, nur dass eine komplette Flotte der Anwendung parallel bereitgestellt wird. Sie können Ihre Bereitstellungen über die zwei Stacks (blau und grün) alternieren. Auch hier können Sie Datenverkehr an die neue Version senden und einen Failback auf die alte Version durchführen, wenn bei der Bereitstellung Probleme auftreten. Normalerweise wird der gesamte Datenverkehr auf einmal umgeschaltet. Sie können Ihren Datenverkehr aber auch auf die Versionen verteilen, um die Einführung der neuen Version mithilfe der gewichteten DNS-Routing-Funktionen von AmazonRoute 53 durchzuführen. Sie können AWS CodeDeploy und [AWS Elastic Beanstalk](#) mit einer Bereitstellungskonfiguration konfigurieren, die eine Blue/Green-Bereitstellung ermöglicht.

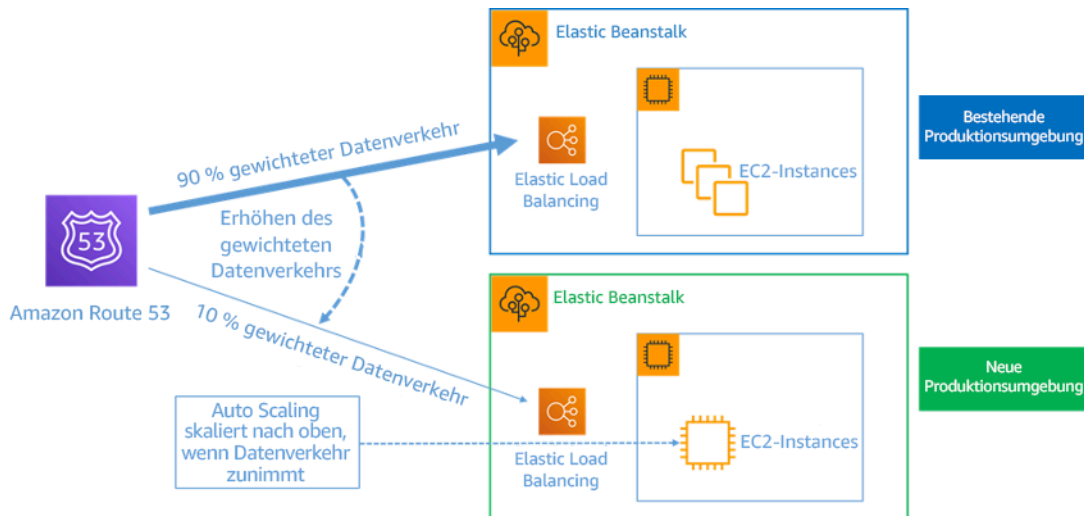


Abbildung 8: Blue/Green-Bereitstellung mit AWS Elastic Beanstalk und Amazon Route 53

Erkennung von Abweichungen

Als Abweichung wird jede Änderung bezeichnet, die dazu führt, dass eine Infrastrukturressource einen anderen Zustand oder eine andere Konfiguration aufweist als erwartet. Jede Art von nicht verwalteter Konfigurationsänderung widerspricht dem Konzept der unveränderlichen Infrastruktur und sollte erkannt und behoben werden, um eine erfolgreiche Implementierung der unveränderlichen Infrastruktur zu gewährleisten.

Implementierungsschritte

- Untersagen Sie die Änderung laufender Infrastruktur-Ressourcen an Ort und Stelle.
 - Sie können [AWS Identity and Access Management \(IAM\)](#) verwenden, um festzulegen, wer oder was auf Services und Ressourcen in AWS zugreifen darf, fein abgestufte Berechtigungen zentral verwalten und den Zugriff analysieren, um die Berechtigungen über AWS hinweg zu optimieren.
- Automatisieren Sie die Bereitstellung von Infrastrukturressourcen, um die Reproduzierbarkeit zu erhöhen und das Potenzial für menschliche Fehler zu minimieren.
 - Wie im Whitepaper [Introduction to DevOps on AWS](#) beschrieben, ist die Automatisierung ein Eckpfeiler der AWS-Services und wird intern in allen Services, Features und Angeboten unterstützt.
 - Durch die [Vorbereitung](#) Ihres Amazon Machine Image (AMI) können Sie die Zeit bis zum Start verkürzen. [EC2 Image Builder](#) ist ein vollständig verwalteter AWS-Service, der Ihnen hilft, die Erstellung, Wartung, Validierung, Freigabe und Bereitstellung von benutzerdefinierten, sicheren und aktuellen Linux- oder Windows-AMIs zu automatisieren.

- Zu den Services, die die Automatisierung unterstützen, gehören:
 - [AWS Elastic Beanstalk](#) ist ein Service zur schnellen Bereitstellung und Skalierung von Webanwendungen, die mit Java, .NET, PHP, Node.js, Python, Ruby, Go und Docker auf bekannten Servern wie Apache, NGINX, Passenger und IIS entwickelt wurden.
 - [AWS Proton](#) unterstützt Plattformteams dabei, all die verschiedenen Tools zu verbinden und zu koordinieren, die Ihre Entwicklungsteams für die Bereitstellung der Infrastruktur, die Bereitstellung von Code, die Überwachung und Updates benötigen. AWS Proton ermöglicht die automatisierte Bereitstellung von Infrastruktur als Code und die Bereitstellung von Serverless und containerbasierten Anwendungen.
- Die Nutzung von Infrastructure as Code erleichtert die Automatisierung der Infrastrukturbereitstellung und trägt zur Unveränderbarkeit der Infrastruktur bei. AWS bietet Services, die die Erstellung, Bereitstellung und Wartung der Infrastruktur auf programmatische, beschreibende und deklarative Weise ermöglichen.
 - [AWS CloudFormation](#) hilft Entwicklern dabei, AWS-Ressourcen in einer geordneten und vorhersehbaren Weise zu erstellen. Ressourcen werden in Textdateien im JSON- oder YAML-Format geschrieben. Die Vorlagen erfordern eine bestimmte Syntax und Struktur, die von den Arten der zu erstellenden und zu verwaltenden Ressourcen abhängt. Sie erstellen Ihre Ressourcen in JSON oder YAML mit einem beliebigen Code-Editor, checken sie in ein Versionskontrollsystem ein und CloudFormation baut dann die angegebenen Services auf sichere, wiederholbare Weise auf.
 - [AWS Serverless Application Model \(AWS SAM\)](#) ist ein Open-Source-Framework, mit dem Sie Serverless-Anwendungen in AWS erstellen können. AWS SAM lässt in anderen AWS-Services integrieren und ist eine Erweiterung von CloudFormation.
 - [AWS Cloud Development Kit \(AWS CDK\)](#) ist ein Open-Source-Framework für die Softwareentwicklung zur Modellierung und Bereitstellung Ihrer Cloud-Anwendungsressourcen mit Hilfe gängiger Programmiersprachen. Sie können AWS CDK verwenden, um die Anwendungsinfrastruktur mit TypeScript, Python, Java und .NET zu modellieren. AWS CDK verwendet CloudFormation im Hintergrund, um Ressourcen auf sichere, wiederholbare Weise bereitzustellen.
 - [AWS -Cloud-Control- API](#) führt einen gemeinsamen Satz von APIs zum Erstellen, Lesen, Aktualisieren, Löschen und Auflisten ein, mit denen Entwickler ihre Cloud-Infrastruktur auf einfache und konsistente Weise verwalten können. Die gemeinsamen Cloud Control API-APIs ermöglichen es Entwicklern, den Lebenszyklus von AWS- und Drittanbieter-Services einheitlich zu verwalten.
- Implementieren Sie Bereitstellungsmuster, die die Auswirkungen auf die Benutzer minimieren.

- Canary-Bereitstellungen:
 - [Einrichten einer API Gateway-Canary-Release-Bereitstellung](#)
 - [Erstellen einer Pipeline mit Canary-Bereitstellungen für Amazon ECS mit AWS App Mesh](#)
- Blau/Grün-Bereitstellungen: Das Whitepaper [Blue/Green Deployments on AWS](#) beschreibt [Beispieltechniken](#) zur Umsetzung von Blau/Grün-Bereitstellungsstrategien.
- Erkennen Sie Konfigurations- oder Zustandsabweichungen. Weitere Informationen finden Sie unter [Detecting unmanaged configuration changes to stacks and resources](#).

Ressourcen

Zugehörige bewährte Methoden:

- [REL08-BP05 Automatisieren von Änderungen](#)

Zugehörige Dokumente:

- [Automatisierung sicherer, vollautomatischer Bereitstellungen](#)
- [Leveraging AWS CloudFormation to create an immutable infrastructure at Nubank](#)
- [Infrastructure as Code](#)
- [Implementing an alarm to automatically detect drift in AWS CloudFormation stacks](#)

Zugehörige Videos:

- [AWS re:Invent 2020: Reliability, consistency, and confidence through immutability](#)

REL08-BP05 Automatisieren von Änderungen

Bereitstellungen und Patches werden automatisiert, um negative Auswirkungen zu vermeiden.

Änderungen an Produktionssystemen gehören in vielen Organisationen zu den größten Risikofaktoren. Neben den geschäftlichen Problemen, die durch die Software behoben werden, betrachten wir Bereitstellungen als vorrangiges Problem, das es zu lösen gilt. Heutzutage bedeutet das, wenn immer möglich und sinnvoll, Vorgänge zu automatisieren. Dazu gehören Tests und die Bereitstellung von Änderungen, das Hinzufügen oder Entfernen von Kapazität und das Migrieren von Daten.

Gewünschtes Ergebnis: Sie integrieren automatische Bereitstellungssicherheit in den Veröffentlichungsprozess mit umfangreichen Tests vor der Produktion, automatischen Rollbacks und gestaffelten Produktionsbereitstellungen. Diese Automatisierung minimiert die potenziellen Auswirkungen auf die Produktion, die durch fehlgeschlagene Bereitstellungen verursacht werden, und Entwickler müssen die Bereitstellungen nicht mehr aktiv bis zur Produktion beobachten.

Typische Anti-Muster:

- Sie führen manuelle Änderungen durch.
- Sie überspringen Schritte in Ihrer Automatisierung durch manuelle Notfall-Workflows.
- Sie folgen nicht Ihren etablierten Plänen und Prozessen zugunsten beschleunigter Zeitpläne.
- Sie führen schnelle Folgebereitstellungen durch, ohne entsprechende Bake-Zeit einzuräumen.

Vorteile der Nutzung dieser bewährten Methode: Wenn Sie alle Änderungen mithilfe der Automatisierung implementieren, vermeiden Sie das Risiko menschlicher Fehler und bieten die Möglichkeit, Tests durchzuführen, bevor Sie die Produktion ändern. Wenn Sie diesen Vorgang vor dem Produktionsstart durchführen, wird sichergestellt, dass Ihre Pläne vollständig sind. Darüber hinaus kann ein automatisches Rollback in Ihren Veröffentlichungsprozess Produktionsprobleme identifizieren und Ihren Workload wieder in den ursprünglichen Betriebszustand versetzen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Automatisieren Sie Ihre Bereitstellungs-Pipeline. Mit Bereitstellungs-Pipelines können Sie Tests und die Entdeckung von Anomalien automatisieren und die Pipeline an einem bestimmten Schritt vor der Bereitstellung in der Produktion anhalten oder eine Änderung automatisch zurückführen. Ein integraler Bestandteil davon ist die Einführung einer Kultur der [Continuous Integration und Continuous Delivery/Deployment](#) (CI/CD), bei der ein Commit oder eine Codeänderung verschiedene automatische Stage-Gates von der Build- und Testphase bis zur Bereitstellung in Produktionsumgebungen durchläuft.

Obwohl es immer noch als sinnvoll erachtet wird, Personen bei den komplexesten betrieblichen Abläufen einzubinden, empfehlen wir, diese Abläufe wegen ihrer Komplexität zu automatisieren.

Implementierungsschritte

Sie können Bereitstellungen automatisieren, um manuelle Operationen zu vermeiden, indem Sie die folgenden Schritte ausführen:

- Richten Sie ein Code-Repository ein, um Ihren Code sicher zu speichern: Verwenden Sie ein gehostetes Quellcodeverwaltungssystem, das auf einer beliebigen Technologie wie Git basiert, um Ihren Quellcode und Ihre Infrastructure-as-Code-Konfiguration (IaC) zu speichern.
- Konfigurieren eines Continuous-Integration-Services, um Ihren Quellcode zu kompilieren, Tests auszuführen und Bereitstellungsartefakte zu erstellen: Informationen zum Einrichten eines Build-Projekts für diesen Zweck finden Sie unter [Getting started with AWS CodeBuild using the console](#).
- Einrichten eines Bereitstellungsservices, der Anwendungsbereitstellungen automatisiert und die Komplexität von Anwendungsupdates mindert, ohne auf fehleranfällige manuelle Bereitstellungen angewiesen zu sein: [AWS CodeDeploy](#) automatisiert Softwarebereitstellungen für eine Vielzahl von Computing-Services wie Amazon EC2, [AWS Fargate](#), [AWS Lambda](#), und auf Ihren On-Premises-Servern. Informationen zur Konfiguration dieser Schritte finden Sie unter [Getting started with CodeDeploy](#).
- Konfigurieren eines Continuous-Integration-Services zur Automatisierung der Veröffentlichungspipelines für schnellere und zuverlässigere Anwendungs- und Infrastrukturupdates: Erwägen Sie die Verwendung von [AWS CodePipeline](#), um die Automatisierung Ihrer Veröffentlichungspipelines zu unterstützen. Weitere Informationen finden Sie in den [CodePipeline-Tutorials](#).

Ressourcen

Zugehörige bewährte Methoden:

- [OPS05-BP04 Einsatz von Systemen zur Build- und Bereitstellungsverwaltung](#)
- [OPS05-BP10 Vollständige Automatisierung von Integration und Bereitstellung](#)
- [OPS06-BP02 Testbereitstellungen](#)
- [OPS06-BP04 Automatisieren von Tests und Rollback](#)

Zugehörige Dokumente:

- [Continuous Delivery of Nested AWS CloudFormation Stacks Using AWS CodePipeline](#)
- [APN-Partner: Partner, die Sie beim Erstellen automatisierter Bereitstellungslösungen unterstützen können](#)
- [AWS Marketplace: Produkte zur Automatisierung Ihrer Bereitstellungen](#)
- [Automate chat messages with webhooks.](#)
- [Die Amazon Builders' Library: Gewährleistung von Rollback-Sicherheit während der Bereitstellung](#)

- [Die Amazon Builders' Library: Schneller mit kontinuierlicher Lieferung](#)
- [Was ist AWS CodePipeline?](#)
- [Was ist CodeDeploy?](#)
- [AWS Systems Manager Patch Manager](#)
- [Was ist Amazon SES?](#)
- [Was ist Amazon Simple Notification Service?](#)

Zugehörige Videos:

- [AWS Summit 2019: CI/CD on AWS](#)

Fehlerverwaltung

Fragen

- [REL 9. Wie werden Daten gesichert?](#)
- [REL 10. Wie wird die Fehlerisolierung zum Schützen der Workload verwendet?](#)
- [REL 11. Wie können Sie Workloads so gestalten, dass sie Komponentenausfällen gegenüber resilient sind?](#)
- [REL 12. Wie lässt sich die Zuverlässigkeit testen?](#)
- [REL 13. Was ist bei der Planung der Notfallwiederherstellung zu beachten?](#)

REL 9. Wie werden Daten gesichert?

Sichern Sie Daten, Anwendungen und Konfigurationen, um Ihre Anforderungen an Recovery Time Objectives (RTO) und Recovery Point Objectives (RPO) zu erfüllen.

Best Practices

- [REL09-BP01 Ermitteln und Sichern aller zu sichernden Daten oder Reproduzieren der Daten aus Quellen](#)
- [REL09-BP02 Schützen und Verschlüsseln von Backups](#)
- [REL09-BP03 Automatische Daten-Backups](#)
- [REL09-BP04 Verifizieren der Sicherungsintegrität und -verfahren durch regelmäßiges Wiederherstellen der Daten](#)

REL09-BP01 Ermitteln und Sichern aller zu sichernden Daten oder Reproduzieren der Daten aus Quellen

Sie sollten die Backup-Funktionen der von dem Workload genutzten Daten-Services und -Ressourcen verstehen und nutzen. Die meisten Services bieten Funktionen zur Sicherung von Workload-Daten.

Gewünschtes Ergebnis: Die Datenquellen wurden identifiziert und nach ihrer Bedeutung klassifiziert. Anschließend legen Sie eine auf dem RPO basierende Strategie für die Datenwiederherstellung fest. Diese Strategie involviert entweder die Sicherung dieser Datenquellen oder die Möglichkeit, Daten aus anderen Quellen zu reproduzieren. Im Falle eines Datenverlusts ermöglicht die implementierte Strategie die Wiederherstellung oder Reproduktion von Daten innerhalb der definierten RPO und RTO.

„Cloud-Reife“-Phase: Grundlegend

Typische Anti-Muster:

- Nicht alle Datenquellen für die Workload und deren Kritikalität sind bekannt.
- Es erfolgen keine Backups kritischer Datenquellen.
- Es erfolgen nur Backups von manchen Datenquellen ohne die Verwendung von Kritikalität als Kriterium.
- Es wurde kein RPO definiert oder die Backup-Häufigkeit kann den RPO nicht erfüllen.
- Es erfolgt keine Bewertung, ob ein Backup erforderlich ist oder ob Daten aus anderen Quellen reproduziert werden können.

Vorteile der Nutzung dieser bewährten Methode: Die Identifizierung der Stellen, an denen Backups erforderlich sind, und die Implementierung eines Mechanismus zur Erstellung von Backups oder die Möglichkeit, die Daten aus einer externen Quelle zu reproduzieren, verbessern die Fähigkeit zur Wiederherstellung und Wiederbeschaffung von Daten während eines Ausfalls.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Alle AWS-Datenspeicher bieten Backup-Möglichkeiten. Services wie Amazon RDS und Amazon DynamoDB unterstützen zusätzlich ein automatisiertes Backup, das eine zeitpunktbezogene Wiederherstellung (PITR) ermöglicht. So können Sie Backups zu einem beliebigen Zeitpunkt bis zu fünf Minuten oder weniger vor dem aktuellen Zeitpunkt wiederherstellen. Viele AWS-Services bieten

die Möglichkeit, Backups in eine andere AWS-Region zu kopieren. AWS Backup ist ein Tool, das Ihnen die Möglichkeit gibt, den Schutz Ihrer Daten über AWS-Services hinweg zu zentralisieren und zu automatisieren. Mit [AWS Elastic Disaster Recovery](#) können Sie komplette Workloads von Servern kopieren und eine kontinuierliche Datensicherung von On-Premises-Ressourcen, AZ-übergreifenden Ressourcen oder Regionen hinweg aufrechterhalten. Das Recovery Point Objective (RPO) liegt dabei im Sekundenbereich.

Amazon S3 kann als Backup-Ziel für selbstverwaltete und AWS-verwaltete Datenquellen verwendet werden. AWS-Services wie Amazon EBS, Amazon RDS und Amazon DynamoDB bieten integrierte Möglichkeiten zur Backup-Erstellung. Sicherungssoftware von Drittanbietern kann ebenfalls eingesetzt werden.

On-Premises-Daten können in der AWS Cloud mit [AWS Storage Gateway](#) oder [AWS DataSync](#) gesichert werden. Mit Amazon S3-Buckets können Sie diese Daten auf AWS speichern. Amazon S3 bietet mehrere Speicherebenen wie [Amazon Glacier](#) oder [Amazon Glacier Deep Archive](#), um die Kosten für den Datenspeicher zu senken.

Möglicherweise können Sie Ihre Datenwiederherstellungs-Anforderungen erfüllen, indem Sie Daten aus anderen Quellen reproduzieren. Zum Beispiel könnten [Amazon ElastiCache-Replikat-Knoten](#) oder [Amazon RDS-Lesereplikate](#) verwendet werden, um Daten zu reproduzieren, wenn der primäre Knoten verloren geht. In Fällen, in denen solche Quellen verwendet werden können, um Ihr [Recovery Point Objective \(RPO\)](#) und [Recovery Time Objective \(RTO\)](#) zu erfüllen, benötigen Sie möglicherweise kein Backup. Ein weiteres Beispiel: Wenn Sie mit Amazon EMR arbeiten, ist es möglicherweise nicht notwendig, ein Backup Ihres HDFS-Datenspeichers zu erstellen, solange Sie die [Daten aus Amazon S3 in Amazon EMR wiederherstellen können](#).

Bei der Auswahl einer Backup-Strategie sollten Sie die für die Datenwiederherstellung benötigte Zeit berücksichtigen. Diese hängt von der Art des Backups (im Falle einer Backup-Strategie) oder von der Komplexität des Datenreproduktions-Mechanismus ab. Die benötigte Zeit sollte im RTO für die Workload liegen.

Implementierungsschritte

1. Identifizieren Sie alle Datenquellen für die Workload. Daten können über verschiedene Ressourcen wie [Datenbanken](#), [Volumes](#), [Dateisysteme](#), [Protokollierungssysteme](#) und [Objektspeicher](#) gespeichert werden. Im Abschnitt Ressourcen finden Sie verwandte Dokumente zu den verschiedenen AWS-Services, mit denen Daten gespeichert werden, und zu den Backup-Möglichkeiten, die diese Services bieten.

2. Klassifizieren Sie Datenquellen basierend auf Kritikalität. Unterschiedliche Datensätze haben unterschiedliche Kritikalitäts-Niveaus für eine Workload und damit auch verschiedene Anforderungen an die Ausfallsicherheit. So können beispielsweise bestimmte kritische Daten einen RPO erfordern, der gegen Null geht, während bei anderen, weniger kritischen Daten, ein höherer RPO und somit ein gewisser Datenverlust toleriert werden kann. Ebenso können unterschiedliche Datensätze auch unterschiedliche RTO-Anforderungen haben.
3. Nutzen Sie AWS- oder Drittanbieter-Services, um Backups der Daten zu erstellen. [AWS Backup](#) ist ein verwalteter Service, der die Erstellung von Backups von verschiedenen Datenquellen auf AWS ermöglicht. [AWS Elastic Disaster Recovery](#) übernimmt die automatisierte sekundengenaue Replikation von Daten in einer AWS-Region. Die meisten AWS-Services verfügen zusätzlich über native Funktionen zur Erstellung von Backups. Der AWS Marketplace umfasst zahlreiche Lösungen, die diese Funktionen ebenfalls bieten. In den unten aufgeführten Ressourcen finden Sie Informationen darüber, wie Sie Backups von Daten aus verschiedenen AWS-Services erstellen können.
4. Für Daten, die nicht gesichert werden, sollten Sie einen Datenreproduktions-Mechanismus festlegen. Es gibt verschiedene Gründe dafür, Daten, die aus anderen Quellen reproduziert werden können, nicht zu sichern. Möglicherweise ergibt sich die Situation, dass es günstiger ist, Daten bei Bedarf aus Quellen zu reproduzieren als ein Backup zu erstellen, da mit der Speicherung von Backups gewisse Kosten verbunden sind. Ein weiterer Grund wäre, wenn das Wiederherstellen aus einem Backup länger dauert als die Reproduktion der Daten aus anderen Quellen, was zu einer Nichteinhaltung des RTO führen würde. In solchen Situationen sollten Sie sich einen Kompromiss überlegen und einen gut definierten Prozess festlegen, wie Daten aus diesen Quellen reproduziert werden können, wenn eine Datenwiederherstellung erforderlich ist. Wenn Sie beispielsweise Daten zur Analyse aus Amazon S3 in ein Data Warehouse (wie Amazon Redshift) oder einen MapReduce-Cluster (wie Amazon EMR) geladen haben, kann es sich dabei z. B. um Daten handeln, die aus anderen Quellen reproduziert werden können. Solange die Ergebnisse dieser Analysen gespeichert werden oder reproduzierbar sind, besteht kein Risiko eines Datenverlusts durch einen Ausfall im Data Warehouse oder MapReduce-Cluster. Andere Daten, die aus Quellen reproduziert werden können, sind Cache-Inhalte (z. B. Amazon ElastiCache) oder RDS Read Replicas.
5. Legen Sie eine Kadenz für die Sicherung von Daten fest. Das Erstellen von Datenquellen ist ein periodischer Prozess und die Häufigkeit sollte vom RPO abhängen.

Aufwand für den Implementierungsplan: Mittel.

Ressourcen

Zugehörige bewährte Methoden:

[REL13-BP01 Definieren von Wiederherstellungszielen bei Ausfällen und Datenverlusten](#)

[REL13-BP02 Verwenden von definierten Wiederherstellungsstrategien, um die Wiederherstellungsziele zu erreichen](#)

Zugehörige Dokumente:

- [Was ist AWS Backup?](#)
- [Was ist AWS DataSync?](#)
- [Was ist Volume Gateway?](#)
- [APN-Partner: Partner, die Sie bei der Sicherung unterstützen können](#)
- [AWS Marketplace: Für die Sicherung geeignete Produkte](#)
- [Amazon-EBS-Snapshots](#)
- [Backing Up Amazon EFS](#)
- [Backing up Amazon FSx for Windows File Server](#)
- [Backup and Restore for ElastiCache for Redis](#)
- [Creating a DB Cluster Snapshot in Neptune](#)
- [Creating a DB Snapshot](#)
- [Creating an EventBridge Rule That Triggers on a Schedule](#)
- [Cross-Region Replication](#) with Amazon S3
- [EFS-zu-EFS AWS Backup](#)
- [Exportieren von Protokolldaten nach Amazon S3](#)
- [Object lifecycle management](#)
- [On-Demand-Backup und Wiederherstellung für DynamoDB](#)
- [Zeitpunktbezogene Wiederherstellung für DynamoDB](#)
- [Working with Amazon OpenSearch Service Index Snapshots](#)
- [Was ist AWS Elastic Disaster Recovery?](#)

Zugehörige Videos:

- [AWS re:Invent 2021 - Backup, disaster recovery, and ransomware protection with AWS](#)
- [AWS Backup Demo: Cross-Account & Cross-Region Backup](#)
- [AWS re:Invent 2019: Deep dive on AWS Backup, ft. Rackspace \(STG341\)](#)

REL09-BP02 Schützen und Verschlüsseln von Backups

Kontrollieren und erkennen Sie den Zugriff auf Backups durch eine Authentifizierung und Autorisierung. Vermeiden und erkennen Sie mittels Verschlüsselung Beeinträchtigungen der Datenintegrität von Backups.

Implementieren Sie Sicherheitskontrollen, um einen unbefugten Zugriff auf Backup-Daten zu verhindern. Verschlüsseln Sie Backups, um die Vertraulichkeit und Integrität Ihrer Daten zu schützen.

Typische Anti-Muster:

- Derselbe Zugriff auf die Sicherungen und die automatisierte Wiederherstellung wie auf die Daten.
- Keine Verschlüsselung der Sicherungen.
- Keine Implementierung von Unveränderlichkeit zum Schutz vor Löschung oder Manipulation.
- Verwendung derselben Sicherheitsdomäne für Produktions- und Backup-Systeme.
- Keine Überprüfung der Integrität der Backups durch regelmäßige Tests.

Vorteile der Nutzung dieser bewährten Methode:

- Die Absicherung Ihrer Backups verhindert eine Manipulation der Daten und die Verschlüsselung der Daten sorgt dafür, dass kein Zugriff auf die Daten möglich ist, wenn sie versehentlich offengelegt werden.
- Verbesselter Schutz vor Ransomware und anderen Cyberbedrohungen, die auf die Backup-Infrastruktur abzielen.
- Kürzere Wiederherstellungszeit nach einem Cybervorfall dank geprüfter Wiederherstellungsprozesse.
- Verbesserte Fähigkeit zur Wahrung der Geschäftskontinuität bei Sicherheitsvorfällen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Steuern und erkennen Sie den Zugriff auf Backups durch Authentifizierung und Autorisierung wie z. B. mit AWS Identity and Access Management (IAM). Vermeiden und erkennen Sie mittels Verschlüsselung Beeinträchtigungen der Datenintegrität von Backups.

Amazon S3 unterstützt mehrere Verschlüsselungsmethoden für Daten im Ruhezustand. Mithilfe der serverseitigen Verschlüsselung akzeptiert Amazon S3 Ihre Objekte als unverschlüsselte Daten und sorgt für ihre Verschlüsselung bei der Speicherung. Bei der clientseitigen Verschlüsselung ist Ihre Workload-Anwendung für die Verschlüsselung der Daten verantwortlich, bevor sie an Amazon S3 gesendet werden. Beide Methoden ermöglichen Ihnen, zum Erstellen und Speichern des Datenschlüssels AWS Key Management Service (AWS KMS) zu verwenden oder einen eigenen Schlüssel bereitzustellen, für den Sie verantwortlich sind. Bei AWS KMS können Sie mithilfe von IAM festlegen, wer auf Ihre Datenschlüssel und entschlüsselten Daten zugreifen kann.

Wenn Sie bei Amazon RDS Ihre Datenbanken verschlüsseln, werden Ihre Sicherungsdaten ebenfalls verschlüsselt. DynamoDB-Sicherungen werden immer verschlüsselt. Bei Verwendung von AWS Elastic Disaster Recovery werden alle Daten während der Übertragung und im Ruhezustand verschlüsselt. Mit Elastic Disaster Recovery können Daten im Ruhezustand entweder mit dem standardmäßigen -Amazon EBS-Volume-Verschlüsselungsschlüssel oder einem vom Kunden verwalteten Schlüssel verschlüsselt werden.

Überlegungen zur Cyberresilienz

Um Ihre Backups besser vor Cyberbedrohungen zu schützen, sollten Sie neben der Verschlüsselung auch die Einführung dieser zusätzlichen Kontrollen in Betracht ziehen:

- Implementieren Sie Unveränderlichkeit mithilfe von AWS Backup Vault Lock oder Amazon S3 Object Lock, um zu verhindern, dass Backup-Daten während ihres Aufbewahrungszeitraums verändert oder gelöscht werden. Auf diese Weise schützen Sie sie vor Ransomware und böswilliger Löschung.
- Sorgen Sie für eine logische Trennung zwischen Produktions- und Backup-Umgebungen mit einem logischen Air-Gapped Vault von AWS Backup für kritische Systeme. Diese Trennung hilft, eine gleichzeitige Kompromittierung beider Umgebungen zu verhindern.
- Überprüfen Sie die Integrität Ihrer Backups regelmäßig. Stellen Sie dazu mithilfe von AWS Backup-Wiederherstellungstests sicher, dass die Backups nicht beschädigt sind und nach einem Cybervorfall erfolgreich wiederhergestellt werden können.
- Nutzen Sie die Mehrparteien-Genehmigung von AWS Backup für kritische Wiederherstellungsvorgänge, um sicherzustellen, dass keine unbefugten oder böswilligen

Wiederherstellungsversuche erfolgen können, da eine Autorisierung durch mehrere benannte Genehmigungsberechtigte erforderlich ist.

Implementierungsschritte

1. Verwenden Sie eine Verschlüsselung für jeden Datenspeicher. Wenn Ihre Quelldaten verschlüsselt sind, wird die Sicherung ebenfalls verschlüsselt.
 - [Nutzen Sie die Verschlüsselung in Amazon RDS](#). Beim Erstellen einer RDS-Instance können Sie die Verschlüsselung im Ruhezustand mit AWS Key Management Service konfigurieren.
 - [Nutzen Sie die Verschlüsselung von Amazon EBS-Volumes](#). Während der Erstellung von Volumes können Sie eine Standardverschlüsselung konfigurieren oder einen eindeutigen Schlüssel angeben.
 - Verwenden Sie die erforderliche [Amazon DynamoDB-Verschlüsselung](#). DynamoDB verschlüsselt alle Daten im Ruhezustand. Sie können entweder einen AWS-eigenen AWS KMS-Schlüssel oder einen AWS-verwalteten KMS-Schlüssel verwenden und dabei einen Schlüssel angeben, der in Ihrem Konto gespeichert wird.
 - [Verschlüsseln Sie Ihre in Amazon EFS gespeicherten Daten](#) Konfigurieren Sie die Verschlüsselung beim Erstellen des Dateisystems.
 - Konfigurieren Sie die Verschlüsselung in den Quell- und Zielregionen. Sie können die Verschlüsselung im Ruhezustand in Amazon S3 mit Schlüsseln konfigurieren, die in KMS gespeichert sind. Die Schlüssel sind jedoch regionsspezifisch. Sie können die Zielschlüssel angeben, während Sie die Replikation konfigurieren.
 - Entscheiden Sie sich für die Standardverschlüsselung oder die angepasste [Amazon EBS-Verschlüsselung für Elastic Disaster Recovery](#). Mit dieser Option werden Ihre replizierten Daten im Ruhezustand auf den Staging-Area Subnetz-Datenträgern und den replizierten Datenträgern verschlüsselt.
2. Implementieren Sie Rechte mit geringsten Berechtigungen für den Zugriff auf Ihre Backups. Begrenzen Sie den Zugriff auf die Backups, Snapshots und Replikate anhand [bewährter Methoden im Bereich Sicherheit](#).
3. Konfigurieren Sie Unveränderlichkeit für kritische Backups. Implementieren Sie für kritische Daten AWS Backup Vault Lock oder S3 Object Lock, um zu verhindern, dass diese Daten während des angegebenen Aufbewahrungszeitraums gelöscht oder verändert werden. Weitere Informationen zur Implementierung finden Sie unter [AWS Backup Vault Lock](#).
4. Schaffen Sie eine logische Trennung für Backup-Umgebungen. Implementieren Sie einen logischen Air-Gapped Vault von AWS Backup für kritische Systeme, die einen verbesserten Schutz

vor Cyberbedrohungen benötigen. Anleitungen zur Implementierung finden Sie unter [Building cyber resiliency with AWS Backup logically air-gapped vault](#).

5. Implementieren Sie Prozesse zur Backup-Validierung. Konfigurieren Sie AWS Backup-Wiederherstellungstests, um regelmäßig sicherzustellen, dass die Backups nicht beschädigt sind und nach einem Cybervorfall erfolgreich wiederhergestellt werden können. Weitere Informationen finden Sie unter [Validate recovery readiness with AWS Backup restore testing](#).
6. Konfigurieren Sie eine Mehrparteien-Genehmigung für Wiederherstellungsvorgänge vertraulicher Daten. Implementieren Sie für kritische Systeme die Mehrparteien-Genehmigung von AWS Backup, sodass die Wiederherstellung erst fortgesetzt werden kann, nachdem eine Genehmigung von mehreren benannten Genehmigungsberechtigten eingeholt wurde. Einzelheiten zur Implementierung finden Sie unter [Improve recovery resilience with AWS Backup support for Multi-party approval](#).

Ressourcen

Zugehörige Dokumente:

- [AWS Marketplace: Für die Sicherung geeignete Produkte](#)
- [Amazon-EBS-Verschlüsselung](#)
- [Amazon S3: Daten durch Verschlüsselung schützen](#)
- [Zusätzliche CRR-Konfiguration: Replizieren von Objekten, die mit serverseitiger Verschlüsselung \(SSE\) unter Verwendung von Verschlüsselungsschlüsseln erstellt wurden, die in gespeichert wurden AWS KMS](#)
- [DynamoDB-Verschlüsselung in Ruhezustand](#)
- [Verschlüsseln von Amazon RDS-Ressourcen](#)
- [Encrypting Data and Metadata in Amazon EFS](#)
- [Encryption for Backups in AWS](#)
- [Managing Encrypted Tables](#)
- [Säule „Sicherheit“ – AWS Well-Architected Framework](#)
- [Was ist AWS Elastic Disaster Recovery?](#)
- [FSISEC11: Wie schützen Sie sich vor Ransomware?](#)
- [Ransomware Risk Management on AWS Using the NIST Cyber Security Framework](#)
- [Building cyber resiliency with AWS Backup logically air-gapped vault](#)
- [Validate recovery readiness with AWS Backup restore testing](#)

- [Improve recovery resilience with AWS Backup support for Multi-party approval](#)

Zugehörige Beispiele:

- [Implementing Bi-Directional Cross-Region Replication \(CRR\) for Amazon S3](#)

REL09-BP03 Automatische Daten-Backups

Sie können die Backups so konfigurieren, dass sie automatisch nach einem Zeitplan, der auf dem Recovery Point Objective (RPO) basiert, oder bei Änderungen am Datensatz durchgeführt werden. Kritische Datensätze, bei denen Datenverlust vermieden werden sollte, müssen regelmäßig automatisch gesichert werden, wohingegen weniger kritische Daten, bei denen ein gewisser Verlust akzeptabel ist, weniger häufig gesichert werden können.

Gewünschtes Ergebnis: Ein automatisierter Prozess, der Backups von Datenquellen in einem festgelegten Rhythmus erstellt.

Typische Anti-Muster:

- Sicherungen werden manuell durchgeführt.
- Es werden Ressourcen mit Sicherungsfunktionen verwendet, die Sicherung wird aber nicht in die Automatisierung einbezogen.

Vorteile der Nutzung dieser bewährten Methode: Durch die Automatisierung von Backups wird sichergestellt, dass diese regelmäßig gemäß Ihrem RPO durchgeführt werden. Sie werden gewarnt, wenn sie nicht durchgeführt werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

AWS Backup kann zum Erstellen von automatisierten Daten-Backups verschiedener AWS-Datenquellen genutzt werden. Amazon RDS-Instances können fast kontinuierlich alle fünf Minuten gesichert werden und Amazon S3-Objekte können praktisch durchgehend alle 15 Minuten gesichert werden, was eine zeitpunktbezogene Wiederherstellung (PITR) an einem bestimmten Zeitpunkt im Backup-Verlauf ermöglicht. Andere AWS-Datenquellen wie Amazon EBS-Volumes, Amazon DynamoDB-Tabellen oder Amazon FSx-Dateisysteme kann AWS Backup stündlich ein automatisiertes Backup ausführen. Diese Services bieten außerdem native Backup-Funktionen. Zu den AWS-Services, die ein automatisiertes Backup mit zeitpunktbezogener Wiederherstellung

anbieten, gehören [Amazon DynamoDB](#), [Amazon RDS](#) und [Amazon Keyspaces \(für Apache Cassandra\)](#). Diese können bis zu einem bestimmten Zeitpunkt innerhalb der Backup-Historie wiederhergestellt werden. Die meisten anderen AWS-Datenspeicher-Services bieten die Möglichkeit, stündliche periodische Backups einzuplanen.

Amazon RDS und Amazon DynamoDB bieten ein kontinuierliches Backup mit zeitpunktbezogener Wiederherstellung. Die Amazon S3-Versionsverwaltung erfolgt nach der Aktivierung automatisch. Mit [Amazon Data Lifecycle Manager](#) können Sie das Erstellen, Kopieren und Löschen von Amazon EBS-Snapshots automatisieren. Außerdem können damit das Erstellen, das Kopieren, die Außerbetriebnahme und die Abmeldung von Amazon EBS-gestützten Amazon Machine Images (AMIs) und den zugrunde liegenden Amazon EBS-Snapshots automatisiert werden.

AWS Elastic Disaster Recovery bietet eine kontinuierliche Replikation auf Blockebene von der Quellumgebung (On-Premises oder AWS) zur Ziel-Wiederherstellungsregion. Point-in-Time-Amazon EBS-Snapshots werden automatisch vom Service erstellt und verwaltet.

Für eine zentrale Ansicht Ihrer Sicherungsautomatisierung und des Verlaufs bietet AWS Backup eine vollständig verwaltete, richtlinienbasierte Sicherungslösung. Diese zentralisiert und automatisiert die Sicherung von Daten in mehreren AWS-Services in der Cloud sowie On-Premises mithilfe des AWS Storage Gateway.

Zusätzlich zur Versionsverwaltung bietet Amazon S3 eine Replikationsfunktion. Der gesamte S3-Bucket kann automatisch in einen anderen Bucket in einer anderen AWS-Region repliziert werden.

Implementierungsschritte

1. Identifizieren Sie Datenquellen, die derzeit manuell gesichert werden. Weitere Details erhalten Sie unter [REL09-BP01 Ermitteln und Sichern aller zu sichernden Daten oder Reproduzieren der Daten aus Quellen](#).
2. Bestimmen Sie das RPO für den Workload. Weitere Details erhalten Sie unter [REL13-BP01 Definieren von Wiederherstellungszielen bei Ausfällen und Datenverlusten](#).
3. Nutzen Sie eine automatisierte Backup-Lösung oder einen verwalteten Service. AWS Backup ist ein vollständig verwalteter Service, der die [Zentralisierung und Automatisierung der Datensicherung über AWS-Services, in der Cloud und On-Premises](#) erleichtert. Mithilfe von Backup-Plänen in AWS Backup erstellen Sie Regeln, die die zu sichernden Ressourcen und die Häufigkeit, mit der diese Backups erstellt werden sollen, festlegen. Diese Häufigkeit sollte auf dem in Schritt 2 festgelegten RPO basieren. Eine praktische Anleitung für die Erstellung automatisierter Backups mit AWS Backup finden Sie unter [Testing Backup and Restore of Data](#). Native Backup-Funktionen werden von den meisten AWS-Services, die Daten speichern, angeboten. So kann

beispielsweise RDS für automatisierte Backups mit zeitpunktbezogener Wiederherstellung (PITR) genutzt werden.

4. Für Datenquellen, die nicht von einer automatisierten Backup-Lösung oder einem verwalteten Service unterstützt werden, wie z. B. On-Premises-Datenquellen oder Warteschlangen, sollten Sie eine zuverlässige Lösung eines Drittanbieters verwenden, um automatische Backups zu erstellen. Als Alternative können Sie die Automatisierung für diesen Vorgang mit der AWS CLI oder mit SDKs erstellen. Sie können AWS Lambda-Funktionen oder AWS Step Functions nutzen, um die Logik für die Erstellung eines Backups von Daten zu definieren und Amazon EventBridge einsetzen, um diese in einer Häufigkeit entsprechend Ihren RPOs aufzurufen.

Aufwand für den Implementierungsplan: Niedrig.

Ressourcen

Zugehörige Dokumente:

- [APN-Partner: Partner, die Sie bei der Sicherung unterstützen können](#)
- [AWS Marketplace: Für die Sicherung geeignete Produkte](#)
- [Creating an EventBridge Rule That Triggers on a Schedule](#)
- [Was ist AWS Backup?](#)
- [Was ist AWS Step Functions?](#)
- [Was ist AWS Elastic Disaster Recovery?](#)

Zugehörige Videos:

- [AWS re:Invent 2019: Deep dive on AWS Backup, ft. Rackspace \(STG341\)](#)

REL09-BP04 Verifizieren der Sicherungsintegrität und -verfahren durch regelmäßiges Wiederherstellen der Daten

Überprüfen Sie mit einem Wiederherstellungstest, ob sich mit Ihren Sicherungsverfahren das Recovery Time Objective (RTO) und das Recovery Point Objective (RPO) einhalten lassen.

Gewünschtes Ergebnis: Daten aus Backups werden regelmäßig mithilfe klar definierter Mechanismen wiederhergestellt, um zu überprüfen, ob die Wiederherstellung innerhalb des festgelegten Recovery Time Objective (RTO) für die Workload möglich ist. Überprüfen Sie, dass die Wiederherstellung aus einem Backup in eine Ressource erfolgt, die die Originaldaten enthält und dass keine dieser Daten

korrupt oder nicht zugänglich sind, sowie dass sich der Datenverlust im Rahmen des Recovery Point Objective (RPO) bewegt.

Typische Anti-Muster:

- Wiederherstellung eines Backups ohne Abfrage oder Abruf von Daten, um zu überprüfen, ob die Wiederherstellung funktionsfähig ist.
- Es wird angenommen, dass ein Backup existiert.
- Es wird angenommen, dass das Backup eines System voll funktionsfähig ist und Daten daraus wiederhergestellt werden können.
- Es wird angenommen, dass die Zeit für das Wiederherstellen von Daten aus einem Backup innerhalb des RTO für die Workload liegt.
- Es wird angenommen, dass die im Backup enthaltenen Daten in das RPO für die Workload fallen.
- Wiederherstellung bei Bedarf, ohne ein Runbook zu verwenden oder außerhalb eines etablierten automatisierten Verfahrens.

Vorteile der Nutzung dieser bewährten Methode: Durch das Testen der Wiederherstellung der Backups wird überprüft, ob Daten bei Bedarf wiederhergestellt werden können, ohne befürchten zu müssen, dass Daten fehlen oder beschädigt sein könnten, ob die Wiederherstellung innerhalb des RTO für die Workload möglich ist und ob Datenverluste innerhalb des RPO für die Workload fallen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Das Testen der Sicherungs- und Wiederherstellungsfunktionen stärkt das Vertrauen in die Fähigkeit zur Durchführung dieser Aktionen während eines Ausfalls. Stellen Sie regelmäßig Backups an einem neuen Speicherort wieder her und führen Sie Tests aus, um die Datenintegrität zu überprüfen. Einige übliche Tests sind die Überprüfung, ob alle Daten verfügbar, nicht beschädigt und zugreifbar sind und ob ein Datenverlust innerhalb des RPO für die Workload liegt. Solche Tests können dabei helfen, zu ermitteln, ob die Wiederherstellungsmechanismen schnell genug sind, um dem RTO der Workload gerecht zu werden.

Mit AWS können Sie eine Testumgebung einrichten und Ihre Sicherungen wiederherstellen, um RTO- und RPO-Funktionen zu bewerten und Tests für Dateninhalte und Integrität durchzuführen.

Darüber hinaus ermöglichen Amazon RDS und Amazon DynamoDB eine zeitpunktbezogene Wiederherstellung (PITR). Durch die kontinuierliche Sicherung können Sie Ihren Datensatz in den

Zustand zurücksetzen, in dem er sich an einem bestimmten Datum und zu einer bestimmten Uhrzeit befand.

Testen Sie, ob alle Daten verfügbar, nicht beschädigt und zugreifbar sind und ob ein Datenverlust innerhalb des RPO für die Workload liegt. Solche Tests können dabei helfen, zu ermitteln, ob die Wiederherstellungsmechanismen schnell genug sind, um dem RTO der Workload gerecht zu werden.

AWS Elastic Disaster Recovery bietet eine kontinuierliche, zeitpunktbezogene Wiederherstellung von Snapshots von Amazon EBS-Volumes. Bei der Replikation von Quellservern werden die Point-in-Time-Zustände auf der Grundlage der konfigurierten Richtlinie im Laufe der Zeit aufgezeichnet. Elastic Disaster Recovery hilft Ihnen, die Integrität dieser Snapshots zu überprüfen, indem Sie Instances zu Test- und Übungszwecken starten, ohne den Datenverkehr weiterzuleiten.

Implementierungsschritte

1. Identifizieren Sie Datenquellen, die derzeit gesichert werden, und den Speicherort dieser Backups. Eine Anleitung zur Implementierung finden Sie unter [REL09-BP01 Ermitteln und Sichern aller zu sichernden Daten oder Reproduzieren der Daten aus Quellen](#).
2. Legen Sie für jede Datenquelle Kriterien für die Datenvalidierung fest. Verschieden Datentypen können unterschiedliche Eigenschaften aufweisen und somit auch unterschiedliche Validierungsmechanismen erfordern. Überlegen Sie, wie diese Daten validiert werden können, bevor Sie sie in der Produktion einsetzen. Häufig werden für die Datenvalidierung Daten- und Sicherungseigenschaften wie Datentyp, Format, Prüfsumme, Größe oder eine Kombination dieser Eigenschaften mit einer benutzerdefinierten Validierungslogik verwendet. Ein Beispiel hierfür wäre der Vergleich der Prüfsummenwerte zwischen der wiederhergestellten Ressource und der Datenquelle zum Zeitpunkt der Erstellung des Backups.
3. Bestimmen Sie das RTO und RPO für die Wiederherstellung der Daten auf der Grundlage der Kritikalität der Daten. Eine Anleitung zur Implementierung finden Sie unter [REL13-BP01 Definieren von Wiederherstellungszielen bei Ausfällen und Datenverlusten](#).
4. Bewerten Sie Ihre Wiederherstellungsfunktion. Prüfen Sie Ihre Sicherungs- und Wiederherstellungsstrategie, um festzustellen, ob sie Ihr RTO und RPO erfüllen kann, und passen Sie die Strategie bei Bedarf an. Mithilfe von [AWSResilience Hub](#) können Sie eine Bewertung Ihrer Workload durchführen. Dabei wird Ihre Anwendungsconfiguration im Hinblick auf die Ausfallsicherheitsrichtlinien bewertet und Sie erfahren, ob Ihre RTO- und RPO-Ziele erfüllt werden können.
5. Führen Sie eine Testwiederherstellung mit derzeit etablierten Prozessen durch, die in der Produktion für die Datenwiederherstellung verwendet werden. Diese Prozesse hängen davon ab,

- wie die ursprüngliche Datenquelle gesichert wurde sowie vom Format und der Speicherung des Backups selbst oder davon, ob die Daten aus anderen Quellen reproduziert werden. Wenn Sie beispielsweise einen verwalteten Service wie [AWS Backup verwenden, kann es sich einfach um das Wiederherstellen des Backups auf einer neuen Ressource handeln](#). Wenn Sie AWS Elastic Disaster Recovery verwendet haben, können Sie [eine Wiederherstellungsübung starten](#).
6. Überprüfen Sie die Datenwiederherstellung von der wiederhergestellten Ressource anhand von Kriterien, die Sie zuvor für die Datenvalidierung festgelegt haben. Enthalten die wiederhergestellten Daten den neuesten Datensatz bzw. das neueste Element zum Zeitpunkt des Backups? Fallen diese Daten in das RPO für die Workload?
 7. Messen Sie den Zeitaufwand für die Wiederherstellung und vergleichen Sie ihn mit Ihrem festgelegten RTO. Ist dieser Prozess Teil des RTO für die Workload? Vergleichen Sie beispielsweise den Zeitstempel des Starts des Wiederherstellungsprozesses und des Abschlusses der Wiederherstellungsbewertung, um zu ermitteln, wie lange dieser Prozess dauert. Alle AWS API-Aufrufe sind mit einem Zeitstempel versehen und diese Informationen sind in [AWS CloudTrail](#) verfügbar. Während diese Informationen Details dazu liefern können, wann der Wiederherstellungsprozess gestartet wurde, sollte der End-Zeitstempel für den Abschluss der Validierung von der Validierungslogik aufgezeichnet werden. Wenn Sie einen automatisierten Prozess verwenden, können Services wie [Amazon DynamoDB](#) verwendet werden, um diese Informationen zu speichern. Darüber hinaus können viele AWS-Services ein Ereignisprotokoll bereitstellen, das mit einem Zeitstempel versehene Informationen dazu enthält, wann bestimmte Aktionen aufgetreten sind. In AWS Backup werden Sicherungs- und Wiederherstellungsaktionen als Aufträge bezeichnet. Diese Aufträge enthalten Zeitstempelinformationen als Teil der Metadaten, anhand derer die für die Wiederherstellung benötigte Zeit gemessen werden kann.
 8. Informieren Sie die Stakeholder, wenn die Datenvalidierung fehlschlägt oder wenn die für die Wiederherstellung benötigte Zeit das für die Workload festgelegte RTO überschreitet. Bei der Implementierung von Automatisierung zu diesem Zweck ([wie in dieser Übung](#)) können Services wie Amazon Simple Notification Service (Amazon SNS) verwendet werden, um Push-Benachrichtigungen wie E-Mail- oder SMS-Nachrichten an Stakeholder zu senden. [Diese Nachrichten können auch in Messaging-Anwendungen wie Amazon Chime, Slack oder Microsoft Teams veröffentlicht werden](#) oder [zum Erstellen von Aufgaben als OpsItems mit AWS Systems Manager OpsCenter verwendet werden](#).
 9. Automatisieren Sie diesen Prozess so, dass er regelmäßig ausgeführt wird. Sie können beispielsweise Services wie AWS Lambda oder einen Zustandsautomaten in AWS Step Functions nutzen, um die Wiederherstellungsprozesse zu automatisieren. Außerdem können Sie Amazon EventBridge verwenden, um diesen automatisierten Workflow regelmäßig aufzurufen, wie im folgenden Architekturdiagramm abgebildet. Erfahren Sie, wie Sie die [Validierung der](#)

[Datenwiederherstellung mit AWS Backup automatisieren](#). Darüber hinaus bietet [dieses Well-Architected-Lab](#) praktische Erfahrungen mit einer Möglichkeit, mehrere der hier beschriebenen Schritte zu automatisieren.

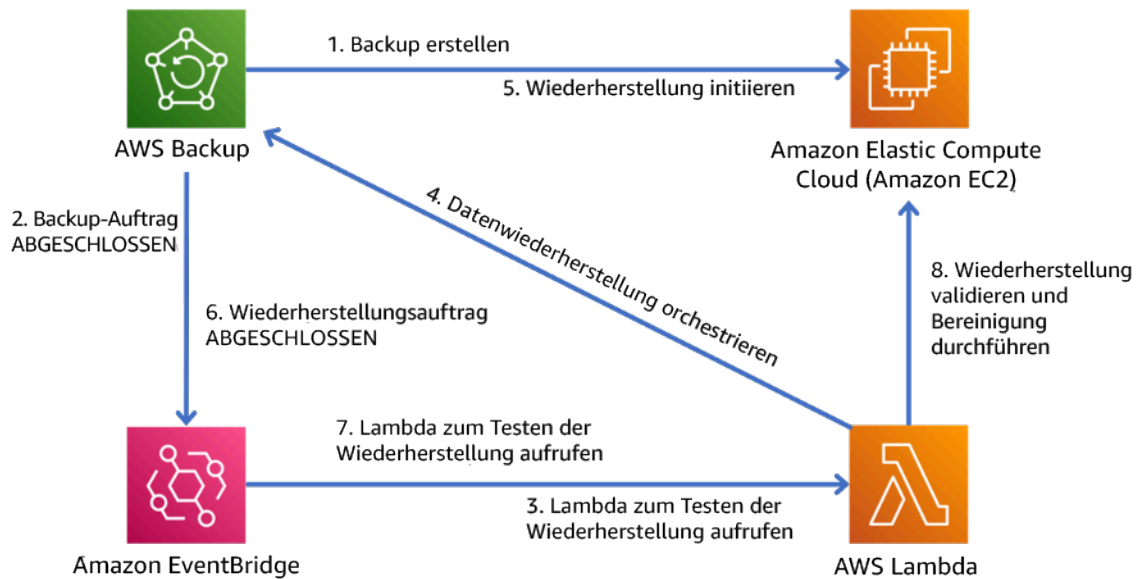


Abbildung 9. Ein automatisierter Sicherungs- und Wiederherstellungsprozess

Aufwand für den Implementierungsplan: Mittel bis hoch, abhängig von der Komplexität der Validierungskriterien.

Ressourcen

Zugehörige Dokumente:

- [Validierung der Datenwiederherstellung mit AWS Backup automatisieren](#).
- [APN-Partner: Partner, die Sie bei der Sicherung unterstützen können](#)
- [AWS Marketplace: Für die Sicherung geeignete Produkte](#)
- [Erstellen einer EventBridge-Regel, die nach einem Zeitplan ausgelöst wird](#)
- [On-Demand-Backup und Wiederherstellung für DynamoDB](#)
- [Was ist AWS Backup?](#)
- [Was ist AWS Step Functions?](#)
- [Was ist AWS Elastic Disaster Recovery](#)
- [AWS Elastic Disaster Recovery](#)

REL 10. Wie wird die Fehlerisolierung zum Schützen der Workload verwendet?

Die Fehlerisolierung begrenzt die Auswirkungen eines Komponenten- oder Systemausfalls auf eine definierte Grenze. Bei ordnungsgemäßer Isolierung sind Komponenten außerhalb der Grenze nicht vom Ausfall betroffen. Wenn Sie Ihren Workload über mehrere Fehlerisolierungsgrenzen hinweg ausführen, kann er anfälliger für Ausfälle werden.

Best Practices

- [REL10-BP01 Bereitstellen des Workloads an mehreren Standorten](#)
- [REL10-BP03 Automatisierte Wiederherstellung für Komponenten, die auf einen einzelnen Standort beschränkt sind](#)
- [REL10-BP04 Verwenden von Bulkhead-Architekturen, um den Umfang von Beeinträchtigungen zu begrenzen](#)

REL10-BP01 Bereitstellen des Workloads an mehreren Standorten

Verteilen Sie die Workload-Daten und -Ressourcen über mehrere Availability Zones oder ggf. über mehrere AWS-Regionen.

Ein grundlegendes Prinzip für das Servicedesign in AWS ist die Vermeidung von Single Points of Failure, einschließlich der zugrunde liegenden physischen Infrastruktur. AWS bietet Cloud-Computing-Ressourcen und -Services weltweit an mehreren geographischen Standorten, den sogenannten [Regionen](#). Jede Region ist physisch und logisch unabhängig und besteht aus drei oder mehr [Availability Zones \(AZs\)](#). Availability Zones liegen geographisch nahe beieinander, sind jedoch physisch voneinander getrennt und isoliert. Wenn Sie Ihre Workloads auf Availability Zones und Regionen verteilen, minimieren Sie das Risiko von Bedrohungen wie Bränden, Überschwemmungen, wetterbedingten Katastrophen, Erdbeben und menschlichem Versagen.

Entwickeln Sie eine Standortstrategie, um eine hohe Verfügbarkeit zu gewährleisten, die für Ihre Workloads geeignet ist.

Gewünschtes Ergebnis: Produktionsworkloads werden auf mehrere Availability Zones (AZs) oder Regionen verteilt, um Fehlertoleranz und Hochverfügbarkeit zu erreichen.

Typische Anti-Muster:

- Ihr Produktionsworkload ist nur in einer einzelnen Availability Zone vorhanden.
- Sie implementieren eine Architektur mit mehreren Regionen, wenn eine Multi-AZ-Architektur die geschäftlichen Anforderungen erfüllen würde.

- Ihre Bereitstellungen oder Daten werden desynchronisiert, was zu Konfigurationsabweichungen oder unzureichend replizierten Daten führt.
- Sie berücksichtigen nicht die Abhängigkeiten zwischen Anwendungskomponenten, wenn sich die Anforderungen an Ausfallsicherheit und mehrere Standorte zwischen diesen Komponenten unterscheiden.

Vorteile der Nutzung dieser bewährten Methode:

- Ihre Workloads sind widerstandsfähiger gegen Vorfälle wie Strom- oder Umgebungssteuerungsausfälle, Naturkatastrophen, Upstream-Serviceausfälle oder Netzwerkprobleme, die sich auf eine AZ oder eine ganze Region auswirken.
- Sie können auf einen größeren Bestand an Amazon-EC2-Instances zugreifen und die Wahrscheinlichkeit von `InsufficientCapacityExceptions` (ICE) verringern, wenn Sie bestimmte EC2-Instance-Typen starten.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Stellen Sie alle Produktionsworkloads in mindestens zwei Availability Zones (AZs) in einer Region bereit und führen Sie diese dort aus.

Verwenden mehrerer Availability Zones

Availability Zones sind Standorte für das Hosting von Ressourcen, die physisch voneinander getrennt sind, um korrelierte Ausfälle aufgrund von Risiken wie Bränden, Überschwemmungen und Tornados zu vermeiden. Jede Availability Zone verfügt über eine unabhängige physische Infrastruktur, einschließlich Stromversorgungsanschlüssen, Notstromquellen, mechanischen Services und Netzwerkkonnektivität. Dadurch bleiben Fehler in einer dieser Komponenten nur auf die betroffene Availability Zone beschränkt. Wenn beispielsweise aufgrund eines AZ-weiten Vorfalls EC2-Instances in der betroffenen Availability Zone nicht verfügbar sind, bleiben Ihre Instances in einer anderen Availability Zone weiterhin verfügbar.

Obwohl die Availability Zones in derselben AWS-Region physisch voneinander getrennt sind, sind sie einander nah genug, um Netzwerke mit hohem Durchsatz und niedriger Latenz (einstellige Millisekundenbeträge) bereitzustellen. Sie können Daten für die meisten Workloads synchron zwischen Availability Zones replizieren, ohne den Benutzerkomfort wesentlich zu beeinträchtigen. So können Sie Availability Zones in einer Aktiv/Aktiv- oder Aktiv/Standby-Konfiguration nutzen.

Die gesamte mit Ihren Workloads verbundene Computingleistung sollte auf mehrere Availability Zones verteilt werden. Dazu gehören [Amazon-EC2-Instances](#), [AWS Fargate](#)-Aufgaben und VPC-verbundene [AWS Lambda](#)-Funktionen. AWS-Computingservices, darunter [EC2 Auto Scaling](#), [Amazon Elastic Container Service \(ECS\)](#) und [Amazon Elastic Kubernetes Service \(EKS\)](#), bieten Ihnen Möglichkeiten, Computingleistung in Availability Zones zu starten und zu verwalten. Konfigurieren Sie diese so, dass die Computingleistung bei Bedarf in einer anderen Availability Zone automatisch ersetzt wird, um die Verfügbarkeit aufrechtzuerhalten. Um Datenverkehr an verfügbare Availability Zones weiterzuleiten, platzieren Sie einen Load Balancer vor Ihrem Computer, z. B. einen Application Load Balancer oder Network Load Balancer. AWS Load Balancer können den Datenverkehr im Falle einer Beeinträchtigung der Availability Zone auf verfügbare Instances umleiten.

Sie sollten auch Daten für Ihre Workloads replizieren und sie in mehreren Availability Zones verfügbar machen. Einige AWS-verwaltete Datenservices wie [Amazon S3](#), [Amazon Elastic File Service \(EFS\)](#), [Amazon Aurora](#), [Amazon DynamoDB](#), [Amazon Simple Queue Service \(SQS\)](#) und [Amazon Kinesis Data Streams](#) replizieren Daten standardmäßig in mehreren Availability Zones und bieten Widerstand gegen Beeinträchtigungen der Availability Zone. Bei anderen AWS-verwalteten Datenservices wie [Amazon Relational Database Service \(RDS\)](#), [Amazon Redshift](#) und [Amazon ElastiCache](#) müssen Sie die Multi-AZ-Replikation aktivieren. Nach der Aktivierung erkennen diese Services automatisch eine Beeinträchtigung der Availability Zone, leiten Anfragen an eine verfügbare Availability Zone weiter und replizieren Daten nach Bedarf nach der Wiederherstellung ohne Kundeneingriff erneut. Machen Sie sich mit dem Benutzerhandbuch für jeden von Ihnen verwendeten AWS-verwalteten Datenservice vertraut, um sich mit den Multi-AZ-Funktionen, Verhaltensweisen und Abläufen auszukennen.

Wenn Sie selbstverwalteten Speicher wie [Amazon Elastic Block Store \(EBS\)](#)-Volumes oder Amazon-EC2-Instance-Speicher verwenden, müssen Sie die Multi-AZ-Replikation selbst verwalten.

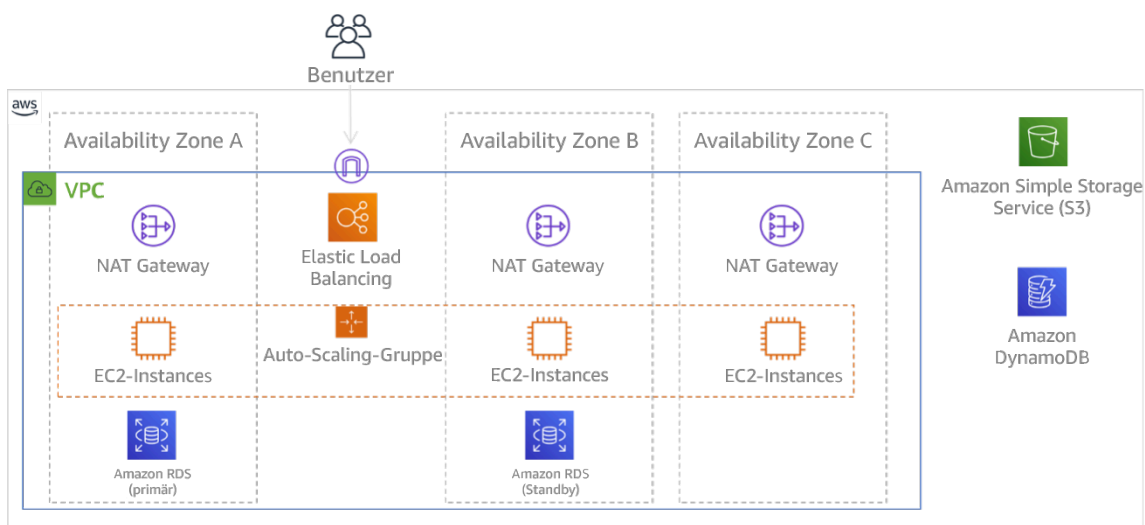


Abbildung 9: Mehrstufige Architektur, die in drei Availability Zones bereitgestellt wird. Amazon S3 und Amazon DynamoDB nutzen immer automatisch mehrere AZs. Auch der ELB wird in allen drei Zonen bereitgestellt.

Verwenden von mehreren AWS-Regionen

Wenn Sie über Workloads verfügen, die extreme Widerstandsfähigkeit erfordern (z. B. kritische Infrastrukturen, gesundheitsbezogene Anwendungen oder Services mit strengen kundenspezifischen oder gesetzlich vorgeschriebenen Verfügbarkeitsanforderungen), benötigen Sie möglicherweise zusätzliche Verfügbarkeit, die über das hinausgeht, was eine einzelne AWS-Region bieten kann. In diesem Fall sollten Sie Ihre Workloads auf mindestens zwei AWS-Regionen verteilen und dort ausführen (vorausgesetzt, dass Ihre Anforderungen an die Datenresidenz dies zulassen).

AWS-Regionen befinden sich in verschiedenen geographischen Regionen der Welt und auf mehreren Kontinenten. AWS-Regionen verfügen über eine noch stärkere physische Trennung und Isolierung als Availability Zones allein. AWS-Services, mit wenigen Ausnahmen, nutzen dieses Konzept, um völlig unabhängig zwischen verschiedenen Regionen zu operieren (auch bekannt als Regionalservices). Ein Ausfall eines Services in einer AWS-Region soll den Services in einer anderen Region nicht beeinträchtigen.

Wenn Sie Ihre Workloads in mehreren Regionen betreiben, sollten Sie zusätzliche Anforderungen berücksichtigen. Da Ressourcen in verschiedenen Regionen voneinander getrennt und unabhängig voneinander sind, müssen Sie die Komponenten Ihrer Workloads in jeder Region duplizieren. Dazu gehören neben Rechen- und Datenservices auch grundlegende Infrastrukturen wie VPCs.

HINWEIS: Wenn Sie ein multiregionales Design in Betracht ziehen, stellen Sie sicher, dass Ihre Workloads in einer einzigen Region ausgeführt werden können. Wenn Sie Abhängigkeiten zwischen Regionen erstellen, in denen eine Komponente in einer Region auf Services oder Komponenten in einer anderen Region angewiesen ist, könnt dies das Ausfallrisiko erhöhen und die Zuverlässigkeit erheblich beeinträchtigen.

Um multiregionale Bereitstellungen zu vereinfachen und die Konsistenz aufrechtzuerhalten, kann [AWS CloudFormation StackSets](#) Ihre gesamte AWS-Infrastruktur über mehrere Regionen hinweg replizieren. [AWS CloudFormation](#) kann auch Konfigurationsabweichungen erkennen und Sie informieren, wenn Ihre AWS-Ressourcen in einer Region nicht mehr synchron sind. Viele AWS-Services bieten die Replikation wichtiger Workload-Assets in mehreren Regionen. Beispielsweise kann [EC2 Image Builder](#) Ihre EC2-Machine-Images (AMIs) nach jedem Build in jeder Region, die Sie verwenden, veröffentlichen. [Amazon Elastic Container Registry \(ECR\)](#) kann Ihre Container-Images in Ihre ausgewählten Regionen replizieren.

Sie müssen Ihre Daten auch in jeder der von Ihnen ausgewählten Regionen replizieren. Viele AWS-verwaltete Datenservices bieten multiregionale Replikationsfunktionen, wie z. B. Amazon S3, Amazon DynamoDB, Amazon RDS, Amazon Aurora, Amazon Redshift, Amazon ElastiCache und Amazon EFS. [Globale Amazon DynamoDB-Tabellen](#) akzeptieren Schreibvorgänge in jeder unterstützten Region und replizieren Daten zwischen allen Ihren anderen konfigurierten Regionen. Bei anderen Services müssen Sie eine primäre Region für Schreibvorgänge angeben, da andere Regionen schreibgeschützte Replikate enthalten. Informationen zu den einzelnen AWS-verwalteten Datenservices, die Ihre Workloads verwenden, finden Sie im zugehörigen Benutzer- und Entwicklerhandbuch, wo Sie mehr über die Funktionen und Einschränkungen für mehrere Regionen erfahren. Achten Sie besonders darauf, wohin Schreibvorgänge geleitet werden müssen, welche Transaktionsmöglichkeiten und Einschränkungen gelten, wie die Replikation durchgeführt wird, und wie die Synchronisation zwischen Regionen überwacht wird.

AWS bietet außerdem die Möglichkeit, den Anforderungsdatenverkehr mit großer Flexibilität an Ihre regionalen Bereitstellungen weiterzuleiten. Beispielsweise können Sie Ihre DNS-Datensätze mithilfe von [Amazon Route 53](#) so konfigurieren, dass der Datenverkehr in die Region geleitet wird, die dem Benutzer am nächsten liegt. Alternativ können Sie Ihre DNS-Datensätze in einer Aktiv-/Standby-Konfiguration konfigurieren, in der Sie eine Region als primär festlegen und nur dann auf ein regionales Replikat zurückgreifen, wenn die primäre Region fehlerhaft wird. Sie können [Route 53-Zustandsprüfungen](#) so konfigurieren, dass sie fehlerhafte Endpunkte erkennen und einen automatischen Failover durchführen. Darüber hinaus können Sie [Amazon Application Recovery Controller \(ARC\)](#) verwenden, um eine hochverfügbare Routing-Steuerung für die manuelle Umleitung von Datenverkehr nach Bedarf bereitzustellen.

Auch wenn Sie sich dafür entscheiden, aus Gründen der Hochverfügbarkeit nicht in mehreren Regionen zu operieren, sollten Sie mehrere Regionen als Teil Ihrer Notfallwiederherstellungsstrategie (DR) in Betracht ziehen. Wenn möglich, replizieren Sie die Infrastrukturkomponenten und Daten Ihrer Workloads in einer Warm-Standby- oder Pilot-Light-Konfiguration in einer sekundären Region. Dabei replizieren Sie die Basisinfrastruktur aus der primären Region wie VPCs, Auto-Scaling-Gruppen, Container-Orchestratoren und andere Komponenten, konfigurieren aber die Komponenten variabler Größe in der Standby-Region (wie die Anzahl der EC2-Instances und Datenbankreplikate) so, dass sie eine minimal funktionsfähige Größe haben. Sie sorgen auch für eine kontinuierliche Datenreplikation von der primären Region zur Standby-Region. Wenn ein Vorfall eintritt, können Sie die Ressourcen in der Standby-Region skalieren oder vergrößern und sie dann zur primären Region heraufstufen.

Implementierungsschritte

1. Ermitteln Sie gemeinsam mit geschäftlichen Stakeholdern und Datenresidenzexperten die AWS-Regionen, die zum Hosten Ihrer Ressourcen und Daten verwendet werden können.
2. Bewerten Sie Ihre Workloads in Zusammenarbeit mit geschäftlichen und technischen Stakeholdern, um festzustellen, ob die Anforderungen an die Ausfallsicherheit durch einen Multi-AZ-Ansatz (einzelne AWS-Region) erfüllt werden können, oder ob ein multiregionaler Ansatz erforderlich ist (wenn mehrere Regionen zulässig sind). Die Verwendung mehrerer Regionen kann zu einer höheren Verfügbarkeit führen, jedoch auch zusätzliche Komplexität und Kosten mit sich bringen. Berücksichtigen Sie bei Ihrer Bewertung die folgenden Faktoren:
 - a. Geschäftsziele und Kundenanforderungen: Welche Ausfallzeiten sind zulässig, wenn in einer Availability Zone oder einer Region ein Vorfall eintritt, der sich auf Workloads auswirkt? Bewerten Sie Ihre Ziele für den Wiederherstellungspunkt wie in [REL13-BP01 Definieren von Wiederherstellungszielen bei Ausfällen und Datenverlusten](#) beschrieben.
 - b. Anforderungen an die Notfallwiederherstellung (DR): Gegen welche Art von potenziellen Notfällen möchten Sie sich versichern? Ziehen Sie die Möglichkeit eines Datenverlusts oder einer langfristigen Nichtverfügbarkeit in unterschiedlichen Ausprägungen in Betracht, von einer einzelnen Availability Zone bis hin zu einer ganzen Region. Wenn Sie Daten und Ressourcen über Availability Zones hinweg replizieren und in einer einzelnen Availability Zone ein länger dauernder Ausfall auftritt, können Sie den Service in einer anderen Availability Zone wiederherstellen. Wenn Sie Daten und Ressourcen regionsübergreifend replizieren, können Sie den Service in einer anderen Region wiederherstellen.
3. Stellen Sie Ihre Computingressourcen in mehreren Availability Zones bereit.
 - a. Erstellen Sie in Ihrer VPC mehrere Subnetze in verschiedenen Availability Zones. Konfigurieren Sie jedes Subnetz so, dass es groß genug ist, um die Ressourcen aufzunehmen, die zur Bewältigung des Workloads benötigt werden, auch bei einem Vorfall. Weitere Informationen finden Sie unter [REL02-BP03 Berücksichtigen von Erweiterung und Verfügbarkeit bei der Zuweisung von IP-Adressen für Subnetze](#).
 - b. Wenn Sie Amazon-EC2-Instances verwenden, verwenden Sie [EC2 Auto Scaling](#), um Ihre Instances zu verwalten. Geben Sie die Subnetze an, die Sie im vorherigen Schritt ausgewählt haben, als Sie Ihre Auto Scaling-Gruppen erstellt haben.
 - c. Wenn Sie AWS Fargate Compute für [Amazon ECS](#) oder [Amazon EKS](#) verwenden, wählen Sie die Subnetze aus, die Sie im ersten Schritt ausgewählt haben, wenn Sie einen ECS-Service erstellen, eine ECS-Aufgabe starten oder ein [Fargate-Profil](#) für EKS erstellen.

- d. Wenn Sie AWS Lambda-Funktionen verwenden, die in Ihrer VPC ausgeführt werden müssen, wählen Sie die Subnetze aus, die Sie im ersten Schritt bei der Erstellung der Lambda-Funktion ausgewählt haben. AWS Lambda verwaltet die Verfügbarkeit für alle Funktionen, die keine VPC-Konfiguration haben, automatisch für Sie.
 - e. Platzieren Sie Traffic Directors wie Load Balancer vor Ihren Computingressourcen. Wenn zonenübergreifendes Load Balancing aktiviert ist, erkennen [AWS Application Load Balancer](#) und [Network Load Balancer](#), wenn Ziele wie EC2-Instances und Container aufgrund einer Beeinträchtigung der Availability Zone nicht erreichbar sind, und leiten den Datenverkehr zu Zielen in intakten Availability Zones um. Wenn Sie das zonenübergreifende Load Balancing deaktivieren, verwenden Sie Amazon Application Recovery Controller (ARC), um die Zonenverschiebungsfunktion bereitzustellen. Wenn Sie einen Load Balancer eines Drittanbieters verwenden oder Ihre eigenen Load Balancer implementiert haben, konfigurieren Sie diese mit mehreren Front-Ends in verschiedenen Availability Zones.
4. Replizieren Sie die Daten Ihrer Workloads in mehrere Availability Zones.
- a. Wenn Sie einen AWS-verwalteten Datenservice wie Amazon RDS, Amazon ElastiCache oder Amazon FSx verwenden, lesen Sie dessen Benutzerhandbuch, um mehr über seine Datenreplikations- und Ausfallsicherheitsfunktionen zu erfahren. Aktivieren Sie bei Bedarf AZ-übergreifende Replikation und Failover.
 - b. Wenn Sie AWS-verwaltete Speicherservices wie Amazon S3, Amazon EFS und Amazon FSx verwenden, vermeiden Sie die Verwendung von Single-AZ- oder One-Zone-Konfigurationen für Daten, die eine hohe Dauerhaftigkeit erfordern. Verwenden Sie für diese Services eine Multi-AZ-Konfiguration. Prüfen Sie im Benutzerhandbuch des jeweiligen Services, ob die Multi-AZ-Replikation standardmäßig aktiviert ist, oder ob Sie sie aktivieren müssen.
 - c. Wenn Sie eine selbstverwaltete Datenbank, eine Warteschlange oder einen anderen Speicherservice ausführen, organisieren Sie die Multi-AZ-Replikation gemäß den Anweisungen zu der Anwendung oder bewährten Methoden. Machen Sie sich mit den Failover-Verfahren für Ihre Anwendung vertraut.
5. Konfigurieren Sie Ihren DNS-Service so, dass er AZ-Beeinträchtigungen erkennt und den Datenverkehr in eine fehlerfreie Availability Zone umleitet. Amazon Route 53 kann dies automatisch tun, wenn es in Kombination mit Elastic Load Balancers verwendet wird. Route 53 kann auch mit Failover-Datensätzen konfiguriert werden, bei denen mithilfe von Integritätsprüfungen nur Anfragen mit fehlerfreien IP-Adressen beantwortet werden. Geben Sie für alle DNS-Datensätze, die für das Failover verwendet werden, einen kleinen TTL-Wert (Time to Live) an (z. B. 60 Sekunden oder weniger), um zu verhindern, dass das Zwischenspeichern von

Einträgen die Wiederherstellung behindert (Route 53-Aliaseinträge stellen die entsprechenden TTLs für Sie bereit).

Zusätzliche Schritte bei Verwendung mehrerer AWS-Regionen

1. Replizieren Sie den gesamten Betriebssystem- (OS) und Anwendungscode, der von Ihren Workloads verwendet wird, in den ausgewählten Regionen. Replizieren Sie bei Bedarf Amazon Machine Images (AMIs), die von Ihren EC2-Instances verwendet werden, mithilfe von Lösungen wie Amazon EC2 Image Builder. Replizieren Sie Container-Images, die in Registern gespeichert sind, mithilfe von Lösungen wie der regionsübergreifenden Replikation in Amazon ECR. Aktivieren Sie die regionale Replikation für alle Amazon-S3-Buckets, die zum Speichern von Anwendungsressourcen verwendet werden.
2. Stellen Sie Ihre Computingressourcen und Konfigurationsmetadaten (z. B. als im AWS Systems Manager Parameter Store gespeicherte Parameter) in mehreren Regionen bereit. Verwenden Sie dieselben Verfahren, die in den vorherigen Schritten beschrieben wurden, replizieren Sie jedoch die Konfiguration für jede Region, die Sie für Ihre Workloads verwenden. Verwenden Sie Infrastructure-as-Code-Lösungen wie AWS CloudFormation, um beispielsweise die Konfigurationen in allen Regionen einheitlich zu reproduzieren. Wenn Sie eine sekundäre Region in einer Pilot-Light-Konfiguration für die Notfallwiederherstellung verwenden, können Sie die Anzahl Ihrer Computingressourcen auf einen Mindestwert reduzieren, um Kosten zu sparen, was die Zeit bis zur Wiederherstellung entsprechend verlängert.
3. Replizieren Sie Ihre Daten aus Ihrer primären Region in Ihre sekundären Regionen.
 - a. Globale Amazon DynamoDB-Tabellen stellen globale Replikate Ihrer Daten bereit, in die aus jeder unterstützten Region geschrieben werden kann. Bei anderen AWS-verwalteten Datenservices wie Amazon RDS, Amazon Aurora und Amazon ElastiCache legen Sie eine primäre (Lese-/Schreib-) Region und Replikatregionen (schreibgeschützt) fest. Einzelheiten zur regionalen Replikation finden Sie in den Benutzer- und Entwicklerhandbüchern der jeweiligen Services.
 - b. Wenn Sie eine selbstverwaltete Datenbank verwenden, richten Sie die Replikation in mehreren Regionen gemäß den Anweisungen zur Anwendung oder zu den bewährten Methoden ein. Machen Sie sich mit den Failover-Verfahren für Ihre Anwendung vertraut.
 - c. Wenn Ihre Workloads AWS EventBridge verwenden, müssen Sie möglicherweise ausgewählte Ereignisse aus Ihrer primären Region an Ihre sekundären Regionen weiterleiten. Geben Sie dazu Ereignisbusse in Ihren sekundären Regionen als Ziele für übereinstimmende Ereignisse in Ihrer Hauptregion an.

4. Überlegen Sie, ob und in welchem Umfang Sie in allen Regionen identische Verschlüsselungsschlüssel verwenden möchten. Ein typischer Ansatz, der Sicherheit und Benutzerfreundlichkeit in Einklang bringt, ist die Verwendung von regionsspezifischen Schlüsseln für die Region lokale Daten und Authentifizierung und die Verwendung von Schlüsseln mit globalem Geltungsbereich für die Verschlüsselung von Daten, die zwischen verschiedenen Regionen repliziert werden. [AWS Key Management Service \(KMS\)](#) unterstützt Schlüssel für [mehrere Regionen](#), um Schlüssel, die zwischen Regionen gemeinsam genutzt werden, sicher zu verteilen und zu schützen.
5. Ziehen Sie AWS Global Accelerator in Betracht, um die Verfügbarkeit Ihrer Anwendung zu verbessern, indem Sie den Datenverkehr in Regionen mit fehlerfreien Endpunkten umleiten.

Ressourcen

Zugehörige bewährte Methoden:

- [REL02-BP03 Berücksichtigen von Erweiterung und Verfügbarkeit bei der Zuweisung von IP-Adressen für Subnetze](#)
- [REL11-BP05 Verhindern von bimodalem Verhalten mithilfe statischer Stabilität](#)
- [REL13-BP01 Definieren von Wiederherstellungszielen bei Ausfällen und Datenverlusten](#)

Zugehörige Dokumente:

- [Globale AWS-Infrastruktur](#)
- [Whitepaper: Grenzen der AWS-Fehlerisolierung](#)
- [Ausfallsicherheit in Amazon EC2 Auto Scaling](#)
- [Amazon EC2 Auto Scaling: Beispiel: Aufteilen von Instances über mehrere Availability Zones](#)
- [So funktioniert EC2 Image Builder](#)
- [Wie Amazon ECS Aufgaben auf Container-Instances platziert \(einschließlich Fargate\)](#)
- [Ausfallsicherheit in AWS Lambda](#)
- [Amazon S3: Überblick über das Replizieren von Objekten](#)
- [Replikation privater Images in Amazon ECR](#)
- [Globale Tabellen: Multiregionale Replikation mit DynamoDB](#)
- [Amazon ElastiCache for Redis OSS: Replikation über AWS-Regionen hinweg mit globalen Datenspeichern](#)

- [Ausfallsicherheit in Amazon RDS](#)
- [Verwenden von Amazon Aurora Global Databases](#)
- [AWS Global Accelerator – Entwicklerhandbuch](#)
- [Multiregionale Schlüssel in AWS KMS](#)
- [Amazon Route 53: Konfiguration des DNS-Failovers](#)
- [Amazon Application Recovery Controller \(ARC\) – Entwicklerhandbuch](#)
- [Senden und Empfangen von Amazon-EventBridge-Ereignissen zwischen AWS-Regionen](#)
- [Blogserie „Erstellen einer regionsübergreifenden Anwendung mit AWS-Services“](#)
- [Notfallwiederherstellung \(DR\)-Architektur in AWS, Teil I: Strategien für die Wiederherstellung in der Cloud](#)
- [Architektur für die Notfallwiederherstellung in AWS, Teil III: Pilot Light und Warm Standby](#)

Zugehörige Videos:

- [AWS re:Invent 2018: Architecture Patterns for Multi-Region Active-Active Applications](#)
- [AWS re:Invent 2019: Innovation and operation of the AWS global network infrastructure](#)

REL10-BP03 Automatisierte Wiederherstellung für Komponenten, die auf einen einzelnen Standort beschränkt sind

Wenn Komponenten der Workload nur in einer einzigen Availability Zone oder in einem On-Premises-Rechenzentrum ausgeführt werden können, implementieren Sie die Möglichkeit, die Workload innerhalb Ihrer definierten Wiederherstellungsziele komplett neu aufzusetzen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Wenn die bewährte Methode zur Bereitstellung der Workload an mehreren Standorten aufgrund technologischer Einschränkungen nicht möglich ist, müssen Sie einen alternativen Pfad zur Ausfallsicherheit implementieren. Sie müssen die Möglichkeit automatisieren, die erforderliche Infrastruktur neu zu erstellen, Anwendungen neu bereitzustellen und die erforderlichen Daten für diese Fälle neu zu erstellen.

Amazon EMR startet beispielsweise alle Knoten für einen bestimmten Cluster in derselben Availability Zone, da die Ausführung eines Clusters in derselben Zone eine höhere Datenzugriffsrate bietet und

dadurch eine höhere Leistung für die Aufgabenbearbeitung bereitstellt. Wenn diese Komponente für die Ausfallsicherheit von Workloads erforderlich ist, müssen Sie die Möglichkeit haben, den Cluster und seine Daten erneut bereitzustellen. Für Amazon EMR sollten Sie nicht nur Multi-AZs verwenden, um für Redundanz zu sorgen. Sie können [mehrere Knoten](#) bereitstellen. Mit [EMR File System \(EMRFS\)](#) können Daten in EMR in Amazon S3 gespeichert und dann über mehrere Availability Zones oder AWS-Regionen repliziert werden.

Ähnlich wird Ihr Cluster bei Amazon Redshift standardmäßig in einer zufällig ausgewählten Availability Zone innerhalb der ausgewählten AWS-Region bereitgestellt. Alle Knoten des Clusters werden in derselben Zone bereitgestellt.

Für zustandsbehaftete serverbasierte Workloads, die in einem On-Premises-Rechenzentrum bereitgestellt werden, können Sie AWS Elastic Disaster Recovery verwenden, um Ihre Workloads in AWS zu schützen. Wenn Sie bereits in AWS gehostet sind, können Sie Elastic Disaster Recovery verwenden, um Ihre Workload in einer alternativen Availability Zone oder Region zu schützen. Elastic Disaster Recovery verwendet eine kontinuierliche Replikation auf Block-Ebene in eine schlanke Staging-Area, um eine schnelle, zuverlässige Wiederherstellung von On-Premises-Anwendungen und cloudbasierten Anwendungen zu gewährleisten.

Implementierungsschritte

1. Implementieren Sie die Selbstreparatur. Stellen Sie Ihre Instances oder Container nach Möglichkeit mit automatischer Skalierung bereit. Wenn dies nicht möglich ist, nutzen Sie für EC2-Instances die automatische Wiederherstellung oder implementieren Sie eine automatische Selbstreparatur basierend auf Amazon EC2- oder ECS-Container-Lebenszyklusereignissen.
 - Verwenden Sie [Amazon EC2-Auto-Scaling-Gruppen](#) für Instances und Container-Workloads, die keine IP-Adresse für eine einzelne Instance, keine private IP-Adresse, keine elastische IP-Adresse und keine Instance-Metadaten benötigen.
 - Die Benutzerdaten der Startvorlage können zur Implementierung einer Automatisierung verwendet werden, die die meisten Workloads automatisch reparieren kann.
 - Verwenden Sie die automatische [Wiederherstellung von Amazon EC2-Instances](#) für Workloads, die eine IP-Adresse für eine einzelne Instance, eine private IP-Adresse, eine elastische IP-Adresse und Instance-Metadaten benötigen.
 - Automatic Recovery sendet Benachrichtigungen zum Wiederherstellungsstatus an ein SNS-Thema, wenn der Instance-Fehler erkannt wird.

- Verwenden Sie [Amazon EC2-Instanzlebenszyklusereignisse](#) bzw. [Amazon ECS-Ereignisse](#) für die Automatisierung der Selbstreparatur, wenn die automatische Skalierung oder EC2-Wiederherstellung nicht verwendet werden kann.
- Verwenden Sie die Ereignisse, um die Automatisierung der Reparatur der Komponente entsprechend der erforderlichen Prozesslogik aufzurufen.
- Schützen Sie statusbehaftete Workloads, die auf einen einzelnen Standort beschränkt sind, mithilfe von [AWS Elastic Disaster Recovery](#).

Ressourcen

Zugehörige Dokumente:

- [Amazon ECS-Ereignisse](#)
- [Lebenszyklus-Hooks bei Amazon EC2 Auto Scaling](#)
- [Wiederherstellen der Instance.](#)
- [Automatische Skalierung von Services](#)
- [Was ist Amazon EC2 Auto Scaling?](#)
- [AWS Elastic Disaster Recovery](#)

REL10-BP04 Verwenden von Bulkhead-Architekturen, um den Umfang von Beeinträchtigungen zu begrenzen

Implementieren Sie Bulkhead-Architekturen (zellenbasierte Architekturen), um die Auswirkungen von Fehlern innerhalb einer Workload auf eine begrenzte Anzahl von Komponenten zu beschränken.

Gewünschtes Ergebnis: Eine zellenbasierte Architektur verwendet mehrere isolierte Instances einer Workload, wobei jede Instance als Zelle bezeichnet wird. Jede Zelle ist unabhängig. Sie teilt ihren Status nicht mit anderen Zellen und bearbeitet eine Teilmenge der gesamten Workload-Anfragen. Dadurch werden die möglichen Auswirkungen eines Fehlers, z. B. eines fehlerhaften Software-Updates, auf eine einzelne Zelle und die von ihr verarbeiteten Anfragen reduziert. Wenn in einer Workload 10 Zellen für die Beantwortung von 100 Anfragen verwendet werden, sind bei einem Fehler 90 % der gesamten Anfragen nicht davon betroffen.

Typische Anti-Muster:

- Es wird ein unbegrenztes Wachstum der Zellen zugelassen.

- Code-Updates oder Bereitstellungen werden auf alle Zellen gleichzeitig angewandt.
- Status oder Komponenten werden von den Zellen geteilt (mit Ausnahme der Router-Schicht).
- Es werden komplexe Geschäfts- oder Routing-Logiken in die Routing-Schicht eingefügt.
- Es gibt keine Minimierung der zellenübergreifenden Interaktionen.

Vorteile der Nutzung dieser bewährten Methode: Bei zellenbasierten Architekturen sind viele gängige Fehlertypen in der Zelle selbst enthalten, was eine zusätzliche Fehlerisolierung ermöglicht. Diese Fehlergrenzen können die Widerstandsfähigkeit gegenüber Fehlertypen erhöhen, die sonst schwer einzudämmen sind, wie z. B. erfolglose Codebereitstellungen oder Anfragen, die beschädigt sind oder einen bestimmten Fehlermodus aufrufen (auch bekannt als Poison-Pill-Anfragen).

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Auf einem Schiff sorgen Schotten dafür, dass eine Beschädigung des Rumpfes auf einen Teil des Schiffes beschränkt bleibt. In komplexen Systemen wird dieses Muster oft kopiert, um eine Fehlerisolierung zu ermöglichen. Fehlerisolierte Grenzen beschränken die Auswirkungen eines Fehlers innerhalb einer Workload auf eine begrenzte Anzahl von Komponenten. Komponenten außerhalb der Grenze sind von dem Ausfall nicht betroffen. Durch die Verwendung mehrerer fehlerisolierter Grenzen können Sie die Auswirkungen auf Ihre Workload einschränken. Bei AWS können Kunden mehrere Availability Zones und Regionen verwenden, um eine Fehlerisolierung zu gewährleisten. Das Konzept der Fehlerisolierung lässt sich jedoch auch auf die Architektur Ihrer Workload ausweiten.

Die gesamte Workload wird durch einen Partitionsschlüssel in Zellen unterteilt. Dieser Schlüssel muss mit der Struktur des Service übereinstimmen oder mit der natürlichen Art und Weise, wie die Workload eines Service mit minimalen zellenübergreifenden Interaktionen unterteilt werden kann. Beispiele für Partitionsschlüssel sind die ID des Kunden, die ID der Ressource oder jeder andere Parameter, der in den meisten API-Aufrufen leicht zugänglich ist. Eine Schicht für das Routing von Zellen verteilt Anfragen auf der Grundlage des Partitionsschlüssels an einzelne Zellen und präsentiert den Kunden einen einzigen Endpunkt.

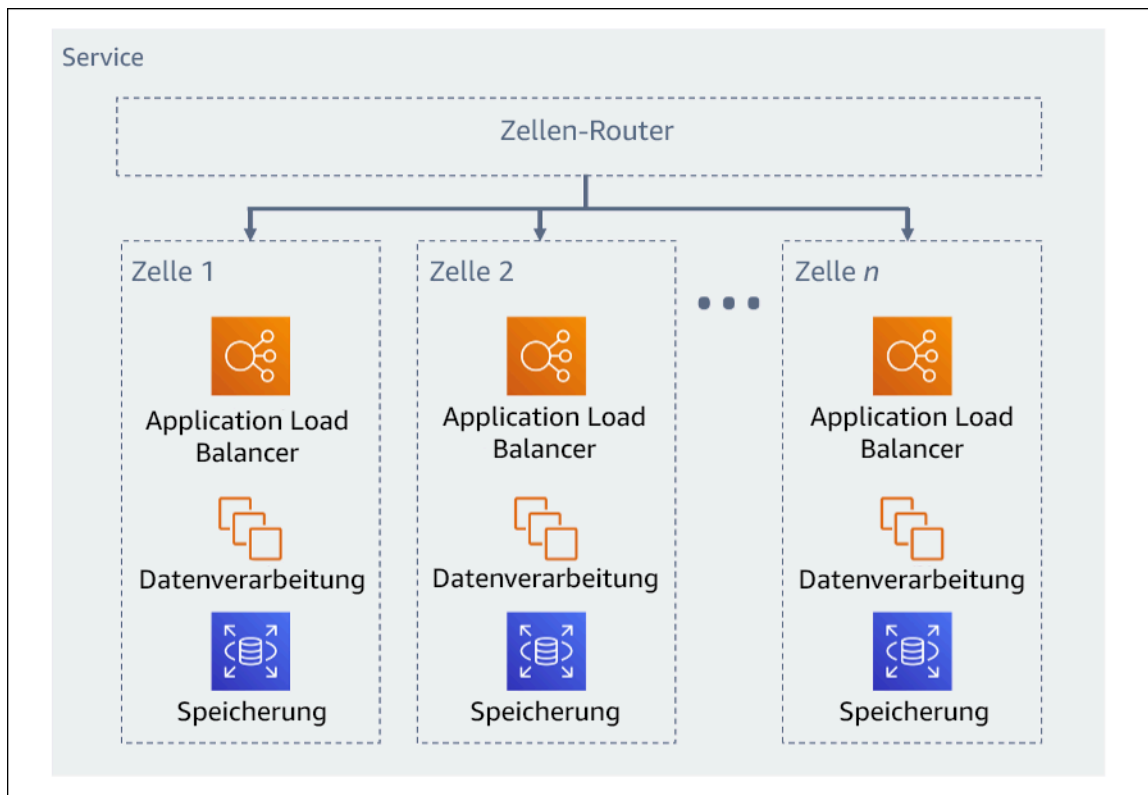


Abbildung 11: Zellenbasierte Architektur

Implementierungsschritte

Bei der Entwicklung einer zellenbasierten Architektur sind mehrere Designüberlegungen zu berücksichtigen:

1. Partitionsschlüssel: Bei der Auswahl des Partitionsschlüssels sollten besondere Überlegungen angestellt werden.
 - Er sollte mit der Struktur des Service übereinstimmen oder mit der natürlichen Art und Weise, wie die Workload eines Service mit minimalen zellenübergreifenden Interaktionen unterteilt werden kann. Beispiele sind `customer ID` oder `resource ID`.
 - Der Partitionsschlüssel muss in allen Anfragen verfügbar sein – entweder direkt oder in einer Weise, die sich durch andere Parameter leicht deterministisch ableiten lässt.
2. Persistente Zellenzuordnung: Upstream-Services sollten während des gesamten Lebenszyklus ihrer Ressourcen nur mit einer einzelnen Zelle interagieren.
 - Je nach Workload kann eine Strategie zur Migration von Zellen erforderlich sein, um Daten von einer Zelle in eine andere zu migrieren. Ein mögliches Szenario, in dem eine Migration von

Zellen erforderlich sein kann, ist, wenn ein bestimmter Benutzer oder eine bestimmte Ressource in Ihrer Workload zu groß wird und eine eigene Zelle benötigt.

- Zellen sollten keinen Status und keine Komponenten gemeinsam nutzen.
- Folglich sollten zellenübergreifende Interaktionen vermieden oder auf ein Minimum beschränkt werden, da diese Interaktionen Abhängigkeiten zwischen den Zellen schaffen und somit die Möglichkeiten zur Fehlerisolierung verringern.

3. Router-Schicht: Die Router-Schicht ist eine Komponente, die von Zellen gemeinsam genutzt wird, sodass nicht dieselbe Segmentierungsstrategie verfolgt werden kann wie bei Zellen.

- Es wird empfohlen, dass die Routing-Schicht Anfragen auf einzelne Zellen verteilt, indem sie einen effizienten Algorithmus für die Zuordnung von Partitionen einsetzt – z. B. als die Kombination von kryptographischen Hash-Funktionen und einer modularen Arithmetik.
- Um Auswirkungen auf mehrere Zellen zu vermeiden, muss die Routing-Schicht so einfach und horizontal skalierbar wie möglich bleiben, was den Verzicht auf eine komplexe Geschäftslogik innerhalb dieser Schicht erforderlich macht. Dies hat den zusätzlichen Nutzen, dass das erwartete Verhalten jederzeit leicht nachvollziehbar ist, was eine gründliche Testbarkeit ermöglicht. Wie Colm MacCárthaigh in [Zuverlässigkeit, stetige Ausführung und eine gute Tasse Kaffee](#) erklärt, sorgen einfache Designs und Muster mit konstanter Ausführung für zuverlässige Systeme und verringern die Widerstandsfähigkeit gegen Fragilität.

4. Zellgröße: Für Zellen sollte eine maximale Größe festgelegt sein, die sie nicht überschreiten dürfen.

- Die maximale Größe sollte durch gründliche Tests ermittelt werden – bis Sollbruchstellen erreicht und sichere operative Margen etabliert sind. Weitere Details zur Implementierung von Testverfahren finden Sie unter [REL07-BP04 Belastungstest Ihr Workload](#).
- Die gesamte Workload sollte durch Hinzufügen zusätzlicher Zellen wachsen, sodass die Workload mit der steigenden Nachfrage skalieren kann.

5. Multi-AZ- oder multiregionale Strategien: Zum Schutz vor unterschiedlichen Ausfall-Domains sollten mehrere Resilienzebenen genutzt werden.

- Für die Ausfallsicherheit sollten Sie einen Ansatz wählen, bei dem verschiedene Verteidigungsebenen aufgebaut werden. Eine Ebene schützt vor kleineren, häufiger auftretenden Unterbrechungen, indem eine hochverfügbare Architektur mit mehreren AZs erstellt wird. Eine weitere Verteidigungsebene schützt vor seltenen Ereignissen wie Naturkatastrophen mit großer Reichweite und Unterbrechungen auf Regionsebene. Für diese zweite Ebene muss die Architektur Ihrer Anwendung mehrere AWS-Regionen umfassen. Wenn Sie eine Multi-Region-Strategie für Ihre Workload implementieren, ist sie vor weitreichenden

Naturkatastrophen, die einen großen geografischen Bereich in einem Land betreffen, oder technischen Fehlern in einer ganzen Region geschützt. Beachten Sie dabei, dass das Implementieren einer Multi-Region-Architektur äußerst komplex sein kann und bei den meisten Workloads nicht erforderlich ist. Weitere Details erhalten Sie unter [REL10-BP01 Bereitstellen des Workloads an mehreren Standorten](#).

6. Codebereitstellung: Eine Strategie zur gestaffelten Codebereitstellung sollte der gleichzeitigen Bereitstellung von Codeänderungen in allen Zellen vorgezogen werden.
- Auf diese Weise werden mögliche Fehler in mehreren Zellen aufgrund einer fehlerhaften Bereitstellung oder menschlichen Versagens minimiert. Weitere Informationen finden Sie unter [Automatisierung sicherer, vollautomatischer Bereitstellungen](#).

Ressourcen

Zugehörige bewährte Methoden:

- [REL07-BP04 Belastungstest Ihr Workload](#)
- [REL10-BP01 Bereitstellen des Workloads an mehreren Standorten](#)

Zugehörige Dokumente:

- [Zuverlässigkeit, stetige Ausführung und eine gute Tasse Kaffee](#)
- [AWS und Segmentierung](#)
- [Workload-Isolation mit Shuffle Sharding](#)
- [Automatisierung sicherer, vollautomatischer Bereitstellungen](#)

Zugehörige Videos:

- [AWS re:Invent 2018: Close Loops & Opening Minds: Wie man die Kontrolle über große und kleine Systeme übernimmt](#)
- [AWS re:Invent 2018: How AWS Minimizes the Blast Radius of Failures \(ARC338\)](#)
- [Shuffle-Sharding: AWS re:Invent 2019: Einführung in die Amazon Builders' Library \(DOP328\)](#)
- [AWS Summit ANZ 2021 – Everything fails, all the time: Designing for resilience](#)

REL 11. Wie können Sie Workloads so gestalten, dass sie Komponentenausfällen gegenüber resilient sind?

Workloads, für die eine hohe Verfügbarkeit und eine niedrige mittlere Reparaturzeit (MTTR) erforderlich ist, müssen auf Ausfallsicherheit ausgelegt sein.

Best Practices

- [REL11-BP01 Überwachen aller Komponenten der Workload auf Fehler](#)
- [REL11-BP02 Failover zu fehlerfreien Ressourcen](#)
- [REL11-BP03 Automatisieren der Reparatur auf allen Ebenen](#)
- [REL11-BP04 Nutzen der Datenebene und nicht der Steuerebene während der Wiederherstellung](#)
- [REL11-BP05 Verhindern von bimodalem Verhalten mithilfe statischer Stabilität](#)
- [REL11-BP06 Senden von Benachrichtigungen, wenn sich Ereignisse auf die Verfügbarkeit auswirken](#)
- [REL11-BP07 Architektur Ihres Produkts zur Erfüllung von Verfügbarkeitszielen und Uptime-SLAs \(Service Level Agreements\)](#)

REL11-BP01 Überwachen aller Komponenten der Workload auf Fehler

Überwachen Sie den Zustand Ihrer Workload kontinuierlich, damit Sie und Ihre automatisierten Systeme auf Fehler oder Verschlechterungen aufmerksam werden, sobald diese auftreten.

Überwachen Sie Key Performance Indicators (KPIs, wichtige Leistungskennzahlen) auf Grundlage des geschäftlichen Wertes.

Alle Wiederherstellungs- und Reparaturmechanismen müssen auf eine schnelle Erkennung von Problemen ausgelegt sein. Technische Fehler sollten zuerst erkannt werden, damit sie behoben werden können. Die Verfügbarkeit basiert jedoch auf der Fähigkeit Ihrer Workload, einen Unternehmenswert zu liefern. Daher müssen wichtige Leistungskennzahlen (KPIs), die dies messen, in Ihre Erkennungs- und Behebungsstrategie integriert sein.

Gewünschtes Ergebnis: Wesentliche Komponenten einer Workload werden unabhängig überwacht, um Fehler zu erkennen und anzuzeigen, wann und wo sie auftreten.

Typische Anti-Muster:

- Es sind keine Alarme konfiguriert, sodass Ausfälle ohne Benachrichtigung auftreten.

- Alarme sind vorhanden, aber mit Schwellenwerten, die keine ausreichende Zeit für die Reaktion bieten.
- Metriken werden nicht häufig genug erfasst, um das Recovery Time Objective (RTO) zu erreichen.
- Nur die kundenorientierten Schnittstellen der Workload werden aktiv überwacht.
- Es werden nur technische Metriken erfasst, keine Metriken für Geschäftsfunktionen.
- Es gibt keine Metriken, die die Benutzererfahrung der Workload messen.
- Es werden zu viele Überwachungen erstellt.

Vorteile der Nutzung dieser bewährten Methode: Mit einer angemessenen Überwachung auf allen Ebenen können Sie die Wiederherstellungszeit reduzieren, indem Sie die Zeit bis zur Erkennung verkürzen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Identifizieren Sie alle Workloads, die für die Überwachung überprüft werden sollen. Sobald Sie alle zu überwachenden Komponenten der Workload identifiziert haben, müssen Sie das Überwachungsintervall festlegen. Das Überwachungsintervall wirkt sich direkt darauf aus, wie schnell eine Wiederherstellung eingeleitet werden kann (abhängig davon, wie lange die Erkennung eines Fehlers dauert). Die Mittlere Zeit bis zur Erkennung ist die Zeitspanne zwischen dem Auftreten eines Fehlers und dem Beginn der Reparaturarbeiten. Die Liste der Services sollte umfassend und vollständig sein.

Die Überwachung muss alle Ebenen des Anwendungs-Stacks (inklusive Anwendung, Plattform, Infrastruktur und Netzwerk) abdecken.

Ihre Überwachungsstrategie sollte außerdem die Auswirkungen von grauen Fehlern berücksichtigen. Weitere Informationen zu grauen Fehlern finden Sie unter [Graue Fehler](#) im Whitepaper „Erweiterte Multi-AZ Resilience-Muster“.

Implementierungsschritte

- Das Überwachungsintervall hängt davon ab, wie schnell Wiederherstellungen durchgeführt werden müssen. Die Wiederherstellungszeit hängt davon ab, wie viel Zeit für eine Wiederherstellung benötigt wird. Daher müssen Sie die Häufigkeit der Erfassung bestimmen, indem Sie diese Zeit und das RTO einkalkulieren.

- Konfigurieren Sie eine detaillierte Überwachung für Komponenten und verwaltete Services.
 - Legen Sie fest, ob eine [detaillierte Überwachung für EC2-Instances](#) und [Auto Scaling](#) erforderlich ist. Eine detaillierte Überwachung liefert Metriken in einminütigen Intervallen, die Standardüberwachung liefert Metriken in fünfminütigen Intervallen.
 - Legen Sie fest, ob eine [erweiterte Überwachung](#) für RDS notwendig ist. Die erweiterte Überwachung verwendet einen Agenten auf RDS-Instances, um nützliche Informationen über verschiedene Prozesse oder Threads zu erhalten.
 - Ermitteln Sie die Überwachungsanforderungen kritischer Serverless-Komponenten für [Lambda](#), [API Gateway](#), [Amazon EKS](#), [Amazon ECS](#) und alle Arten von [Load Balancern](#).
 - Ermitteln Sie die Überwachungsanforderungen der Speicherkomponenten für [Amazon S3](#), [Amazon FSx](#), [Amazon EFS](#) und [Amazon EBS](#).
- Erstellen Sie [benutzerdefinierte Metriken](#), um Leistungskennzahlen (KPIs) zu messen. Workloads implementieren wichtige geschäftliche Funktionen, die als KPIs verwendet werden sollten, um zu erkennen, wann ein indirektes Problem auftritt.
- Überwachen Sie das Benutzererlebnis mithilfe von Benutzer-Canarys auf Fehler. [Synthetische Transaktionstests](#) (auch bekannt als Canary-Tests, aber nicht zu verwechseln mit Canary-Bereitstellungen), die das Kundenverhalten simulieren können, gehören zu den wichtigsten Testprozessen. Führen Sie diese Tests für Ihre Workload-Endpunkte konstant von verschiedenen Remote-Standorten aus.
- Erstellen Sie [benutzerdefinierte Metriken](#) zur Verfolgung des Benutzererlebnisses. Wenn Sie das Kundenerlebnis instrumentieren können, können Sie die Verschlechterung des Kundenerlebnisses feststellen.
- [Richten Sie Alarmer ein](#), um zu erkennen, wenn ein Teil Ihrer Workload nicht ordnungsgemäß funktioniert, und um anzuzeigen, wann die Ressourcen automatisch skaliert werden müssen. Alarmer können visuell auf Dashboards angezeigt werden, Warnungen über Amazon SNS oder E-Mail versenden und mit Auto Scaling zusammenarbeiten, um Workload-Ressourcen hoch- oder herunterskalieren zu können.
- Erstellen Sie [Dashboards](#), um Ihre Metriken zu visualisieren. Dashboards können verwendet werden, um Trends, Ausreißer und andere Indikatoren für potenzielle Probleme zu visualisieren, und auf Probleme hinweisen, die Sie untersuchen sollten.
- Erstellen Sie eine [verteilte Ablaufverfolgungsüberwachung](#) für Ihre Services. Mit der verteilten Überwachung können Sie nachvollziehen, wie Ihre Anwendung und die ihr zugrunde liegenden Services arbeiten, um die Ursache von Leistungsproblemen und Fehlern zu identifizieren und zu beheben.

- Erstellen Sie Dashboards und Datenerfassung für Überwachungssysteme (mit [CloudWatch](#) oder [X-Ray](#)) in einer separaten Region und einem separaten Konto.
- Bleiben Sie mit [AWS Health](#) über Serviceeinbußen auf dem Laufenden. [Erstellen Sie maßgeschneiderte AWS Health-Ereignisbenachrichtigungen](#) für E-Mail- und Chat-Kanäle über [AWS-Benutzerbenachrichtigungen](#), und integrieren Sie sie programmgesteuert in [Ihre Überwachungs- und Warnmeldungs-Tools über Amazon EventBridge](#).

Ressourcen

Zugehörige bewährte Methoden:

- [Definition der Verfügbarkeit](#)
- [REL11-BP06 Senden von Benachrichtigungen, wenn sich Ereignisse auf die Verfügbarkeit auswirken](#)

Zugehörige Dokumente:

- [Amazon CloudWatch Synthetics unterstützt Sie bei der Erstellung von Benutzer-Canarys](#)
- [Aktivieren oder Deaktivieren der detaillierten Überwachung für Ihre Instance](#)
- [Verbesserte Überwachung](#)
- [Überwachung Ihrer Auto-Scaling-Gruppen und -Instances mithilfe von Amazon CloudWatch](#)
- [Veröffentlichen von benutzerdefinierten Metriken](#)
- [Verwenden von Amazon CloudWatch Alarms](#)
- [Verwenden von CloudWatch Dashboards](#)
- [Verwenden von regionen- und kontoübergreifenden Amazon CloudWatch-Dashboards](#)
- [Verwenden der regionen- und kontoübergreifenden X-Ray-Nachverfolgung](#)
- [Verstehen der Verfügbarkeit](#)

Zugehörige Videos:

- [Beheben von grauen Fehlern](#)

Zugehörige Beispiele:

- [Workshop zur Beobachtbarkeit: X-Ray erkunden](#)

Zugehörige Tools:

- [CloudWatch](#)
- [CloudWatch X-Ray](#)

REL11-BP02 Failover zu fehlerfreien Ressourcen

Wenn ein Fehler bei einer Ressource auftritt, sollten intakte Ressourcen weiterhin Anfragen bedienen. Stellen Sie sicher, dass Sie bei Standortbeeinträchtigungen (z. B. Availability Zone oder AWS-Region) über Systeme verfügen, die einen Failover auf intakte Ressourcen an nicht beeinträchtigten Standorten ermöglichen.

Wenn Sie einen Service entwerfen, verteilen Sie die Last auf Ressourcen, Availability Zones oder Regionen. So kann der Fehler einer einzelnen Ressource oder eine Beeinträchtigung durch die Verlagerung des Datenverkehrs auf die verbleibenden intakten Ressourcen aufgefangen werden. Überlegen Sie, wie Services im Falle eines Fehlers erkannt und geroutet werden.

Entwerfen Sie Ihre Services mit Blick auf die Fehlerbehebung. Bei AWS konzipieren wir Services mit dem Ziel, die Wiederherstellungszeit nach Ausfällen und die Auswirkungen auf Daten zu minimieren. Unsere Services verwenden primär Datenspeicher, die Anfragen erst akzeptieren, nachdem sie dauerhaft auf mehreren Replikaten in einer Region gespeichert wurden. Sie sind so aufgebaut, dass sie eine zellenbasierte Isolation und die Fehlerisolierung von Availability Zones nutzen. In unseren betrieblichen Abläufen setzen wir sehr stark auf Automatisierung. Außerdem optimieren wir unsere Funktionalität für Ersetzungsvorgänge und Neustarts, um nach Unterbrechungen eine schnelle Wiederherstellung zu ermöglichen.

Die Muster und Entwürfe, die den Failover ermöglichen, variieren für jeden AWS-Plattform-Service. Viele native verwaltete Services von AWS nutzen von Haus aus mehrere Availability Zones (wie Lambda oder API Gateway). Andere AWS-Services (wie EC2 und EKS) erfordern spezielle bewährte Methoden, um einen Failover von Ressourcen oder Datenspeichern über AZs hinweg zu unterstützen.

Es sollte eine Überwachung eingerichtet werden, um zu überprüfen, ob die Failover-Ressource in Ordnung ist, den Fortschritt der Failover-Ressourcen zu verfolgen und die Wiederherstellung von Geschäftsprozessen zu überwachen.

Gewünschtes Ergebnis: Die Systeme sind in der Lage, automatisch oder manuell neue Ressourcen zu verwenden, um sich von Störungen zu erholen.

Typische Anti-Muster:

- Die Planung für Fehler ist nicht Teil der Planungs- und Designphase.
- RTO und RPO sind nicht festgelegt.
- Unzureichende Überwachung, um ausfallende Ressourcen zu erkennen.
- Ordnungsgemäße Isolierung von fehlerhaften Domains.
- Multi-Region-Failover wird nicht berücksichtigt.
- Die Erkennung von Fehlern ist bei der Entscheidung für einen Failover zu empfindlich oder zu aggressiv.
- Failover-Design wird nicht getestet oder validiert.
- Durchführen automatischer Reparaturen ohne die Benachrichtigung, dass eine Reparatur erforderlich war.
- Fehlender Ausgleichszeitraum, um einen zu frühen Failover zu vermeiden.

Vorteile der Nutzung dieser bewährten Methode: Sie können widerstandsfähigere Systeme aufbauen, die auch bei Fehlern zuverlässig bleiben, indem sie ordnungsgemäß reduziert werden und sich schnell erholen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

AWS-Services wie [Elastic Load Balancing](#) und [Amazon EC2 Auto Scaling](#) helfen dabei, die Last auf Ressourcen und Availability Zones zu verteilen. Daher können der Ausfall einer einzelnen Ressource (wie etwa einer EC2-Instance) oder die Beeinträchtigung einer Availability Zone gemindert werden, indem Datenverkehr verlagert wird, um Ressourcen fehlerfrei zu halten.

Bei Workloads, die mehrere Regionen umfassen, sind Designs etwas komplizierter. Mit regionenübergreifenden Lesereplikaten können Sie beispielsweise Ihre Daten für mehrere AWS-Regionen bereitstellen. Der Failover ist jedoch immer noch erforderlich, um das Lesereplikat zum primären Endpunkt zu machen und den Datenverkehr auf den neuen Endpunkt zu lenken. Amazon Route 53, [Amazon Application Recovery Controller \(ARC\)](#), Amazon CloudFront und AWS Global Accelerator können dabei helfen, den Datenverkehr über AWS-Regionen hinweg weiterzuleiten.

AWS-Services wie Amazon S3, Lambda, API Gateway, Amazon SQS, Amazon SNS, Amazon SES, Amazon Pinpoint, Amazon ECR, AWS Certificate Manager, EventBridge und Amazon DynamoDB

werden von AWS automatisch in mehreren Availability Zones bereitgestellt. Im Falle eines Fehlers leiten diese AWS-Services den Datenverkehr automatisch an intakte Standorte um. Die Daten werden redundant in mehreren Availability Zones gespeichert und bleiben verfügbar.

Für Amazon RDS, Amazon Aurora, Amazon Redshift, Amazon EKS und Amazon ECS ist Multi-AZ eine Konfigurationsoption. AWS kann den Datenverkehr an die fehlerfreie Instance weiterleiten, wenn ein Failover eingeleitet wird. Diese Failover-Aktion kann von AWS oder auf Wunsch des Kunden durchgeführt werden.

Wählen Sie für Amazon-EC2-Instances, Amazon Redshift, Amazon-ECS-Aufgaben oder Amazon-EKS-Pods die Availability Zones für die Bereitstellung aus. Bei einigen Designs bietet Elastic Load Balancing die Lösung, um Instances in fehlerhaften Zonen zu erkennen und den Datenverkehr an die fehlerfreien Zonen weiterzuleiten. Elastic Load Balancing kann den Datenverkehr auch an Komponenten in Ihrem On-Premises-Rechenzentrum weiterleiten.

Für den Failover von Datenverkehr aus mehreren Regionen kann das Rerouting mit Amazon Route 53, Route 53 ARC, AWS Global Accelerator, Route 53 Private DNS für VPCs oder CloudFront eine Möglichkeit bieten, Internetdomains zu definieren und Routing-Richtlinien einschließlich Zustandsprüfungen zuzuweisen, um den Datenverkehr in intakte Regionen zu leiten. AWS Global Accelerator stellt statische IP-Adressen bereit, die als fester Einstiegspunkt für Ihre Anwendung fungieren und dann zu Endpunkten Ihrer Wahl in AWS-Regionen weitergeleitet werden, wobei das globale Netzwerk von AWS anstelle des Internets für eine bessere Leistung und Zuverlässigkeit genutzt wird.

Implementierungsschritte

- Erstellen Sie Failover-Designs für alle entsprechenden Anwendungen und Services. Isolieren Sie jede Komponente der Architektur und erstellen Sie Failover-Designs, die das RTO und RPO für jede Komponente erfüllen.
- Konfigurieren Sie weniger anspruchsvolle Umgebungen (wie Entwicklungs- oder Testumgebungen) mit allen Services, die für einen Failover-Plan erforderlich sind. Stellen Sie die Lösungen mit Infrastructure as Code (IaC) bereit, um die Reproduzierbarkeit sicherzustellen.
- Konfigurieren Sie einen Wiederherstellungsstandort, z. B. eine zweite Region, um die Failover-Designs zu implementieren und zu testen. Falls erforderlich, können die Ressourcen für die Tests vorübergehend konfiguriert werden, um die zusätzlichen Kosten zu begrenzen.
- Bestimmen Sie, welche Failover-Pläne durch AWS automatisiert sind, welche durch einen DevOps-Prozess automatisiert werden können und welche möglicherweise manuell sind. Dokumentieren und messen Sie die RTO- und RPO-Zeiten der einzelnen Services.

- Erstellen Sie ein Failover-Playbook, das alle Schritte zum Failover jeder Ressource, Anwendung und jedes Services enthält.
- Erstellen Sie ein Failback-Playbook, das alle Schritte zum Failback (mit Zeitangabe) für jede Ressource, jede Anwendung und jeden Service enthält.
- Erstellen Sie einen Plan, um das Playbook zu initiieren und zu proben. Verwenden Sie Simulationen und Chaostests, um die Schritte des Playbooks und die Automatisierung zu testen.
- Stellen Sie sicher, dass Sie bei einer Beeinträchtigung des Standorts (z. B. Availability Zone oder AWS-Region) über Systeme verfügen, die einen Failover auf intakte Ressourcen an nicht beeinträchtigten Standorten ermöglichen. Überprüfen Sie Kontingente, die automatische Skalierung und laufende Ressourcen vor dem Failover-Test.

Ressourcen

Zugehörige bewährte Methoden für Well-Architected:

- [REL13 – Planen für DR](#)
- [REL10 – Schützen von Workloads durch Fehlerisolierung](#)

Zugehörige Dokumente:

- [Einstellen von RTO- und RPO-Zielen](#)
- [Failover mithilfe von gewichtetem Route 53-Routing](#)
- [Notfallwiederherstellung mit Amazon Application Recovery Controller](#)
- [EC2 mit automatischer Skalierung](#)
- [EC2-Bereitstellungen – Multi-AZ](#)
- [ECS-Bereitstellungen – Multi-AZ](#)
- [Wechseln des Datenverkehrs mithilfe von Amazon Application Recovery Controller](#)
- [Lambda mit einem Application Load Balancer und Failover](#)
- [ACM-Replikation und -Failover](#)
- [Parameter Store-Replikation und -Failover](#)
- [Regionsübergreifende ECR-Replikation und Failover](#)
- [Konfigurieren der regionsübergreifenden Replikation von Secrets Manager](#)
- [Aktivieren der regionsübergreifenden Replikation für EFS und Failover](#)

- [Regionsübergreifende EFS-Replikation und Failover](#)
- [Netzwerk-Failover](#)
- [S3-Endpunkt-Failover mit MRAP](#)
- [Erstellen einer regionsübergreifenden Replikation für S3](#)
- [Anleitung für regionsübergreifendes Failover und Graceful Failback auf AWS](#)
- [Failover mit Global Accelerator über mehrere Regionen](#)
- [Failover mit DRS](#)

Zugehörige Beispiele:

- [Notfallwiederherstellung in AWS](#)
- [Elastische Notfallwiederherstellung in AWS](#)

REL11-BP03 Automatisieren der Reparatur auf allen Ebenen

Verwenden Sie bei Erkennung eines Fehlers automatisierte Funktionen, um Maßnahmen zur Behebung durchzuführen. Beeinträchtigungen können automatisch durch interne Service-Mechanismen behoben werden. Es kann aber auch erforderlich sein, Ressourcen neu zu starten oder Abhilfemaßnahmen durchzuführen.

Für selbstverwaltete Anwendungen und regionenübergreifende Korrekturen können Wiederherstellungskonzepte und automatisierte Korrekturprozesse aus [bestehenden bewährten Methoden](#) verwendet werden.

Die Möglichkeit, eine Ressource neu zu starten oder zu entfernen, ist ein wichtiges Instrument zur Behebung von Fehlern. Eine bewährte Methode besteht darin, Services nach Möglichkeit zustandslos zu betreiben. Dies verhindert den Datenverlust oder den Verlust der Verfügbarkeit bei einem Neustart der Ressource. In der Cloud können Sie (und sollten Sie üblicherweise) die gesamte Ressource (z. B. eine Datenverarbeitungs-Instance oder eine Serverless-Funktion) im Rahmen des Neustarts ersetzen. Der Neustart selbst ist eine einfache und zuverlässige Methode zur Wiederherstellung nach einem Ausfall. Bei Workloads treten viele verschiedene Arten von Fehlern auf. Fehler können bei Hardware, Software, Kommunikation und Betrieb auftreten.

Der Neustart oder Wiederholungsversuch gilt auch für Netzwerkanfragen. Nutzen Sie denselben Wiederherstellungsansatz für eine Netzwerk-Zeitüberschreitung und einen Abhängigkeitsfehler, bei dem die Abhängigkeit einen Fehler ausgibt. Beide Ereignisse wirken sich in ähnlicher

Weise auf das System aus. Statt also zu versuchen, aus den einzelnen Ereignissen einen „Sonderfall“ zu konstruieren, sollten Sie eine ähnliche Strategie anwenden und versuchen, ein exponentielles Backoff mit Jitter durchzuführen. Die Fähigkeit zum Neustart ist eine Funktion, die in wiederherstellungsorientierten Datenverarbeitungs- und Hochverfügbarkeits-Cluster-Architekturen empfohlen wird.

Gewünschtes Ergebnis: Automatisierte Aktionen werden durchgeführt, um die Erkennung eines Fehlers zu beheben.

Typische Anti-Muster:

- Bereitstellung von Ressourcen ohne automatische Skalierung.
- Einzelne Bereitstellung von Anwendungen in Instances oder Containern.
- Bereitstellen von Anwendungen, die nicht ohne automatische Wiederherstellung an mehreren Standorten bereitgestellt werden können.
- Manuelle Reparatur von Anwendungen, die sich mit Auto Scaling und einer automatischen Wiederherstellung nicht reparieren lassen.
- Keine Automatisierung beim Failover von Datenbanken.
- Keine automatisierten Methoden zur Umleitung des Datenverkehrs auf neue Endpunkte.
- Keine Speicherreplikation.

Vorteile der Nutzung dieser bewährten Methode: Eine automatisierte Korrektur kann die mittlere Zeit bis zur Wiederherstellung verkürzen und Ihre Verfügbarkeit verbessern.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Designs für Amazon EKS oder andere Kubernetes-Services sollten sowohl minimale und maximale Replikate oder zustandsbehaftete Sets als auch die minimale Größenanpassung von Clustern und Knotengruppen umfassen. Diese Mechanismen sorgen für ein Minimum an kontinuierlich verfügbaren Verarbeitungsressourcen und beheben gleichzeitig automatisch alle Fehler über die Steuerebene von Kubernetes.

Entwurfsmuster, auf die über einen Load Balancer mit Datenverarbeitungs-Clustern zugegriffen wird, sollten Auto-Scaling-Gruppen nutzen. Elastic Load Balancing (ELB) verteilt den eingehenden Datenverkehr von Anwendungen automatisch auf mehrere Ziele und virtuelle Appliances in einer oder mehreren Availability Zones (AZs).

Bei Cluster-Datenverarbeitungs-Instances, die kein Load Balancing nutzen, sollte die Größe für den Verlust von mindestens einem Knoten ausgelegt sein. Auf diese Weise kann der Service mit möglicherweise reduzierter Kapazität weiterlaufen, während er einen neuen Knoten wiederherstellt. Beispielservices sind Mongo, DynamoDB Accelerator, Amazon Redshift, Amazon EMR, Cassandra, Kafka, MSK-EC2, Couchbase, ELK und Amazon OpenSearch Service. Viele dieser Services können mit zusätzlichen Features zur Selbstheilung ausgestattet werden. Einige Cluster-Technologien müssen beim Verlust eines Knotens einen Alarm generieren, der einen automatisierten oder manuellen Workflow zur Wiederherstellung eines neuen Knotens auslöst. Dieser Workflow kann mit AWS Systems Manager automatisiert werden, um Probleme schnell zu beheben.

Mit Amazon EventBridge können Ereignisse wie CloudWatch-Alarme oder Statusänderungen in anderen AWS-Services überwacht und gefiltert werden. Auf der Grundlage von Ereignisinformationen kann es dann AWS Lambda, Systems Manager Automation oder andere Ziele aufrufen, um eine angepasste Wiederherstellungslogik für Ihre Workload auszuführen. Amazon EC2 Auto Scaling kann dafür konfiguriert werden, den Zustand der EC2-Instance zu prüfen. Wenn sich die Instance nicht im ausgeführten Status befindet oder der Systemstatus beeinträchtigt ist, betrachtet Amazon EC2 Auto Scaling die Instance als fehlerhaft und startet eine Ersatz-Instance. Bei Large-Scale-Ersetzungen (z. B. dem Verlust einer ganzen Availability Zone) ist für eine Hochverfügbarkeit die statische Stabilität vorzuziehen.

Implementierungsschritte

- Verwenden Sie Auto-Scaling-Gruppen, um Tiers in einem Workload bereitzustellen. [Auto Scaling](#) kann zustandslose Anwendungen selbst reparieren und Kapazitäten hinzufügen oder entfernen.
- Verwenden Sie [Load Balancing](#) für die zuvor genannten Datenverarbeitungs-Instances und wählen Sie den entsprechenden Load Balancer-Typ aus.
- Erwägen Sie die Reparatur für Amazon RDS. Konfigurieren Sie bei Standby-Instances das [automatische Failover](#) zur Standby-Instance. Bei Amazon RDS-Lesereplikaten ist ein automatisierter Workflow erforderlich, um ein Lesereplikat zur primären Instance zu machen.
- Implementieren Sie eine [automatische Wiederherstellung für EC2-Instances](#), in denen Anwendungen bereitgestellt werden, die nicht an mehreren Standorten bereitgestellt werden können, und die einen Neustart nach Ausfällen tolerieren. Mithilfe der automatischen Wiederherstellung kann ausgefallene Hardware ersetzt und die Instance neu gestartet werden, wenn die Anwendung sich nicht an mehreren Standorten bereitstellen lässt. Die Metadaten der Instance und die zugehörigen IP-Adressen werden beibehalten, ebenso wie die [EBS-Volumes](#) und Bindungspunkte für [Amazon Elastic File Systems](#) oder [Dateisysteme für Lustre](#) und [Windows](#). Mit

[AWS OpsWorks](#) können Sie die automatische Wiederherstellung von EC2-Instances auf Layer-Ebene konfigurieren.

- Implementieren Sie die automatische Wiederherstellung mit [AWS Step Functions](#) und [AWS Lambda](#), wenn Sie keine automatische Skalierung oder automatische Wiederherstellung verwenden können oder wenn die automatische Wiederherstellung fehlschlägt. Wenn Sie keine automatische Skalierung verwenden können und die automatische Wiederherstellung entweder nicht genutzt werden kann oder fehlschlägt, können Sie die Reparatur mithilfe von AWS Step Functions und AWS Lambda automatisieren.
- Mit [Amazon EventBridge](#) können Ereignisse wie [CloudWatch-Alarme](#) oder Statusänderungen in anderen AWS-Services überwacht und gefiltert werden. Auf der Grundlage von Ereignisinformationen kann es dann AWS Lambda (oder andere Ziele) aufrufen, um eine angepasste Wiederherstellungslogik für Ihre Workload auszuführen.

Ressourcen

Zugehörige bewährte Methoden:

- [Definition der Verfügbarkeit](#)
- [REL11-BP01 Überwachen aller Komponenten der Workload auf Fehler](#)

Zugehörige Dokumente:

- [Wie AWS Auto Scaling funktioniert](#)
- [Automatische Wiederherstellung in Amazon EC2](#)
- [Amazon Elastic Block Store \(Amazon EBS\)](#)
- [Amazon Elastic File System \(Amazon EFS\)](#)
- [Was ist Amazon FSx für Lustre?](#)
- [Was ist Amazon FSx für Windows File Server?](#)
- [AWS OpsWorks: Verwenden von Auto Healing zum Austausch fehlgeschlagener Instances](#)
- [Was ist AWS Step Functions?](#)
- [Was ist AWS Lambda?](#)
- [Was ist Amazon EventBridge?](#)
- [Verwenden von Amazon CloudWatch Alarms](#)
- [Amazon RDS-Failover](#)

- [SSM – Systems Manager Automation](#)
- [Bewährte Methoden für eine widerstandsfähige Architektur](#)

Zugehörige Videos:

- [Automatische Bereitstellung und Skalierung des OpenSearch-Services](#)
- [Automatisches Amazon RDS-Failover](#)

Zugehörige Beispiele:

- [Amazon RDS-Failover-Workshop](#)

Zugehörige Tools:

- [CloudWatch](#)
- [CloudWatch X-Ray](#)

REL11-BP04 Nutzen der Datenebene und nicht der Steuerebene während der Wiederherstellung

Steuerebenen stellen die administrativen APIs zum Erstellen, Lesen und Schreiben, Aktualisieren, Löschen und Auflisten (CRUDL) von Ressourcen bereit, während Datenebenen den normalen Datenverkehr des Services abwickeln. Konzentrieren Sie sich bei der Implementierung von Wiederherstellungs- oder Abhilfemaßnahmen für Ereignisse, die sich möglicherweise auf die Ausfallsicherheit auswirken, auf eine minimale Anzahl von Operationen auf der Steuerebene, um den Service wiederherzustellen, zu skalieren, zu reparieren oder einen Failover durchzuführen. Aktionen auf der Datenebene sollten während dieser Beeinträchtigungen Vorrang vor allen anderen Aktivitäten haben.

Die folgenden Aktionen gehören beispielsweise alle zur Steuerebene: Starten einer neuen Datenverarbeitungs-Instance, Erstellen von Block-Speicher und Beschreiben von Warteschlangen-Services. Wenn Sie Datenverarbeitungs-Instances starten, muss die Steuerebene mehrere Aufgaben erfüllen, z. B. einen physischen Host mit Kapazität finden, Netzwerkschnittstellen zuweisen, lokale Block-Speicher-Volumes vorbereiten, Anmeldeinformationen generieren und Sicherheitsregeln hinzufügen. Steuerebenen neigen zu einer komplizierten Orchestrierung.

Gewünschtes Ergebnis: Wenn bei einer Ressource eine Störung auftritt, ist das System in der Lage, diese automatisch oder manuell zu beheben, indem es den Datenverkehr von gestörten auf intakte Ressourcen umleitet.

Typische Anti-Muster:

- Abhängigkeit von der Änderung von DNS-Einträgen, um den Datenverkehr umzuleiten.
- Abhängigkeit von Skalierungsoperationen auf Steuerebene, um beeinträchtigte Komponenten aufgrund einer unzureichenden Bereitstellung von Ressourcen zu ersetzen.
- Abhängigkeit von umfangreichen Aktionen auf der Steuerebene, in die mehrere Services und APIs involviert sind, um Störungen jeglicher Art zu beheben.

Vorteile der Nutzung dieser bewährten Methode: Eine höhere Erfolgsquote bei der automatisierten Behebung kann Ihre mittlere Zeit bis zur Wiederherstellung verkürzen und die Verfügbarkeit des Workloads verbessern.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel: Bei bestimmten Arten von Service-Einschränkungen sind Steuerebenen betroffen. Die Abhängigkeit von einer umfassenden Nutzung der Steuerebene für die Behebung kann die Wiederherstellungszeit (RTO) und die mittlere Zeit bis zur Wiederherstellung (MTTR) erhöhen.

Implementierungsleitfaden

Um die Aktionen auf der Datenebene zu begrenzen, bewerten Sie für jeden Service, welche Aktionen zur Wiederherstellung des Services erforderlich sind.

Nutzen Sie Amazon Application Recovery Controller, um den DNS-Datenverkehr zu verschieben. Diese Features überwachen kontinuierlich die Fähigkeit Ihrer Anwendung, nach Fehlern wiederhergestellt zu werden, und ermöglichen es Ihnen, die Wiederherstellung Ihrer Anwendung über mehrere AWS-Regionen, Availability Zones und On-Premises zu steuern.

Route 53-Routingrichtlinien verwenden die Steuerebene. Verlassen Sie sich also bei der Wiederherstellung nicht auf diese Ebene. Die Route 53-Datenebenen beantworten DNS-Abfragen und führen Zustandsprüfungen durch und werten diese aus. Sie werden weltweit vertrieben und sind für ein [Service Level Agreement \(SLA\) mit 100-prozentiger Verfügbarkeit](#) konzipiert.

Die Route 53-Verwaltungs-APIs und -Konsolen, über die Sie Route 53-Ressourcen erstellen, aktualisieren und löschen, arbeiten auf Steuerebenen, die so konzipiert sind, dass die starke Konsistenz und Stabilität, die Sie beim Verwalten von DNS benötigen, Priorität haben. Zu diesem

Zweck befinden sich die Steuerebenen in einer einzelnen Region, USA Ost (Nord-Virginia). Beide Systeme sind zwar äußerst zuverlässig, aber die Steuerebenen sind nicht in der SLA enthalten. In seltenen Fällen kann es vorkommen, dass das ausfallsichere Design der Datenebene es ermöglicht, die Verfügbarkeit aufrechtzuerhalten, während die Steuerebene dies nicht tut. Verwenden Sie für die Notfallwiederherstellung und Failover-Mechanismen Datenebenen-Funktionen, um die bestmögliche Zuverlässigkeit bereitzustellen.

Entwerfen Sie Ihre Computing-Infrastruktur so, dass sie statisch stabil ist, um zu vermeiden, dass die Steuerebene während eines Vorfalls verwendet wird. Wenn Sie beispielsweise Amazon-EC2-Instances verwenden, vermeiden Sie es, neue Instances manuell bereitzustellen oder Auto-Scaling-Gruppen anzuweisen, als Reaktion darauf Instances hinzuzufügen. Für ein Höchstmaß an Ausfallsicherheit stellen Sie ausreichende Kapazitäten in dem für den Failover verwendeten Cluster bereit. Wenn diese Kapazität begrenzt werden muss, legen Sie Drosselungen für das gesamte System fest, um den Gesamtdatenverkehr an die beschränkte Ressourcenmenge sicher zu begrenzen.

Bei Services wie Amazon DynamoDB, Amazon API Gateway, Load Balancern und AWS Lambda Serverless wird die Datenebene für diese Services genutzt. Die Erstellung neuer Funktionen, Load Balancers, API-Gateways oder DynamoDB-Tabellen ist jedoch eine Aktion auf der Steuerebene und sollte vor der Störung als Vorbereitung auf ein Ereignis und zum Üben von Failover-Aktionen durchgeführt werden. Für Amazon RDS ermöglichen Aktionen auf der Datenebene den Zugriff auf Daten.

Weitere Informationen über Datenebenen, Steuerebenen und wie AWS Services aufbaut, um Hochverfügbarkeitsziele zu erfüllen, finden Sie unter [Statische Stabilität mithilfe von Availability Zones](#).

Erfahren Sie, welche Operationen auf der Datenebene und welche Operationen auf der Steuerebene ausgeführt werden.

Implementierungsschritte

Bewerten Sie für jede Workload, die nach einem Störfall wiederhergestellt werden muss, das Failover-Runbook, das Hochverfügbarkeitsdesign, das Auto Healing Design oder den Plan zur Wiederherstellung von HA-Ressourcen. Identifizieren Sie jede Aktion, die als Aktion auf der Steuerebene in Frage kommt.

Ziehen Sie in Erwägung, eine Aktion auf der Steuerebene in eine Aktion auf der Datenebene umzuwandeln:

- Auto Scaling (Steuerebene) im Vergleich zu vorab skalierten Amazon-EC2-Ressourcen (Datenebene)
- Instance-Skalierung mit Amazon EC2 (Steuerebene) im Vergleich zu AWS Lambda-Skalierung (Datenebene)
- Bewerten Sie alle Entwürfe unter Verwendung von Kubernetes und der Art der Aktionen auf der Steuerebene. Das Hinzufügen von Pods ist eine Aktion auf der Datenebene von Kubernetes. Aktionen sollten sich auf das Hinzufügen von Pods und nicht von Knoten beschränken. Die Verwendung von [überdimensionierten Knoten](#) ist die bevorzugte Methode zur Begrenzung von Aktionen auf der Steuerebene.

Ziehen Sie alternative Ansätze in Betracht, bei denen Aktionen auf der Datenebene dieselbe Maßnahme bewirken können.

- Änderung des Route-53-Datensatzes (Steuerebene) im Vergleich zu Amazon Application Recovery Controller (Datenebene)
- [Route 53-Zustandsprüfungen für weitere automatisierte Aktualisierungen](#)

Ziehen Sie einige Services in einer sekundären Region in Betracht, wenn der Service geschäftskritisch ist, um mehr Aktionen auf der Steuerebene und Datenebene in einer nicht betroffenen Region zu ermöglichen.

- Amazon EC2 Auto Scaling oder Amazon EKS in einer primären Region im Vergleich zu Amazon EC2 Auto Scaling oder Amazon EKS in einer sekundären Region und Weiterleitung des Datenverkehrs in eine sekundäre Region (Aktion auf der Steuerebene)
- Ein Lesereplikat in der sekundären primären Region erstellen oder Versuchen derselben Aktion in der primären Region (Aktion auf der Steuerebene)

Ressourcen

Zugehörige bewährte Methoden:

- [Definition der Verfügbarkeit](#)
- [REL11-BP01 Überwachen aller Komponenten der Workload auf Fehler](#)

Zugehörige Dokumente:

- [APN-Partner: Partner, die Sie bei der Automatisierung der Fehlertoleranz unterstützen können](#)
- [AWS Marketplace: Zur Erzielung von Fehlertoleranz geeignete Produkte](#)
- [Amazon Builders' Library: Vermeiden von Überlastungen verteilter Systeme durch Übernahme der Steuerung durch den kleineren Service](#)
- [Amazon DynamoDB API \(Steuerebene und Datenebene\)](#)
- [AWS Lambda-Aufteilungen](#) (aufgeteilt in die Steuerebene und die Datenebene)
- [AWS Elemental MediaStore Datenebene](#)
- [Entwickeln hoch resilienter Anwendungen mit Amazon Application Recovery Controller, Teil 1: Stack für eine einzelne Region](#)
- [Entwickeln hoch resilienter Anwendungen mit Amazon Application Recovery Controller, Teil 2: Stack für mehrere Regionen](#)
- [Erstellen von Mechanismen für die Notfallwiederherstellung mit Amazon Route 53](#)
- [Was ist Amazon Application Recovery Controller](#)
- [Kubernetes-Steuerebene und -Datenebene](#)

Zugehörige Videos:

- [Zurück zu den Basics – Verwendung statischer Stabilität](#)
- [Aufbau belastbarer Workloads an mehreren Standorten mit globalen AWS-Services](#)

Zugehörige Beispiele:

- [Einführung zu Amazon Application Recovery Controller](#)
- [Amazon Builders' Library: Vermeiden von Überlastungen verteilter Systeme durch Übernahme der Steuerung durch den kleineren Service](#)
- [Entwickeln hoch resilienter Anwendungen mit Amazon Application Recovery Controller, Teil 1: Stack für eine einzelne Region](#)
- [Entwickeln hoch resilienter Anwendungen mit Amazon Application Recovery Controller, Teil 2: Stack für mehrere Regionen](#)
- [Statische Stabilität mithilfe von Availability Zones](#)

Zugehörige Tools:

- [Amazon CloudWatch](#)
- [AWS X-Ray](#)

REL11-BP05 Verhindern von bimodalem Verhalten mithilfe statischer Stabilität

Workloads sollten statisch stabil sein und nur in einem einzigen Normalmodus ausgeführt werden. Bimodales Verhalten liegt vor, wenn sich die Workload im Normalmodus und im Fehlermodus unterschiedlich verhält.

Sie können beispielsweise versuchen, nach einem Ausfall der Availability Zone eine Wiederherstellung durchzuführen, indem Sie neue Instances in einer anderen Availability Zone starten. Dies kann zu einer bimodalen Reaktion während eines Ausfallmodus führen. Stattdessen sollten Sie Workloads erstellen, die statisch stabil sind und nur in einem Modus betrieben werden. In diesem Beispiel hätten diese Instances vor dem Ausfall in der zweiten Availability Zone bereitgestellt werden sollen. Dieses statische Stabilitätsdesign verifiziert, dass die Workload nur in einem einzigen Modus ausgeführt wird.

Gewünschtes Ergebnis: Workloads zeigen im Normalmodus und im Fehlermodus kein bimodales Verhalten.

Typische Anti-Muster:

- Es wird davon ausgegangen, dass Ressourcen unabhängig vom Umfang des Fehlers immer bereitgestellt werden können.
- Während eines Fehlers wird versucht, dynamisch Ressourcen zu erwerben.
- Es werden keine ausreichenden Ressourcen für Zonen oder Regionen bereitgestellt, bis ein Fehler auftritt.
- Statische stabile Designs werden nur für Rechenressourcen in Erwägung gezogen.

Vorteile der Nutzung dieser bewährten Methode: Workloads, die mit statisch stabilen Designs ausgeführt werden, sind in der Lage, bei normalen Ereignissen und bei Ausfällen vorhersehbare Ergebnisse erzielen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Bimodales Verhalten bedeutet, dass eine Workload im normalen Modus und im Fehlermodus unterschiedliche Verhaltensweisen zeigt (z. B. Verlassen auf den Start neuer Instances bei Ausfall

einer Availability Zone). Ein Beispiel für bimodales Verhalten ist, wenn stabile Amazon EC2-Designs genügend Instances in jeder Availability Zone bereitstellen, so dass die Verarbeitung der Workload auch beim Entfernen einer Availability Zone gewährleistet ist. Elastic Load Balancing oder Amazon Route 53 Health würde prüfen, ob eine Last von den beeinträchtigten Instances weg verlagert werden kann. Nachdem der Datenverkehr verlagert wurde, können Sie AWS Auto Scaling verwenden, um Instances in der ausgefallenen Zone asynchron zu ersetzen und sie in den fehlerfreien Zonen zu starten. Statische Stabilität für die Bereitstellung von Rechenleistung (z. B. EC2-Instances oder -Container) führt zu höchster Zuverlässigkeit.



Statische Stabilität von EC2-Instances über Availability Zones hinweg

Dies muss gegen die Kosten für dieses Modell und den geschäftlichen Nutzen der Aufrechterhaltung der Workload in allen Ausfallsituationen abgewogen werden. Es ist kostengünstiger, weniger Rechenkapazität bereitzustellen und bei einem Ausfall neue Instances zu starten. Bei großen Ausfällen (z. B. bei Beeinträchtigung einer Availability Zone oder Region) ist dieser Ansatz jedoch weniger effektiv, da er sowohl auf einer Betriebsebene als auch auf der Verfügbarkeit ausreichender Ressourcen in den nicht betroffenen Zonen oder Regionen beruht.

Ihre Lösung sollte die Anforderungen an die Zuverlässigkeit und Kosten für Ihre Workload gegeneinander abwägen. Ansätze mit statischer Stabilität gelten für eine Vielzahl von Architekturen, darunter Datenverarbeitungs-Instances, die über Availability Zones verteilt sind, Designs mit Lesereplikaten für Datenbanken, Kubernetes (Amazon EKS)-Clusterdesigns und Failover-Architekturen für mehrere Regionen.

Es ist auch möglich, ein statisch stabileres Design zu implementieren, indem mehr Ressourcen in jeder Zone verwendet werden. Wenn Sie eine größere Anzahl von Zonen hinzufügen, verringert sich die Menge der zusätzlichen Rechenleistung, die Sie für die statische Stabilität benötigen.

Ein weiteres Beispiel für bimodales Verhalten ist eine Netzwerk-Zeitüberschreitung, die dazu führen kann, dass ein System versucht, den Konfigurationsstatus des gesamten Systems zu aktualisieren. Dies kann zur unerwarteten Auslastung einer anderen Komponente führen, die daraufhin ausfallen könnte, was möglicherweise weitere unerwartete Konsequenzen nach sich zieht. Diese negative Feedback-Schleife wirkt sich auf die Verfügbarkeit Ihrer Workload aus. Deshalb sollten Sie stattdessen Systeme erstellen, die statisch stabil sind und nur in einem Modus betrieben werden. Ein statisch stabiles Design arbeitet konstant und aktualisiert den Konfigurationsstatus in regelmäßigen Abständen. Wenn ein Aufruf fehlschlägt, verwendet der Workload den zuvor zwischengespeicherten Wert und löst einen Alarm aus.

Ein weiteres Beispiel für bimodales Verhalten: Sie lassen zu, dass Clients im Fehlerfall den Workload-Cache umgehen. Dies scheint eine Lösung zu sein, die Clientanforderungen erfüllt, sie kann aber die Belastung Ihrer Workload erheblich ändern und führt wahrscheinlich zu Fehlern.

Bewerten Sie kritische Workloads, um festzustellen, für welche Workloads diese Art von Resilienzdesign erforderlich ist. Für diejenigen, die als kritisch eingestuft werden, muss jede Anwendungskomponente überprüft werden. Beispiele für Services, für die statische Stabilitätsbewertungen erforderlich sind:

- Datenverarbeitung: Amazon EC2, EKS-EC2, ECS-EC2, EMR-EC2
- Datenbanken: Amazon Redshift, Amazon RDS, Amazon Aurora
- Speicher: Amazon S3 (einzelne Zone), Amazon EFS (Mounts), Amazon FSx (Mounts)
- Load Balancer: bei bestimmten Designs

Implementierungsschritte

- Erstellen Sie Systeme, die statisch stabil sind und nur in einem einzigen Modus ausgeführt werden. Stellen Sie in diesem Fall in jeder Availability Zone oder Region genügend Instances bereit, um die Workload-Kapazität zu bewältigen, falls eine Availability Zone oder Region entfernt würde. Eine Vielzahl von Services kann für das Routing zu intakten Ressourcen verwendet werden, z. B.:
 - [Regionsübergreifendes DNS-Routing](#)
 - [MRAP Amazon S3-Routing mit mehreren Regionen](#)
 - [AWS Global Accelerator](#)
 - [Amazon Application Recovery Controller](#)
- Konfigurieren Sie [Datenbank-Lesereplikate](#), um den Verlust einer einzelnen primären Instance oder eines Lesereplikats zu berücksichtigen. Wenn der Datenverkehr von Lesereplikaten bedient

wird, sollte die Menge in jeder Availability Zone und jeder Region dem Gesamtbedarf im Fall eines Zonen- oder Regionsausfalls entsprechen.

- Konfigurieren Sie kritische Daten in einem Amazon S3-Speicher, der so konzipiert ist, dass er für die gespeicherten Daten beim Ausfall einer Availability Zone statisch stabil ist. Wenn die [Amazon S3 One Zone-IA](#)-Speicherklasse verwendet wird, sollte diese nicht als statisch stabil angesehen werden, da der Ausfall dieser Zone den Zugriff auf die zugehörigen gespeicherten Daten minimiert.
- [Load Balancer](#) sind manchmal falsch oder so konfiguriert, dass sie eine bestimmte Availability Zone bedienen. In diesem Fall könnte das statisch stabile Design darin bestehen, eine Workload über mehrere AZs in einem komplexeren Design zu verteilen. Das ursprüngliche Design kann aus Sicherheits-, Latenz- oder Kostengründen verwendet werden, um den Verkehr zwischen den Zonen zu reduzieren.

Ressourcen

Zugehörige bewährte Methoden für Well-Architected:

- [Definition der Verfügbarkeit](#)
- [REL11-BP01 Überwachen aller Komponenten der Workload auf Fehler](#)
- [REL11-BP04 Nutzen der Datenebene und nicht der Steuerebene während der Wiederherstellung](#)

Zugehörige Dokumente:

- [Minimierung der Abhängigkeiten bei der Planung der Notfallwiederherstellung](#)
- [Die Amazon Builders' Library: Statische Stabilität mithilfe von Availability Zones](#)
- [Grenzen für die Fehlerisolierung](#)
- [Statische Stabilität mithilfe von Availability Zones](#)
- [Mehrzonen-RDS](#)
- [Minimierung der Abhängigkeiten bei der Planung der Notfallwiederherstellung](#)
- [Regionsübergreifendes DNS-Routing](#)
- [MRAP Amazon S3-Routing mit mehreren Regionen](#)
- [AWS Global Accelerator](#)
- [Amazon Application Recovery Controller](#)
- [Amazon S3 \(einzelne Zone\)](#)
- [Zonenübergreifendes Load Balancing](#)

Zugehörige Videos:

- [Statische Stabilität in AWS: AWS re:Invent 2019: Einführung in die Amazon Builders' Library \(DOP328\)](#)

REL11-BP06 Senden von Benachrichtigungen, wenn sich Ereignisse auf die Verfügbarkeit auswirken

Benachrichtigungen werden nach Erkennung von Schwellenwertüberschreitungen gesendet, auch wenn das durch das Ereignis verursachte Problem automatisch behoben wurde.

Auto Healing sorgt dafür, dass Ihre Workload zuverlässig ist. Allerdings können dadurch auch zugrunde liegende Probleme verschleiert werden, die behoben werden müssen. Implementieren Sie geeignete Überwachungsfunktionen und Ereignisse, damit Sie Problemmuster erkennen können, einschließlich solcher, die durch Auto Healing behoben werden. Auf diese Weise können Sie die Fehlerursachen beheben.

Resiliente Systeme sind so konzipiert, dass Verschlechterungsereignisse sofort an die entsprechenden Teams gemeldet werden. Diese Benachrichtigungen sollten über einen oder mehrere Kommunikationskanäle gesendet werden.

Gewünschtes Ergebnis: Bei Überschreitung von Schwellenwerten wie Fehlerraten, Latenz oder anderen kritischen Leistungsindikatoren (KPIs) werden sofort Benachrichtigungen an die Betriebsteams gesendet, sodass diese Probleme so schnell wie möglich behoben und Auswirkungen auf die Benutzer vermieden oder minimiert werden.

Typische Anti-Muster:

- Es werden zu viele Alarme gesendet.
- Es werden Alarme gesendet, die keine Maßnahmen erfordern.
- Die Schwellenwerte für den Alarm sind zu hoch (überempfindlich) oder zu niedrig (nicht empfindlich genug).
- Es werden keine Alarme für externe Abhängigkeiten gesendet.
- [Graue Fehler](#) werden bei der Planung von Überwachung und Alarmen nicht berücksichtigt.
- Es werden automatische Reparaturen ausgeführt, ohne das entsprechende Team darüber zu benachrichtigen, dass eine Reparatur erforderlich war.

Vorteile der Nutzung dieser bewährten Methode: Durch Benachrichtigungen über die Wiederherstellung werden Betriebs- und Geschäftsteams über Service-Einschränkungen informiert,

sodass sie sofort reagieren können, um sowohl die mittlere Zeit zur Erkennung (Mean Time to Detect, MTDD) als auch die mittlere Wiederherstellungszeit (Mean Time to Repair, MTTR) zu minimieren. Benachrichtigungen zu Wiederherstellungen stellen sicher, dass Sie selten auftretende Probleme nicht ignorieren.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel. Wenn keine geeigneten Überwachungsfunktionen und Mechanismen zur Benachrichtigung bei Ereignissen implementiert werden, kann dies dazu führen, dass Problemmuster nicht erkannt werden, einschließlich solcher, die durch Auto Healing behoben werden. Ein Team wird nur dann auf eine Verschlechterung des Systems aufmerksam gemacht, wenn Benutzer den Kundendienst kontaktieren oder der Fehler zufällig bemerkt wird.

Implementierungsleitfaden

Bei der Definition einer Überwachungsstrategie ist ein ausgelöster Alarm ein häufiges Ereignis. Dieses Ereignis würde wahrscheinlich eine Kennung für den Alarm enthalten, den Alarmstatus (z. B. IN ALARM oder OK) und Einzelheiten darüber, was ihn ausgelöst hat. In vielen Fällen sollte ein Alarmereignis erkannt und eine E-Mail-Benachrichtigung gesendet werden. Dies ist ein Beispiel für eine Aktion bei einem Alarm. Die Alarmbenachrichtigung ist für die Beobachtbarkeit von entscheidender Bedeutung, da hiermit die richtigen Personen darüber informiert werden, dass ein Problem vorliegt. Wenn die Aktionen bei Ereignissen in Ihrer Lösung für die Beobachtbarkeit ausgereift sind, kann das Problem automatisch behoben werden, ohne dass menschliches Eingreifen erforderlich ist.

Sobald Alarmer zur KPI-Überwachung eingerichtet wurden, sollten die entsprechenden Teams Warnmeldungen erhalten, wenn Schwellenwerte überschritten werden. Diese Warnungen können auch verwendet werden, um automatisierte Prozesse auszulösen, die versuchen, die Verschlechterung zu beheben.

Für eine komplexere Schwellenwertüberwachung sollten zusammengesetzte Alarmer in Betracht gezogen werden. Zusammengesetzte Alarmer verwenden eine Reihe von Alarmen zur KPI-Überwachung, um eine Warnung auf Grundlage der Geschäftslogik zu erstellen. CloudWatch-Alarmer können so konfiguriert werden, dass E-Mails gesendet oder Vorfälle mithilfe der Amazon SNS-Integration oder Amazon EventBridge in Drittanbietersystemen zur Nachverfolgung von Vorfällen protokolliert werden.

Implementierungsschritte

Erstellen Sie verschiedene Arten von Alarmen, je nachdem, wie Workloads überwacht werden, z. B.:

- Anwendungsalarme werden verwendet, um zu erkennen, wenn ein Teil der Workload nicht ordnungsgemäß funktioniert.
- [Infrastrukturalarme](#) geben an, wann Ressourcen skaliert werden müssen. Alarmer können visuell auf Dashboards angezeigt werden, Warnungen über Amazon SNS oder E-Mail versenden und mit Auto Scaling zusammenarbeiten, um Workload-Ressourcen hoch- oder herunterskalieren zu können.
- Einfache [statische Alarmer](#) können erstellt werden, um zu überwachen, wann eine Metrik für eine bestimmte Anzahl von Bewertungszeiträumen einen statischen Schwellenwert überschreitet.
- [Zusammengesetzte Alarmer](#) können komplexe Alarmer aus mehreren Quellen berücksichtigen.
- Nachdem der Alarm erstellt wurde, erstellen Sie entsprechende Benachrichtigungsereignisse. Sie können eine [Amazon SNS-API](#) direkt aufrufen, um Benachrichtigungen zu senden und Automatisierung zur Problembehebung oder Kommunikation zu verknüpfen.
- Bleiben Sie mit [AWS Health](#) über Serviceeinbußen auf dem Laufenden. [Erstellen Sie maßgeschneiderte AWS Health-Ereignisbenachrichtigungen](#) für E-Mail- und Chat-Kanäle über [AWS-Benutzerbenachrichtigungen](#), und integrieren Sie sie programmgesteuert in [Ihre Überwachungs- und Warnmeldungs-Tools über Amazon EventBridge](#).

Ressourcen

Zugehörige bewährte Methoden für Well-Architected:

- [Definition der Verfügbarkeit](#)

Zugehörige Dokumente:

- [Erstellen eines CloudWatch-Alarms basierend auf einem statischen Schwellenwert](#)
- [Was ist Amazon EventBridge?](#)
- [Was ist Amazon Simple Notification Service?](#)
- [Veröffentlichen von benutzerdefinierten Metriken](#)
- [Verwenden von Amazon CloudWatch Alarms](#)
- [Einrichten von zusammengesetzten CloudWatch-Alarmen](#)
- [Neuheiten im Bereich AWS-Beobachtbarkeit bei der re:Invent 2022](#)

Zugehörige Tools:

- [CloudWatch](#)
- [CloudWatch X-Ray](#)

REL11-BP07 Architektur Ihres Produkts zur Erfüllung von Verfügbarkeitszielen und Uptime-SLAs (Service Level Agreements)

Erstellen Sie Ihr Produkt so, dass Verfügbarkeit und Betriebszeiten laut SLAs (Service Level Agreements) erfüllt werden. Wenn Sie Verfügbarkeitsziele oder Uptime-SLAs veröffentlichen oder privat vereinbaren, stellen Sie sicher, dass Ihre Architektur und Ihre operativen Prozesse so konzipiert sind, dass sie diese unterstützen.

Gewünschtes Ergebnis: Jede Anwendung hat ein definiertes Verfügbarkeitsziel und ein SLA für Leistungskennzahlen, die überwacht und verwaltet werden können, um die Geschäftsergebnisse zu erreichen.

Typische Anti-Muster:

- Entwurf und Bereitstellung von Workloads ohne Festlegung von SLAs.
- SLA-Metriken werden ohne Begründung oder geschäftliche Anforderungen zu hoch angesetzt.
- SLAs werden ohne Berücksichtigung von Abhängigkeiten und den ihnen zugrunde liegenden SLAs festgelegt.
- Anwendungsdesigns werden ohne Berücksichtigung des Modells der geteilten Verantwortung für die Ausfallsicherheit erstellt.

Vorteile der Nutzung dieser bewährten Methode: Die Entwicklung von Anwendungen auf der Grundlage wichtiger Ausfallsicherheitsziele hilft Ihnen dabei, Ihre Geschäftsziele und Kundenerwartungen zu erfüllen. Diese Ziele sind die Grundlage für die Entwicklung von Anwendungen, bei der verschiedene Technologien bewertet und verschiedene Kompromisse in Betracht gezogen werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Bei der Entwicklung von Anwendungen müssen Sie eine Reihe von Anforderungen berücksichtigen, die sich aus geschäftlichen, operativen und finanziellen Zielen ergeben. Im Rahmen der operativen Anforderungen müssen für Workloads spezifische Metriken für die Ausfallsicherheit festgelegt

werden, damit sie angemessen überwacht und unterstützt werden können. Die Metriken für die Ausfallsicherheit sollten nicht nach der Bereitstellung der Workload festgelegt oder ermittelt werden. Sie sollten in der Entwurfsphase festgelegt werden und als Leitlinien für verschiedene Entscheidungen und Abwägungen dienen.

- Jede Workload sollte ihre eigenen Metriken für die Ausfallsicherheit haben. Diese Metriken können sich von anderen geschäftlichen Anwendungen unterscheiden.
- Die Reduzierung von Abhängigkeiten kann sich positiv auf die Verfügbarkeit auswirken. Jede Workload sollte ihre Abhängigkeiten und deren SLAs berücksichtigen. Wählen Sie im Allgemeinen Abhängigkeiten mit Verfügbarkeitszielen aus, die den Zielen Ihrer Workload entsprechen oder höher sind.
- Ziehen Sie eine lose Verkoppelung in Betracht, damit Ihre Workload trotz der Beeinträchtigung durch Abhängigkeiten korrekt arbeiten kann, sofern dies möglich ist.
- Reduzieren Sie die Abhängigkeiten auf der Steuerebene, insbesondere während der Wiederherstellung oder einer Beeinträchtigung. Evaluieren Sie Designs, die für geschäftskritische Workloads statisch stabil sind. Nutzen Sie den sparsamen Umgang mit Ressourcen, um die Verfügbarkeit dieser Abhängigkeiten in einer Workload zu erhöhen.
- Die Beobachtbarkeit und die Instrumentierung sind entscheidend für das Erreichen von SLAs. Sie reduzieren die Mean Time to Detection (MTTD) und die Mean Time to Repair (MTTR).
- Weniger häufige Störungen (längere MTBF), kürzere Fehlererkennungszeiten (kürzere MTTD) und kürzere Reparaturzeiten (kürzere MTTR) sind die drei Faktoren, die zur Verbesserung der Verfügbarkeit in verteilten Systemen eingesetzt werden.
- Das Festlegen und Einhalten von Metriken für die Ausfallsicherheit einer Workload ist eine der Grundlagen für jedes effektive Design. Diese Designs müssen Kompromisse in Bezug auf Designkomplexität, Service-Abhängigkeiten, Leistung, Skalierung und Kosten berücksichtigen.

Implementierungsschritte

- Überprüfen und dokumentieren Sie das Workload-Design unter Berücksichtigung der folgenden Fragen:
 - Wo werden die Steuerebenen in der Workload verwendet?
 - Wie implementiert die Workload die Ausfallsicherheit?
 - Wie sehen die Designmuster für die Skalierung, automatische Skalierung, Redundanz und hochverfügbare Komponenten aus?
 - Welche Anforderungen gibt es an die Datenkonsistenz und -verfügbarkeit?

- Gibt es Überlegungen zur sparsamen Nutzung von Ressourcen oder zur statischen Stabilität von Ressourcen?
- Welche Abhängigkeiten bestehen zwischen den Services?
- Definieren Sie in Zusammenarbeit mit den Stakeholdern SLA-Metriken auf der Grundlage der Workload-Architektur. Berücksichtigen Sie die SLAs aller Abhängigkeiten, die die Workload nutzt.
- Sobald das SLA-Ziel festgelegt ist, optimieren Sie die Architektur, um die SLA zu erfüllen.
- Sobald das Design festgelegt ist, das die SLA erfüllt, implementieren Sie operative Änderungen, Prozessautomatisierungen und Runbooks, die ebenfalls auf die Reduzierung von MTTD und MTTR ausgerichtet sind.
- Sobald die Bereitstellung erfolgt ist, überwachen Sie die SLA und erstatten Sie darüber Bericht.

Ressourcen

Zugehörige bewährte Methoden:

- [REL03-BP01 Wählen Sie, wie Sie Ihre Arbeitslast segmentieren möchten](#)
- [REL10-BP01 Bereitstellen des Workloads an mehreren Standorten](#)
- [REL11-BP01 Überwachen aller Komponenten der Workload auf Fehler](#)
- [REL11-BP03 Automatisieren der Reparatur auf allen Ebenen](#)
- [REL12-BP04 Testen der Ausfallsicherheit mit Chaos-Engineering](#)
- [REL13-BP01 Definieren von Wiederherstellungszielen bei Ausfällen und Datenverlusten](#)
- [Grundlegendes zum Workload-Status](#)

Zugehörige Dokumente:

- [Verfügbarkeit mit Redundanz](#)
- [Säule der Zuverlässigkeit – Verfügbarkeit](#)
- [Messung der Verfügbarkeit](#)
- [Grenzen für die AWS-Fehlerisolierung](#)
- [Modell der geteilten Verantwortung für Ausfallsicherheit](#)
- [Statische Stabilität mithilfe von Availability Zones](#)
- [AWS Service Level Agreements \(SLAs\)](#)

- [Empfehlungen zur zellenbasierten Architektur in AWS](#)
- [AWS-Infrastruktur](#)
- [Whitepaper „Erweiterte Multi-AZ-Resilience-Muster“](#)

Zugehörige Services:

- [Amazon CloudWatch](#)
- [AWS Config](#)
- [AWS Trusted Advisor](#)

REL 12. Wie lässt sich die Zuverlässigkeit testen?

Nachdem Sie Ihre Workload so konzipiert haben, dass sie den Belastungen der Produktion standhält, sind Tests die einzige Möglichkeit, sie auf die erwartete Funktionalität und Ausfallsicherheit hin zu testen.

Best Practices

- [REL12-BP01 Untersuchen von Fehlern mit Playbooks](#)
- [REL12-BP02 Durchführen von Analysen nach Vorfällen](#)
- [REL12-BP03 Testen von Skalierbarkeits- und Leistungsanforderungen](#)
- [REL12-BP04 Testen der Ausfallsicherheit mit Chaos-Engineering](#)
- [REL12-BP05 Regelmäßiges Durchführen von Gamedays](#)

REL12-BP01 Untersuchen von Fehlern mit Playbooks

Gestatten Sie konsistente und schnelle Antworten auf noch unbekannte Fehlerszenarien, indem Sie den Untersuchungsprozess in Playbooks dokumentieren. Playbooks sind vordefinierte Abläufe zum Identifizieren der Faktoren, die zu einem Fehlerszenario beitragen. Die Ergebnisse aus jedem Prozessschritt sind die Grundlage für die nächsten Schritte. Nach diesem Muster wird vorgegangen, bis das Problem identifiziert oder eskaliert wird.

Das Playbook ist eine proaktive Planung, die für effektive Reaktionen erforderlich ist. Wenn nicht vom Playbook abgedeckte Fehlerszenarien in der Produktion auftreten, beheben Sie zunächst das Problem. Analysieren Sie danach die unternommenen Schritte und verwenden Sie diese, um einen neuen Eintrag im Playbook hinzuzufügen.

Beachten Sie, dass Playbooks als Reaktion auf bestimmte Vorfälle verwendet werden, während Runbooks verwendet werden, um bestimmte Ergebnisse zu erzielen. Häufig werden Runbooks für Routineaktivitäten verwendet, Playbooks hingegen, um auf außergewöhnliche Ereignisse zu reagieren.

Typische Anti-Muster:

- Planen der Bereitstellung einer Workload, ohne die Prozesse zur Diagnose von Problemen oder Reaktion auf Vorfälle zu kennen.
- Ungeplante Entscheidungen darüber, in welchen Systemen bei der Untersuchung von Ereignissen Protokolle und Metriken erfasst werden sollen.
- Metriken und Ereignisse werden nicht lange genug aufbewahrt, um die Daten abrufen zu können.

Vorteile der Nutzung dieser bewährten Methode: Durch die Erfassung von Playbooks wird sichergestellt, dass Prozesse konsistent befolgt werden können. Ihre Playbooks werden als Code festgehalten, um die Entstehung von Fehlern durch manuelle Aktivitäten zu reduzieren. Durch die Automatisierung von Playbooks kann schneller auf Ereignisse reagiert werden, weil Teammitglieder nicht eingreifen müssen oder ihnen vor dem Eingreifen zusätzliche Informationen zur Verfügung gestellt werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

- Ermitteln Sie Probleme mithilfe von Playbooks. Playbooks sind dokumentierte Prozesse für die Untersuchung von Problemen. Durch die Dokumentation der Prozesse in Playbooks werden die Voraussetzungen für eine einheitliche und schnelle Reaktion auf Fehlerszenarien geschaffen. Playbooks müssen die Informationen und Anleitungen enthalten, die eine entsprechend qualifizierte Person zum Zusammentragen sachdienlicher Informationen, zum Identifizieren möglicher Fehlerursachen, zum Isolieren von Fehlern und zum Bestimmen beitragender Faktoren (zum Analysieren nach einem Vorfall) benötigt.
- Implementieren Sie Playbooks als Code. Führen Sie Ihre Operationen als Code aus, indem Sie Skripts für Ihre Playbooks erstellen, um Konsistenz sicherzustellen und Fehler zu reduzieren, die durch manuelle Prozesse verursacht werden. Playbooks können aus mehreren Skripts bestehen, die die verschiedenen Schritte darstellen, die erforderlich sein können, um die zu einem Problem beitragenden Faktoren zu identifizieren. Runbook-Aktivitäten können aufgerufen oder im Rahmen von Playbook-Aktivitäten ausgeführt werden. Sie können auch als Antwort auf identifizierte Ereignisse die Ausführung eines Playbooks auslösen.

- [Automatisieren Ihrer betrieblichen Playbooks mit AWS Systems Manager](#)
- [AWS Systems Manager Run Command](#)
- [AWS Systems Manager Automation](#)
- [Was ist AWS Lambda?](#)
- [Was ist Amazon EventBridge?](#)
- [Verwenden von Amazon CloudWatch Alarms](#)

Ressourcen

Zugehörige Dokumente:

- [AWS Systems Manager Automation](#)
- [AWS Systems Manager Run Command](#)
- [Automatisieren Ihrer betrieblichen Playbooks mit AWS Systems Manager](#)
- [Verwenden von Amazon CloudWatch Alarms](#)
- [Verwenden von Canaries \(Amazon CloudWatch Synthetics\)](#)
- [Was ist Amazon EventBridge?](#)
- [Was ist AWS Lambda?](#)

Zugehörige Beispiele:

- [Automating operations with Playbooks and Runbooks](#)

REL12-BP02 Durchführen von Analysen nach Vorfällen

Überprüfen Sie die Ereignisse mit Auswirkungen auf Kunden und bestimmen Sie die beitragenden Faktoren und Präventivmaßnahmen. Entwickeln Sie anhand dieser Informationen Abhilfemaßnahmen, um Wiederholungen einzuschränken oder zu verhindern. Entwickeln Sie Verfahren für schnelle und effektive Reaktionen. Informieren Sie nach Bedarf auf zielgruppengerechte Weise über beitragende Faktoren und Korrekturmaßnahmen. Legen Sie eine Kommunikationsmethode fest, um andere bei Bedarf über die Ursachen zu informieren.

Bewerten Sie, warum bestehende Tests das Problem nicht gefunden haben. Fügen Sie Tests für diesen Fall hinzu, wenn noch keine Tests vorhanden sind.

Gewünschtes Ergebnis: Ihre Teams haben einen konsistenten und vereinbarten Ansatz für den Umgang mit Analysen nach einem Vorfall. Ein Mechanismus ist der [Correction of Error \(CoE\)-Prozess](#). Der COE-Prozess hilft Ihren Teams, die Ursachen für Vorfälle zu identifizieren, zu verstehen und zu beseitigen. Gleichzeitig werden Mechanismen und Leitlinien entwickelt, um die Wahrscheinlichkeit zu verringern, dass sich ein solcher Vorfall wiederholt.

Typische Anti-Muster:

- Beitragende Faktoren werden ermittelt, es wird jedoch nicht weiter nach anderen potenziellen Problemen und Lösungsansätzen gesucht.
- Es werden nur menschliche Fehlerursachen ermittelt, es wird aber keine Schulung oder Automatisierung bereitgestellt, die menschliche Fehler verhindern könnte.
- Der Fokus liegt auf Schuldzuweisungen, anstatt die Ursache zu verstehen, wodurch eine Kultur der Angst entsteht und eine offene Kommunikation behindert wird.
- Es wird versäumt, Erkenntnisse weiterzugeben, wodurch die Ergebnisse der Ereignisanalyse in einer kleinen Gruppe bleiben und andere nicht von den gewonnenen Erkenntnissen profitieren können.
- Es gibt keine Mechanismen zur Erfassung des institutionellen Wissens, wodurch wertvolle Erkenntnisse verloren gehen, da die gewonnenen Erkenntnisse nicht in Form von aktualisierten bewährten Methoden festgehalten werden und es zu wiederholten Vorfällen mit derselben oder einer ähnlichen Ursache kommt.

Vorteile der Nutzung dieser bewährten Methode: Durch Analysen von Vorfällen und das Teilen von Ergebnissen können die Risiken für andere Workloads mit den gleichen beitragenden Faktoren die Risiken verringert werden. Außerdem können Abhilfemaßnahmen oder automatisierte Wiederherstellungen implementiert werden, bevor es zu einem Vorfall kommt.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Durch gute Analysen nach Vorfällen lassen sich allgemeine Lösungen für Probleme mit Architekturmustern ermitteln, die Sie bereits an anderer Stelle in den Systemen anwenden.

Ein Grundpfeiler des COE-Prozesses ist die Dokumentation und Behandlung von Problemen. Es wird empfohlen, ein standardisiertes Verfahren zur Dokumentation kritischer Ursachen festzulegen und sicherzustellen, dass diese überprüft und behoben werden. Weisen Sie die Verantwortung für den

Analyseprozess nach einem Vorfall eindeutig zu. Benennen Sie ein verantwortliches Team oder eine Person, die die Untersuchungen von Vorfällen und die Folgemaßnahmen beaufsichtigt.

Fördern Sie eine Kultur, die sich auf Lernen und Verbesserung konzentriert, anstatt Schuldzuweisungen vorzunehmen. Betonen Sie, dass das Ziel darin besteht, zukünftige Vorfälle zu verhindern, und nicht darin, Einzelpersonen zu strafen.

Entwickeln Sie klar definierte Verfahren für die Durchführung von Analysen nach einem Vorfall. Diese Verfahren sollten die zu ergreifenden Schritte, die zu sammelnden Informationen und die Schlüsselfragen, die während der Analyse zu behandeln sind, darlegen. Untersuchen Sie Vorfälle gründlich und gehen Sie dabei über die unmittelbaren Ursachen hinaus, um die Grundursachen und die beitragenden Faktoren zu ermitteln. Verwenden Sie Techniken wie die [5-Why-Methode](#), um sich eingehend mit den zugrundeliegenden Problemen zu befassen.

Führen Sie eine Sammlung von Erkenntnissen, die Sie aus der Analyse von Vorfällen gewonnen haben. Dieses institutionelle Wissen kann als Referenz für zukünftige Vorfälle und Präventionsmaßnahmen dienen. Tauschen Sie die Ergebnisse und Erkenntnisse aus den Analysen nach dem Vorfall aus und erwägen Sie, offene Besprechungen nach dem Vorfall abzuhalten, um die gewonnenen Erkenntnisse zu diskutieren.

Implementierungsschritte

- Achten Sie bei der Analyse nach einem Vorfall darauf, dass der Prozess frei von Schuldzuweisungen ist. Dies ermöglicht es den an dem Vorfall beteiligten Personen, die vorgeschlagenen Korrekturmaßnahmen sachlich zu beurteilen und fördert eine ehrliche Selbsteinschätzung und die Zusammenarbeit zwischen den Teams.
- Definieren Sie eine standardisierte Methode zur Dokumentation kritischer Probleme. Ein solches Dokument könnte beispielsweise folgendermaßen strukturiert sein:
 - Was ist passiert?
 - Welche Auswirkungen gab es auf Kunden und Ihr Unternehmen?
 - Was war die Ursache?
 - Welche Daten haben Sie, um dies zu unterstützen?
 - Zum Beispiel Metriken und Grafiken
 - Welches waren die kritischen Auswirkungen auf die Säulen, insbesondere in puncto Sicherheit?
 - Beim Entwerfen von Workloads sollten Sie je nach Geschäftskontext zwischen den einzelnen Säulen abwägen. Diese Geschäftsentscheidungen können Ihre technischen Prioritäten beeinflussen. Sie können optimieren, um Kosten zulasten der Zuverlässigkeit in

Entwicklungsumgebungen zu senken, oder Sie können bei unternehmenskritischen Lösungen die Zuverlässigkeit mit höheren Kosten optimieren. Sicherheit ist immer oberstes Gebot, da Sie Ihre Kunden schützen müssen.

- Welche Erkenntnisse haben Sie gewonnen?
- Welche Maßnahmen ergreifen Sie?
 - Aktionselemente
 - Verwandte Elemente
- Erstellen Sie klar definierte Standardverfahren für die Durchführung von Analysen nach einem Vorfall.
- Richten Sie ein standardisiertes Verfahren zur Meldung von Vorfällen ein. Dokumentieren Sie alle Vorfälle ausführlich, einschließlich des ersten Vorfallberichts, der Protokolle, der Kommunikation und der während des Vorfalls getroffenen Maßnahmen.
- Denken Sie daran, dass ein Vorfall nicht unbedingt einen Ausfall zur Folge haben muss. Es könnte sich um einen Beinahe-Unfall handeln oder um ein System, das auf unerwartete Weise funktioniert und dennoch seine Geschäftsfunktion erfüllt.
- Verbessern Sie Ihren Analyseprozess nach einem Vorfall kontinuierlich auf Grundlage von Rückmeldungen und gewonnenen Erkenntnissen.
- Halten Sie die wichtigsten Erkenntnisse in einem Wissensmanagementsystem fest und überlegen Sie, welche Muster in Entwicklerhandbücher oder Checklisten vor der Bereitstellung aufgenommen werden sollten.

Ressourcen

Zugehörige Dokumente:

- [Warum sollten Sie eine Fehlerkorrektur \(Correction of Error, CoE\) entwickeln](#)

Zugehörige Videos:

- [Der Ansatz von Amazon: erfolgreiches Scheitern](#)
- [AWS re:Invent 2021 – Die Amazon Builders' Library: Betriebliche Exzellenz von Amazon](#)

REL12-BP03 Testen von Skalierbarkeits- und Leistungsanforderungen

Nutzen Sie Techniken wie Lasttests, um zu prüfen, ob die Workload die Skalierungs- und Leistungsanforderungen erfüllt.

In der Cloud können Sie bei Bedarf eine Testumgebung in Produktionsgröße für Ihren Workload erstellen. Anstatt sich auf eine herunterskalierte Testumgebung zu verlassen und dadurch ungenaue Vorhersagen des Produktionsverhaltens zu riskieren, können Sie in der Cloud eine Testumgebung bereitzustellen, die Ihrer erwarteten Produktionsumgebung sehr nahe kommt. Diese Umgebung ermöglicht Ihnen Tests in einer genaueren Simulation der realen Bedingungen, denen Ihre Anwendung ausgesetzt ist.

Neben der Durchführung von Leistungstests ist es wichtig, sicherzustellen, dass Ihre Basisressourcen, Skalierungseinstellungen, Service Quotas und die Ausfallsicherheit unter Auslastung wie erwartet funktionieren. Dieser ganzheitliche Ansatz stellt sicher, dass Ihre Anwendung auch unter den anspruchsvollsten Bedingungen zuverlässig skaliert und nach Bedarf ausgeführt werden kann.

Gewünschtes Ergebnis: Ihr Workload behält sein erwartetes Verhalten auch bei Spitzenlast bei. Sie kümmern sich proaktiv um leistungsbezogene Probleme, die auftreten können, wenn die Anwendung wächst und sich weiterentwickelt.

Typische Anti-Muster:

- Sie verwenden Testumgebungen, die nicht genau mit der Produktionsumgebung übereinstimmen.
- Sie behandeln Lasttests als separate, einmalige Aktivität und nicht als integrierten Bestandteil der Continuous Integration (CI)-Pipeline für die Bereitstellung.
- Sie definieren keine klaren und messbaren Leistungsanforderungen wie beispielsweise Ziele in Bezug auf Reaktionszeit, Durchsatz und Skalierbarkeit.
- Sie nutzen Szenarien mit unrealistischer oder unzureichender Last für Ihre Tests und versäumen es, Spitzenlasten, plötzliche Spitzen und anhaltend hohe Lasten zu testen.
- Sie unterziehen den Workload keinem Belastungstest, indem Sie die erwarteten Lastgrenzen überschreiten.
- Sie verwenden unzureichende oder unangemessene Tools für Lasttests und die Erstellung von Leistungsprofilen.
- Sie verfügen nicht über umfassende Überwachungs- und Warnsysteme, um Leistungsmetriken zu verfolgen und Anomalien zu erkennen.

Vorteile der Nutzung dieser bewährten Methode:

- Mithilfe von Lasttests können Sie potenzielle Leistungsengpässe in Ihrem System vor dem Übergang in die Produktion erkennen. Wenn Sie den Datenverkehr und die Workloads auf Produktionsebene simulieren, können Sie Bereiche identifizieren, in denen Ihr System möglicherweise Schwierigkeiten hat, die Last zu bewältigen, beispielsweise langsame Reaktionszeiten, Ressourcenengpässe oder Systemausfälle.
- Wenn Sie Ihr System unter verschiedenen Lastbedingungen testen, können Sie die Ressourcenanforderungen zur Unterstützung Ihres Workloads besser verstehen. Mithilfe dieser Informationen können Sie fundierte Entscheidungen über die Ressourcenzuweisung treffen und eine übermäßige oder zu geringe Bereitstellung von Ressourcen verhindern.
- Um potenzielle Fehlerquellen zu identifizieren, können Sie beobachten, wie sich Ihr Workload in Situationen mit hoher Last verhält. Diese Informationen helfen Ihnen, die Zuverlässigkeit und Ausfallsicherheit Ihres Workloads zu verbessern, indem Sie nach Bedarf Fehlertoleranzmechanismen, Failover-Strategien und Redundanzmaßnahmen implementieren.
- Sie erkennen und beheben Leistungsprobleme frühzeitig und können so die kostspieligen Folgen von Systemausfällen, langsamen Reaktionszeiten und unzufriedenen Benutzern vermeiden.
- Während der Tests erfasste detaillierte Leistungsdaten und Profilinformationen können Ihnen helfen, leistungsbezogene Probleme zu beheben, die in der Produktion auftreten können. Dies kann eine schnellere Reaktion auf Vorfälle und eine zügigere Behebung von Problemen ermöglichen, wodurch die Auswirkungen auf die Benutzer und den Betrieb Ihres Unternehmens verringert werden.
- In bestimmten Branchen können proaktive Leistungstests dazu beitragen, dass Ihr Workload die Compliance-Standards erfüllt, wodurch das Risiko von Strafen oder rechtlichen Problemen verringert wird.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Der erste Schritt besteht darin, eine umfassende Teststrategie zu definieren, die alle Aspekte der Skalierungs- und Leistungsanforderungen abdeckt. Definieren Sie zunächst klar die Servicelevel-Ziele (SLOs, Service-Level Objectives) Ihres Workloads auf der Grundlage Ihrer Geschäftsanforderungen, also beispielsweise Durchsatz, Latenzhistogramm und Fehlerrate. Entwerfen Sie dann eine Reihe von Tests, mit denen verschiedene Lastszenarien simuliert werden können, von durchschnittlicher Auslastung bis hin zu plötzlichen Spitzen und anhaltenden

Spitzenlasten. Stellen Sie sicher, dass das Verhalten des Workloads Ihren SLOs entspricht. Diese Tests sollten automatisiert und in Ihre Pipeline für kontinuierliche Integration und Bereitstellung integriert werden, um Leistungsrückgänge frühzeitig im Entwicklungsprozess zu erkennen.

Um die Skalierung und Leistung effektiv testen zu können, sollten Sie in die richtigen Tools und eine geeignete Infrastruktur investieren. Hierzu gehören Tools für Lasttests, mit denen ein realistischer Benutzerverkehr generiert werden kann, Tools für die Erstellung von Leistungsprofilen zur Ermittlung von Engpässen und Überwachungslösungen zur Verfolgung wichtiger Metriken. Wichtig ist es sicherzustellen, dass Ihre Testumgebungen in Bezug auf Infrastruktur und Umgebungsbedingungen möglichst genau mit der Produktionsumgebung übereinstimmen, um möglichst genaue Testergebnisse zu erzielen. Verwenden Sie Infrastructure-as-Code- und containerbasierte Anwendungen, um die zuverlässige Replikation und Skalierung produktionsähnlicher Setups zu vereinfachen.

Skalierungs- und Leistungstests sind ein fortlaufender Prozess, keine einmalige Aktivität. Implementieren Sie eine umfassende Überwachung und Warnmeldungen, um die Leistung der Anwendung in der Produktion zu verfolgen, und verwenden Sie diese Daten, um Ihre Teststrategien und Optimierungsbemühungen kontinuierlich zu verbessern. Analysieren Sie Leistungsdaten regelmäßig, um neu auftretende Probleme zu identifizieren, neue Skalierungsstrategien zu testen, Optimierungen vorzunehmen und so die Effizienz und Zuverlässigkeit der Anwendung zu verbessern. Wenn Sie einen iterativen Ansatz verfolgen und kontinuierlich aus Produktionsdaten lernen, können Sie überprüfen, ob sich Ihre Anwendung an variable Benutzeranforderungen anpassen und im Laufe der Zeit Ausfallsicherheit und eine optimale Leistung aufrechterhalten kann.

Implementierungsschritte

1. Legen Sie klare und messbare Leistungsanforderungen fest, beispielsweise Ziele in Bezug auf Reaktionszeit, Durchsatz und Skalierbarkeit. Diese Anforderungen sollten auf den Nutzungsmustern Ihres Workloads, den Erwartungen der Benutzer und Ihren Geschäftsanforderungen basieren.
2. Wählen und konfigurieren Sie ein Tool für Lasttests, das die Lastmuster und das Benutzerverhalten in Ihrer Produktionsumgebung genau nachahmen kann.
3. Richten Sie eine Testumgebung ein, die der Produktionsumgebung, einschließlich Infrastruktur und Umgebungsbedingungen, sehr nahe kommt, um die Genauigkeit Ihrer Testergebnisse zu verbessern.
4. Erstellen Sie eine Testsuite, die eine Vielzahl von Szenarien abdeckt, von durchschnittlichen Nutzungsmustern über Spitzenlasten und kurzfristige Spitzen bis hin zu anhaltend hohen Lasten.

Integrieren Sie die Tests in Ihre Pipelines für die kontinuierliche Integration und Bereitstellung, um Leistungsrückgänge schon früh im Entwicklungsprozess zu erkennen.

5. Führen Sie Lasttests durch, um den tatsächlichen Benutzerverkehr zu simulieren und zu verstehen, wie sich Ihre Anwendung unter verschiedenen Lastbedingungen verhält. Überschreiten Sie die erwartete Last, um Ihre Anwendung einem Stresstest zu unterziehen, und beobachten Sie das Verhalten der Anwendung. Prüfen Sie beispielsweise, ob es zu einer Verschlechterung der Reaktionszeit, zu einer Ressourcenüberlastung oder zu Systemausfällen kommt. Auf diese Weise können Sie die Belastungsgrenze Ihrer Anwendung ermitteln und fundierte Skalierungsstrategien entwickeln. Bewerten Sie die Skalierbarkeit Ihres Workloads, indem Sie die Last schrittweise erhöhen, und messen Sie die Auswirkungen auf die Leistung, um Skalierungsgrenzen zu ermitteln und zukünftige Kapazitätsanforderungen zu planen.
6. Implementieren Sie eine umfassende Überwachung und Warnmeldungen, um Leistungsmetriken zu verfolgen, Anomalien zu erkennen und Skalierungsaktionen oder Benachrichtigungen einzuleiten, wenn Schwellenwerte überschritten werden.
7. Überwachen und analysieren Sie die Leistungsdaten kontinuierlich, um Bereiche mit Verbesserungspotenzial zu identifizieren. Verbessern Sie Ihre Teststrategien und Optimierungsbemühungen durch Wiederholung.

Ressourcen

Zugehörige bewährte Methoden:

- [REL01-BP04 Überwachen und Verwalten von Kontingenten](#)
- [REL06-BP01 Überwachen aller Komponenten des Workloads \(Generierung\)](#)
- [REL06-BP03 Senden von Benachrichtigungen \(Verarbeitung und Benachrichtigung in Echtzeit\)](#)

Zugehörige Dokumente:

- [Anwendungen für Lasttests](#)
- [Distributed Load Testing auf AWS](#)
- [Application Performance Monitoring](#)
- [Amazon-EC2-Testrichtlinie](#)

Zugehörige Beispiele:

- [Verteilte Lasttests in AWS \(GitHub\)](#)

Zugehörige Tools:

- [Amazon CodeGuru Profiler](#)
- [Amazon CloudWatch RUM](#)
- [Apache JMeter](#)
- [K6](#)
- [Vegeta](#)
- [Hey](#)
- [ab](#)
- [wrk](#)
- [Distributed Load Testing auf AWS](#)

REL12-BP04 Testen der Ausfallsicherheit mit Chaos-Engineering

Führen Sie regelmäßig Chaos-Experimente in oder nahe an Produktionsumgebungen aus, um zu verstehen, wie Ihr System auf ungünstige Bedingungen reagiert.

Gewünschtes Ergebnis:

Die Ausfallsicherheit der Workload wird regelmäßig durch die Anwendung von Chaos-Engineering in Form von Fehlerinjektionsexperimenten oder einer Injektion unerwarteter Last überprüft. Dazu kommen Tests der Ausfallsicherheit, um das bekannte erwartete Verhalten der Workload während eines Ereignisses zu validieren. Kombinieren Sie Chaos-Engineering mit Tests der Ausfallsicherheit, um sicher zu sein, dass Ihre Workload Komponentenausfällen standhalten und sich von unerwarteten Unterbrechungen erholen kann – mit minimalen oder gar keinen Auswirkungen.

Typische Anti-Muster:

- Auslegung der Systeme auf Ausfallsicherheit, aber keine Überprüfung, wie die Workload als Ganzes funktioniert, wenn Fehler auftreten.
- Keine Experimente unter echten Bedingungen und der erwarteten Last.
- Keine Behandlung der Experimente als Code und fehlendes Aufrechterhalten während des Entwicklungszyklus.

- Keine Durchführung von Chaos-Experimenten als Teil Ihrer CI/CD-Pipeline und außerhalb von Bereitstellungen.
- Keine Nutzung früherer Analysen nach Vorfällen bei der Entscheidung über die Fehler, mit denen experimentiert werden soll.

Vorteile der Nutzung dieser bewährten Methode: Durch die Injektion von Fehlern zur Überprüfung der Resilienz Ihrer Workload gewinnen Sie die nötige Zuversicht, dass die Wiederherstellungsverfahren Ihres resilienten Entwurfs im Fall eines realen Fehlers funktionieren.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Das Chaos-Engineering bietet Ihren Teams die nötigen Chancen, um auf kontrollierte Weise kontinuierlich reale Störungen (Simulationen) auf Serviceanbieter-, Infrastruktur-, Workload- und Komponentenebene zu injizieren – mit nur minimalen oder gar keinen Auswirkungen auf Ihre Kunden. Ihre Teams können so aus Fehlern lernen und die Resilienz Ihrer Workloads beobachten, messen und verbessern. Darüber hinaus können sie überprüfen, ob Warnungen ausgelöst werden und die Teams über Ereignisse benachrichtigt werden.

Bei kontinuierlicher Ausführung kann das Chaos-Engineering Mängel in Ihren Workloads aufzeigen, die sich negativ auf Verfügbarkeit und Ausführung auswirken könnten, wenn sie nicht behoben werden.

Note

Beim Chaos-Engineering geht es um das Experimentieren mit einem System, um sich davon zu überzeugen, dass das System in der Produktion auch außergewöhnlichen Bedingungen standhalten kann. – [Grundlagen des Chaos-Engineering](#)

Wenn ein System diesen Disruptionen standhalten kann, sollte das Chaos-Experiment weiter als automatisierter Regressionstest ausgeführt werden. In dieser Form sollten Chaos-Experimente als Teil Ihres Systementwicklungszyklus (Systems Development Lifecycle, SDLC) und Ihrer CI/CD-Pipeline ausgeführt werden.

Um sicherzustellen, dass Ihre Workload resilient gegenüber dem Ausfall von Komponenten ist, sollten Sie im Rahmen Ihrer Experimente Ereignisse aus der Praxis injizieren. Sie könnten beispielsweise

mit dem Verlust von Amazon EC2-Instances oder einem Failover der primären Amazon RDS-Datenbank-Instance experimentieren und so verifizieren, dass Ihre Workload nicht beeinträchtigt wird (oder nur minimal beeinträchtigt wird). Mit einer Kombination von Komponentenfehlern könnten Sie Ereignisse simulieren, die von einer Disruption in einer Availability Zone verursacht werden könnten.

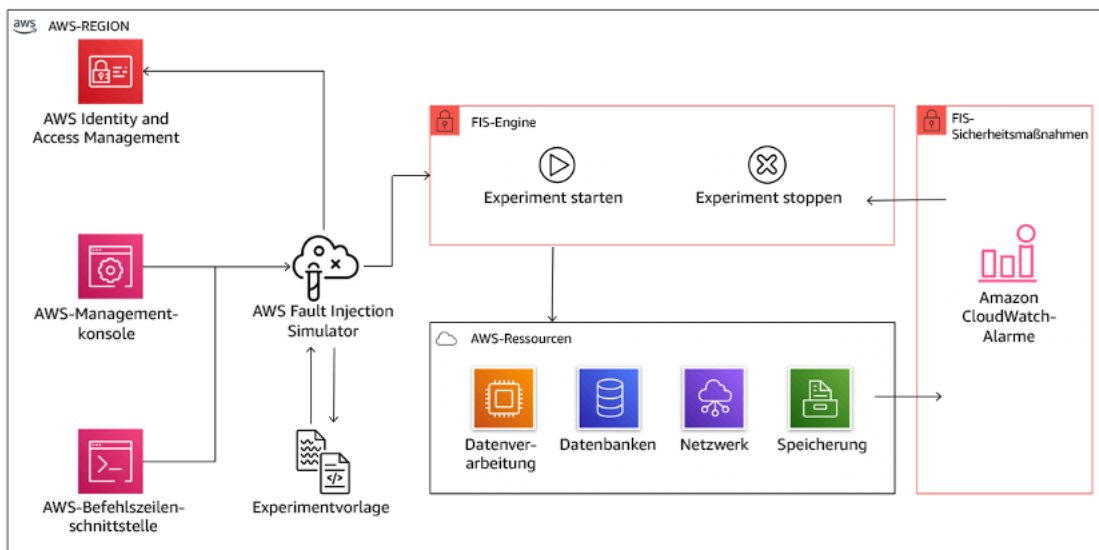
Hinsichtlich Fehlern auf Anwendungsebene (z. B. Abstürzen) könnten Sie mit Stressfaktoren wie Speicher- und CPU-Auslastung beginnen.

Zur Validierung von [Fallback- oder Failover-Mechanismen](#) für externe Abhängigkeiten, die bei zeitweisen Netzwerkdisruptionen ausgelöst werden, sollten Ihre Komponenten diese Ereignisse durch das Blockieren des Zugriffs auf externe Anbieter über einen bestimmten Zeitraum simulieren, der von wenigen Sekunden bis zu mehreren Stunden dauern kann.

Andere Degradierungsmodi führen möglicherweise zu einer reduzierten Funktionalität und zu verzögerten Reaktionen, was eine Disruption Ihrer Services verursachen kann. Bekannte Quellen für diese Degradierung sind eine erhöhte Latenz bei kritischen Services und eine unzuverlässige Netzwerkkommunikation (Verlust von Paketen). Experimente mit diesen Fehlern, darunter Netzwerkeffekten wie Latenz, Nachrichtenverlust und DNS-Ausfällen, könnten die fehlende Fähigkeit zur Auflösung eines Namens, zum Erreichen des DNS-Service oder zur Herstellung von Verbindungen zu abhängigen Services umfassen.

Chaos-Engineering-Tools:

AWS Fault Injection Service (AWS FIS) ist ein vollständig verwalteter Service für die Injektion von Fehlern, den Sie innerhalb oder außerhalb Ihrer CD-Pipeline verwenden können, um mit diesen Fehlern zu experimentieren. AWS FIS ist eine gute Wahl für Gamedays, die dem Chaos-Engineering gewidmet sind. Es unterstützt die gleichzeitige Einführung von Fehlern für verschiedenen Ressourcentypen, darunter Amazon EC2, Amazon Elastic Container Service (Amazon ECS), Amazon Elastic Kubernetes Service (Amazon EKS) und Amazon RDS. Zu diesen Fehlern gehören die Beendigung von Ressourcen, die Erzwingung von Failovern, die Auslastung von CPU oder Arbeitsspeicher, Drosselung, Latenz und Paketverluste. Da dieser Service in Amazon CloudWatch Alarms integriert ist, können Sie Stoppbedingungen als Integritätsschutz einrichten, um Experimente rückgängig zu machen, wenn sie unerwartete Auswirkungen haben.



AWS Fault Injection Service lässt sich in AWS-Ressourcen integrieren, um Ihnen die Ausführung von Fehlerinjektionsexperimenten für Ihre Workloads zu ermöglichen.

Es gibt auch verschiedene Drittanbieteroptionen für Fehlerinjektionsexperimente. Dazu gehören Open-Source-Tools wie [Chaos Toolkit](#), [Chaos Mesh](#) und [Litmus Chaos](#) sowie kommerzielle Tools wie Gremlin. Um den Umfang der Fehler zu erweitern, die in AWS injiziert werden können, ist AWS FIS [in Chaos Mesh und Litmus Chaos integrierbar](#), sodass Sie die Workflows zur Fehlerinjektion zwischen mehreren Tools koordinieren können. Sie können beispielsweise einen Stresstest für die CPU eines Pods mit Chaos-Mesh- oder Litmus-Fehlern ausführen und gleichzeitig einen zufällig ausgewählten Prozentsatz von Cluster-Knoten mit AWS FIS-Fehleraktionen beenden.

Implementierungsschritte

1. Ermitteln Sie die Fehler, mit denen experimentiert werden soll.

Bewerten Sie das Design Ihrer Workload in Bezug auf die Resilienz. Solche Designs (die unter Verwendung der bewährten Methoden des [Well-Architected Framework](#) erstellt wurden) berücksichtigen Risiken, die auf kritischen Abhängigkeiten, vergangenen Ereignissen, bekannten Problemen und Compliance-Anforderungen basieren. Listen Sie die einzelnen Elemente des Designs auf, die Resilienz zeigen sollen, und die Fehler, denen es standhalten soll. Weitere Informationen zur Erstellung solcher Listen finden Sie im [Whitepaper zur Überprüfung der betrieblichen Bereitschaft](#), in dem Sie auch erfahren, wie Sie einen Prozess erstellen können, um das Wiederauftreten früherer Vorfälle zu verhindern. Der Prozess für die Analyse von Fehlerarten und ihren Auswirkungen (Failure Modes and Effects Analysis, FMEA) stellt Ihnen ein Framework für Fehleranalysen auf Komponentenebene und die Analyse der Auswirkungen dieser Fehler

auf Ihre Workload bereit. FMEA wird von Adrian Cockcroft unter [Failure Modes and Continuous Resilience](#) näher beschrieben.

2. Weisen Sie jedem Fehler eine Priorität zu.

Beginnen Sie mit einer groben Kategorisierung wie „Hoch“, „Mittel“ oder „Niedrig“. Berücksichtigen Sie bei der Festlegung der Priorität die Häufigkeit des Fehlers und die Auswirkungen des Fehlers auf die Workload insgesamt.

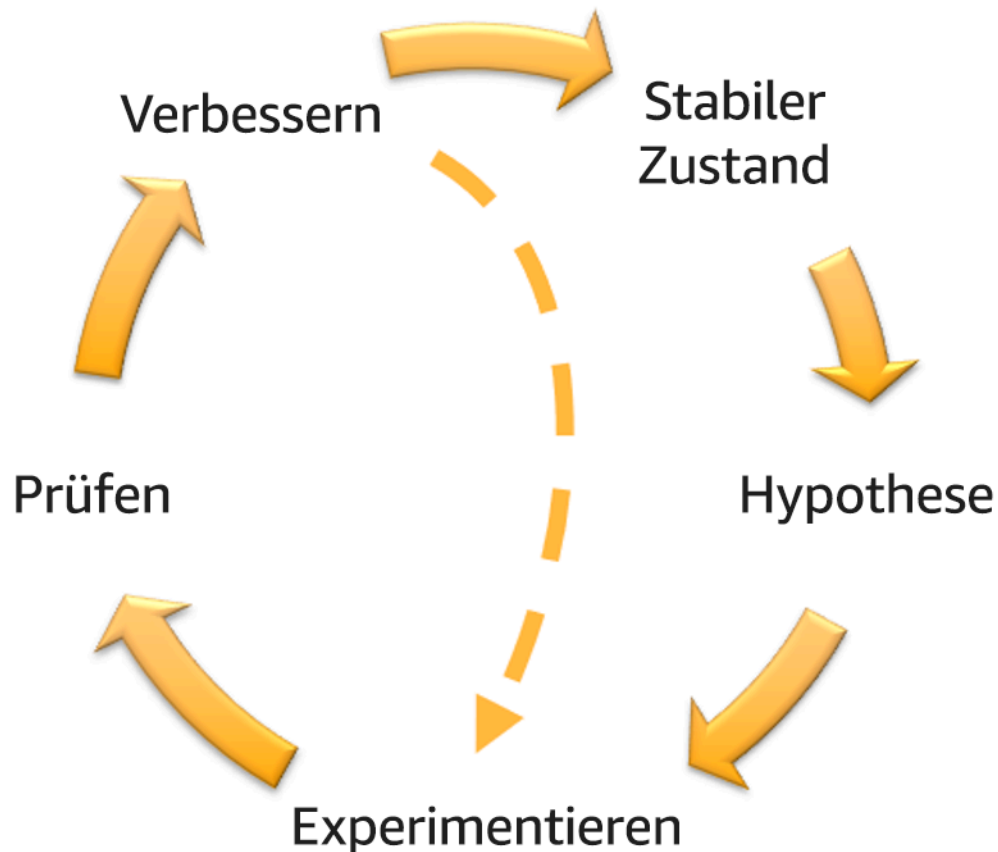
Analysieren Sie hinsichtlich der Häufigkeit eines bestimmten Fehlers frühere Daten für die betreffende Workload, wenn verfügbar. Wenn keine Daten verfügbar sind, verwenden Sie Daten zu anderen Workloads, die in einer ähnlichen Umgebung ausgeführt werden.

Bei der Betrachtung der Auswirkungen eines bestimmten Fehlers gilt, dass die Auswirkungen im Allgemeinen umso größer sind, je größer der vom Fehler betroffene Bereich ist. Sie sollten auch das Design und den Zweck der Workload berücksichtigen. Beispielsweise ist für eine Workload, die Daten transformiert und analysiert, der Zugriff auf die Quelldatenspeicher von kritischer Bedeutung. In diesem Fall würden Sie Experimente im Zusammenhang mit Zugriffsfehlern, Zugriffsdrosselungen und Latenzen priorisieren.

Nach Vorfällen durchgeführte Analysen stellen eine gute Datenquelle dar, um Häufigkeit und Auswirkungen von Fehlerarten besser zu verstehen.

Legen Sie anhand der zugewiesenen Priorität die Fehler fest, mit denen zuerst experimentiert werden soll, und die Reihenfolge, in der neue Fehlerinjektionsexperimente entwickelt werden sollen.

3. Für jedes von Ihnen ausgeführte Experiment sollten Sie sich am Schwungrad für Chaos-Engineering und kontinuierliche Resilienz in der folgenden Abbildung orientieren.



Schwungrad für Chaos-Engineering und kontinuierliche Resilienz unter Verwendung der wissenschaftlichen Methode von Adrian Hornsby.

- a. Definieren Sie den Steady-State als die messbare Ausgabe einer Workload, die ein normales Verhalten zeigt.

Ihre Workload befindet sich im Steady-State, wenn sie zuverlässig und wie erwartet ausgeführt wird. Daher sollten Sie die Integrität Ihrer Workload überprüfen, bevor Sie den Steady-State definieren. Steady-State bedeutet nicht notwendigerweise, dass sich ein Fehler nicht auf die Workload auswirkt, da ein bestimmter Prozentsatz an Fehlern innerhalb akzeptabler Grenzen liegen könnte. Der Steady-State ist die Basislinie, die Sie während des Experiments beobachten. Diese wird Anomalien aufweisen, wenn Ihre Hypothese, die Sie im nächsten Schritt definieren, nicht die erwarteten Ergebnisse zeigt.

Der Steady-State eines Zahlungssystems kann beispielsweise als die Verarbeitung von 300 TPS mit einer Erfolgsrate von 99 % und einer Roundtrip-Zeit von 500 ms definiert sein.

- b. Formulieren Sie eine Hypothese dazu, wie die Workload auf den Fehler reagieren wird.

Eine gute Hypothese basiert darauf, wie die Workload den Fehler voraussichtlich bewältigt, um den Steady-State zu wahren. Die Hypothese besagt, dass bei einem Fehler eines spezifischen Typs das System oder die Workload weiter im Steady-State bleibt, da die Workload mit bestimmten Resilienzmerkmalen entworfen wurde. Der spezifische Fehlertyp und die Fehlerbewältigung sollten in der Hypothese angegeben werden.

Sie können für die Hypothese die folgende Vorlage verwenden (andere Formulierungen sind jedoch auch akzeptabel):

 Note

Wenn ein *bestimmter Fehler* auftritt, beschreibt die *Workload-Name*-Workload *Maßnahmen zur Minderung der Auswirkungen*, um die *Auswirkungen auf geschäftliche oder technische Kennzahlen* beizubehalten.

Beispiel:

- Wenn 20 % der Knoten in der Amazon EKS-Knotengruppe ausfallen, wird die Transaction Create API das 99. Perzentil der Anforderungen weiter in weniger als 100 ms erfüllen (Steady-State). Die Amazon EKS-Knoten werden innerhalb von fünf Minuten wiederhergestellt und die Pods werden geplant und verarbeiten Datenverkehr innerhalb von acht Minuten nach der Einleitung des Experiments. Warnungen werden innerhalb von drei Minuten ausgelöst.
- Wenn eine einzelne Amazon EC2-Instance ausfällt, veranlasst die Elastic Load Balancing-Zustandsprüfung des Bestellsystems Elastic Load Balancing, Anforderungen ausschließlich an die noch intakten Instances zu senden, während Amazon EC2 Auto Scaling die ausgefallene Instance ersetzt. Dabei kommt es zu einer Steigerung der serverseitigen Fehler (5xx) um weniger als 0,01 % (Steady-State).
- Wenn die primäre Amazon RDS-Datenbank-Instance ausfällt, führt die Workload für die Erfassung von Lieferkettendaten einen Failover aus und stellt eine Verbindung zur Amazon RDS-Standby-Datenbank-Instance her, sodass es für weniger als 1 Minute zu Lese- oder Schreibfehlern für die Datenbank kommt (Steady-State).

c. Führen Sie das Experiment aus, indem Sie den Fehler injizieren.

Ein Experiment sollte grundsätzlich nicht zu einem Ausfall führen und von der Workload toleriert werden. Wenn Sie wissen, dass die Workload ausfallen wird, sollten Sie das Experiment nicht durchführen. Das Chaos-Engineering sollte verwendet werden, um bekannt-unbekannte oder unbekannt-unbekannte Ereignisse zu untersuchen. Bekannt-unbekannte Ereignisse sind Dinge, derer Sie sich bewusst sind, die Sie aber nicht vollständig verstehen, und unbekannt-unbekannte Ereignisse sind Dinge, derer Sie sich nicht bewusst sind und die Sie auch nicht vollständig verstehen. Wenn Sie Experimente für eine Workload ausführen, von der Sie wissen, dass sie fehlerhaft ist, werden Sie keine neuen Erkenntnisse gewinnen. Ihr Experiment sollte sorgfältig geplant sein, einen klaren Wirkungsumfang besitzen und einen Rollback-Mechanismus besitzen, der bei unerwarteten Störungen angewendet werden kann. Wenn eine sorgfältige Überprüfung zeigt, dass Ihre Workload das Experiment überstehen sollte, können Sie das Experiment starten. Für die Injektion von Fehlern gibt es verschiedene Optionen. Für Workloads in AWS bietet [AWS FIS](#) viele vordefinierte Fehlersimulationen, die als [Aktionen](#) bezeichnet werden. Sie können auch angepasste Aktionen für AWS FIS definieren, die mithilfe von [AWS Systems Manager-Dokumenten](#) ausgeführt werden.

Wir raten davon ab, angepasste Skripts für Chaos-Experimente zu verwenden, es sei denn, die Skripts können den aktuellen Zustand der Workload erkennen, können Protokolle ausgeben und stellen Rollback-Mechanismen und Stoppbedingungen bereit, soweit möglich.

Ein effektives Framework oder Toolset, das Chaos-Engineering unterstützt, sollte den aktuellen Status des Experiments nachverfolgen, Protokolle ausgeben und Rollback-Mechanismen bereitstellen, um eine kontrollierte Durchführung des Experiments zu unterstützen. Beginnen Sie mit einem verbreitet verwendeten Service wie AWS FIS, der Ihnen die Ausführung von Experimenten mit einem klar definierten Umfang ermöglicht und Sicherheitsmechanismen bereitstellt, um ein Experiment rückgängig machen zu können, wenn es zu unerwarteten Störungen führt. Weitere Informationen zu einer größeren Vielfalt von Experimenten mit AWS FIS finden Sie auch im Lab [Resiliente und Well-Architected-Apps mit Chaos-Engineering](#). Außerdem analysiert [AWS Resilience Hub](#) Ihre Workload und erstellt Experimente, die Sie in AWS FIS implementieren und ausführen können.

 Note

Sie sollten den Umfang und die Auswirkungen jedes Experiments genau verstehen. Wir empfehlen, Fehler zunächst in einer Nichtproduktionsumgebung zu simulieren, bevor sie in der Produktion ausgeführt werden.

Experimente sollten in der Produktion unter realer Last ausgeführt werden, wobei [Canary-Bereitstellungen](#) verwendet werden sollten, die sowohl eine Steuerung als auch eine experimentelle Systembereitstellung ermöglichen, sofern dies möglich ist. Die Ausführung von Experimenten außerhalb von Spitzenzeiten stellt ein empfehlenswertes Verfahren dar, um potenzielle Auswirkungen zu reduzieren, wenn ein Experiment zum ersten Mal in der Produktion durchgeführt wird. Wenn die Verwendung von tatsächlichem Kunden-Traffic ein zu großes Risiko darstellt, können Sie unter Verwendung der Kontroll- und Experimentbereitstellungen Experimente mit synthetischem Datenverkehr in der Produktionsinfrastruktur durchführen. Wenn ein Experiment nicht in der Produktion ausgeführt werden kann, führen Sie es in einer Präproduktionsumgebung aus, die der Produktionsumgebung so nahe wie möglich ist.

Sie müssen einen Integritätsschutz einrichten und überwachen, um sicherzustellen, dass sich das Experiment nicht jenseits akzeptabler Grenzen auf den Datenverkehr in der Produktionsumgebung oder andere Systeme auswirkt. Richten Sie Stoppbedingungen ein, um ein Experiment anhalten zu können, wenn es in einer Integritätsschutz-Metrik einen von Ihnen definierten Schwellenwert erreicht. Diese Metriken sollten die Metrik für den Steady-State der Workload und die Metrik für die Komponenten einschließen, in die Sie den Fehler injizieren. Die [synthetische Überwachung](#) (auch als Benutzer-Canary bezeichnet) gehört zu den Metriken, die Sie in der Regel als Benutzer-Proxy einschließen sollten. [Stoppbedingungen für AWS FIS](#) werden als Teil der Experimentvorlage unterstützt. Es sind bis zu fünf Stoppbedingungen pro Vorlage möglich.

Zu den Grundsätzen des Chaos-Engineering gehört die Minimierung von Umfang und Auswirkungen des Experiments:

Auch wenn einige kurzfristige negative Auswirkungen zulässig sein sollten, ist der Chaos-Engineer dafür verantwortlich, die Auswirkungen der Experimente zu minimieren und einzudämmen.

Eine Methode für die Überprüfung des Umfangs und der möglichen Auswirkungen besteht darin, das Experiment statt in der Produktionsumgebung zunächst in einer Nichtproduktionsumgebung durchzuführen. Dabei wird überprüft, ob die Schwellenwerte für Stoppbedingungen während des Experiments wie vorgesehen aktiviert werden und ob das Experiment beobachtet werden kann, um Ausnahmen abzufangen.

Wenn Sie Fehlerinjektionsexperimente durchführen, müssen alle verantwortlichen Beteiligten gut informiert sein. Teilen Sie den betroffenen Teams mit, wann die Experimente durchgeführt werden und was zu erwarten ist. Dies können Operations-Teams, die für die Servicezuverlässigkeit verantwortlichen Teams und der Kundensupport sein. Stellen Sie diesen Teams Kommunikationstools bereit, damit sie das Team, das das Experiment durchführt, über nachteilige Auswirkungen informieren können.

Sie müssen nach dem Experiment die Workload und die zugrunde liegenden Systeme wieder in den ursprünglichen, gut funktionierenden Zustand zurückversetzen. Häufig führt das resiliente Design der betreffenden Workload eine Selbstreparatur durch. Einige Fehlerdesigns oder fehlgeschlagenen Experimente können Ihre Workload jedoch in einem nicht erwarteten Fehlerzustand zurücklassen. Nach dem Ende des Experiments müssen Sie dies erkennen und die Workload und die Systeme wiederherstellen können. Mit AWS FIS können Sie eine Rollback-Konfiguration innerhalb der Aktionsparameter einrichten (auch als „Post-Aktion“ bezeichnet). Eine Post-Aktion führt das Ziel in den Zustand zurück, in dem es sich vor Ausführung der Aktion befunden hat. Ob automatisiert (bei Verwendung von AWS FIS) oder manuell – diese Post-Aktionen sollten Teil eines Playbooks sein, das die Erkennung und Behandlung von Fehlern und Ausfällen beschreibt.

d. Prüfen Sie die Hypothese.

Unter [Grundlagen des Chaos-Engineering](#) wird die folgende Anleitung für die Verifizierung des Steady-State Ihrer Workload bereitgestellt:

Konzentrieren Sie sich auf die messbare Ausgabe des Systems und nicht auf seine internen Attribute. Messungen dieser Ausgabe über einen kurzen Zeitraum stellen einen Proxy für den Steady-State des Systems dar. Der Gesamtdurchsatz, die Fehlerraten und die Latenz-Perzentile des Systems könnten Metriken sein, die das Steady-State-Verhalten beschreiben. Durch die Konzentration auf die Verhaltensmuster des Systems während Experimenten überprüft das Chaos-Engineering, ob das System funktioniert, statt zu versuchen, die Art der Funktion zu validieren.

In unseren beiden Beispielen oben verwenden wir die Steady-State-Metrik einer Erhöhung von weniger als 0,01 % bei serverseitigen Fehlern (5xx) und von weniger als einer Minute, in der Datenbankschreib- und Lesefehler auftreten.

Die 5xx-Fehler stellen eine gute Metrik dar, da sie die Folge des Fehlermodus sind, dem ein Client der Workload direkt unterliegen wird. Die Messung der Datenbankfehler ist als direkte Folge des Fehlers gut als Metrik geeignet, sollte jedoch durch eine Messung der Client-Auswirkungen ergänzt werden, beispielsweise in Form von fehlgeschlagenen Kundenanfragen oder Fehlern im Client. Zusätzlich sollten Sie für alle APIs oder URIs, auf die der Client Ihrer Workload direkt zugreift, eine synthetische Überwachung einrichten (auch als Benutzer-Canary bezeichnet).

e. Verbessern Sie das Workload-Design hinsichtlich der Resilienz.

Wenn der Steady-State nicht bewahrt wurde, untersuchen Sie, wie das Workload-Design verbessert werden könnte, um den Fehler zu bewältigen. Wenden Sie dabei die bewährten Methoden der [AWS Well-Architected-Säule „Zuverlässigkeit“](#) an. Zusätzliche Anleitungen und Ressourcen finden Sie in der [AWS Builder's Library](#). Dort finden Sie unter anderem Artikel darüber, wie Sie Ihre [Zustandsprüfungen verbessern](#) oder [Wiederholungen mit Backoff in Ihrem Anwendungscode verwenden](#) können.

Führen Sie das Experiment nach der Implementierung dieser Änderungen erneut durch (angezeigt durch die gepunktete Linie im Flywheel für das Chaos-Engineering), um ihre Effektivität zu ermitteln. Wenn der Verifizierungsschritt zeigt, dass die Hypothese zutrifft, befindet sich die Workload im Steady-State und der Zyklus wird fortgesetzt.

4. Führen Sie regelmäßig Experimente durch.

Ein Chaos-Experiment ist ein Zyklus. Daher sollten Experimente regelmäßig als Teil des Chaos-Engineering durchgeführt werden. Wenn die Hypothese eines Experiments auf eine Workload zutrifft, sollte das Experiment automatisiert werden, um innerhalb Ihrer CI/CD-Pipeline kontinuierlich als Regression ausgeführt zu werden. Wie das geht, erfahren Sie in diesem Blogbeitrag [zur Durchführung von AWS FIS-Experimenten mit AWS CodePipeline](#). In diesem Lab zu wiederkehrenden [AWS FIS-Experimenten in einer CI/CD-Pipeline](#) können Sie praktische Erfahrungen sammeln.

Fehlerinjektionsexperimente sind auch Bestandteil von Gamedays (siehe [REL12-BP05 Regelmäßiges Durchführen von Gamedays](#)). Bei Gamedays wird ein Fehler oder Ereignis simuliert, um Systeme, Prozesse und die Reaktionen von Teams zu testen. Dabei sollen die

auszuführenden Aktionen vom Team wie im Fall eines außergewöhnlichen Ereignisses tatsächlich ausgeführt werden.

5. Erfassen und speichern Sie die Ergebnisse der Experimente.

Die Ergebnisse von Fehlerinjektionsexperimenten müssen erfasst und gespeichert werden. Erfassen Sie dabei alle notwendigen Daten (z. B. Zeit, Workload und Bedingungen), um die Ergebnisse und Trends von Experimenten später analysieren zu können. Beispiele für erfasste Ergebnisse können Screenshots von Dashboards, CSV-Versionen der Metrikdatenbank oder manuell eingegebene Aufzeichnungen von Ereignissen und Beobachtungen während des Experiments sein. Die [Protokollierung von Experimenten mit AWS FIS](#) kann Teil dieser Datenerfassung sein.

Ressourcen

Zugehörige bewährte Methoden:

- [REL08-BP03 Integrieren von Ausfallsicherheitstests in die Bereitstellung](#)
- [REL13-BP03 Testen der Implementierung der Notfallwiederherstellung zur Validierung](#)

Zugehörige Dokumente:

- [Was ist AWS Fault Injection Service?](#)
- [Was ist AWS Resilience Hub?](#)
- [Grundlagen des Chaos Engineering](#)
- [Chaos-Engineering: Planung Ihres ersten Experiments](#)
- [Resilience Engineering: Aus Fehlern lernen](#)
- [Chaos-Engineering-Geschichten](#)
- [Vermeiden von Fallback in verteilten Systemen](#)
- [Canary-Bereitstellung für Chaos-Experimente](#)

Zugehörige Videos:

- [AWS re:Invent 2020: Testing resiliency using chaos engineering \(ARC316\)](#)
- [AWS re:Invent 2019: Improving resiliency with chaos engineering \(DOP309-R1\)](#)
- [AWS re:Invent 2019: Performing chaos engineering in a serverless world \(CMY301\)](#)

Zugehörige Tools:

- [AWS Fault Injection Service](#)
- AWS Marketplace: [Gremlin Chaos Engineering Platform](#)
- [Chaos Toolkit](#)
- [Chaos Mesh](#)
- [Litmus](#)

REL12-BP05 Regelmäßiges Durchführen von Gamedays

Führen Sie Gamedays durch, um Ihre Verfahren zur Reaktion auf Ereignisse und Beeinträchtigungen, die sich auf die Workloads auswirken, regelmäßig zu trainieren. Beziehen Sie dieselben Teams ein, die für die Bearbeitung von Produktionsszenarien verantwortlich wären. Diese Übungen helfen, Maßnahmen durchzusetzen, um eine Beeinträchtigung der Benutzer durch Ereignisse in der Produktionsumgebung zu verhindern. Wenn Sie Ihre Reaktionsverfahren unter realistischen Bedingungen üben, können Sie etwaige Lücken oder Schwächen erkennen und beheben, bevor ein tatsächliches Ereignis eintritt.

Bei Gamedays werden Ereignisse in produktionsähnlichen Umgebungen simuliert, um Systeme, Prozesse und die Reaktionen des Teams zu testen. Dabei sollen die Aktionen ausgeführt werden, die das Team im Falle eines tatsächlichen Eintretens des Ereignisses ausführen würde. So können Sie nachvollziehen, wo nachgebessert werden kann, und üben, mit Ereignissen und Beeinträchtigungen umzugehen. Gamedays sollten regelmäßig ausgeführt werden, damit Ihr Team tief verwurzelte Reaktionsgewohnheiten aufbaut.

Gamedays bereiten die Teams darauf vor, Produktionsereignisse mit größerer Sicherheit zu bewältigen. Geübte Teams sind besser in der Lage, verschiedene Szenarien schnell zu erkennen und darauf zu reagieren. Dies führt zu einer deutlich besseren Bereitschaft und Resilienz.

Gewünschtes Ergebnis: Sie veranstalten die Gamedays für Resilienz immer wieder zu geplanten Zeitpunkten. Diese Gamedays werden als normaler und erwarteter Teil der Geschäftstätigkeit angesehen. Ihr Unternehmen hat eine Kultur der Bereitschaft entwickelt, und wenn Produktionsprobleme auftreten, sind Ihre Teams gut darauf vorbereitet, effektiv zu reagieren, die Probleme effizient zu lösen und die Auswirkungen auf die Kunden zu minimieren.

Typische Anti-Muster:

- Sie dokumentieren die eigenen Verfahren, trainieren diese aber nie.

- Entscheidungsträger werden bei den Tests nicht mit einbezogen.
- Sie veranstalten einen Gameday, informieren aber nicht alle relevanten Stakeholder.
- Sie konzentrieren sich ausschließlich auf technische Fehler, beziehen aber geschäftliche Stakeholder nicht mit ein.
- Sie lassen die bei den Gamedays gewonnenen Erkenntnisse nicht in Ihre Wiederherstellungsprozesse einfließen.
- Sie geben den Teams die Schuld für Fehler oder Bugs.

Vorteile der Nutzung dieser bewährten Methode:

- Verbesserung der Reaktionsfähigkeit: An Gamedays üben die Teams ihre Aufgaben und testen ihre Kommunikationsmechanismen bei simulierten Ereignissen. So kommt es zu einer besser koordinierten, effizienteren Reaktion in Produktionssituationen.
- Identifizierung und Behebung von Abhängigkeiten: Komplexe Umgebungen beinhalten oft vielschichtige Abhängigkeiten zwischen verschiedenen Systemen, Services und Komponenten. Gamedays können Ihnen dabei helfen, diese Abhängigkeiten zu identifizieren und zu beheben und zu überprüfen, ob Ihre kritischen Systeme und Services ordnungsgemäß durch Ihre Runbook-Verfahren abgedeckt sind und zeitnah hochskaliert oder wiederhergestellt werden können.
- Förderung einer Kultur der Resilienz: Gamedays können dazu beitragen, eine Haltung der Resilienz innerhalb eines Unternehmens zu fördern. Wenn Sie funktionsübergreifende Teams und Stakeholder einbeziehen, fördern diese Übungen das Bewusstsein, die Zusammenarbeit und ein gemeinsames Verständnis für die Bedeutung von Resilienz im gesamten Unternehmen.
- Kontinuierliche Verbesserung und Anpassung: Regelmäßige Gamedays helfen Ihnen, Ihre Resilienzstrategien kontinuierlich zu bewerten und anzupassen, sodass sie auch bei sich ändernden Umständen relevant und wirksam bleiben.
- Stärkung des Vertrauens in das System: Erfolgreiche Gamedays tragen dazu bei, Vertrauen in die Fähigkeit des Systems aufzubauen, Störungen standzuhalten und sich von ihnen zu erholen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Sobald Sie die erforderlichen Resilienzmaßnahmen entwickelt und umgesetzt haben, führen Sie einen Gameday durch, um zu überprüfen, ob in der Produktion alles wie geplant funktioniert. An einem Gameday, insbesondere am ersten Gameday, sollten alle Teammitglieder beteiligt sein und

alle Stakeholder und Teilnehmer sollten vorab über das Datum, die Uhrzeit und die simulierten Szenarien informiert werden.

Während des Gamedays simulieren die beteiligten Teams verschiedene Ereignisse und mögliche Szenarien gemäß den vorgeschriebenen Verfahren. Die Teilnehmer beobachten und bewerten die Auswirkungen dieser simulierten Ereignisse genau. Wenn das System wie vorgesehen funktioniert, sollten die automatischen Erkennungs-, Skalierungs- und Selbstreparaturmechanismen aktiviert werden und das Ereignis sollte nur geringe bis keine Auswirkungen auf die Benutzer haben. Wenn das Team negative Auswirkungen feststellt, setzt es den Test zurück und behebt die festgestellten Probleme, entweder auf automatisiertem Wege oder durch manuelles Eingreifen, wie in den entsprechenden Runbooks dokumentiert.

Um die Resilienz kontinuierlich zu verbessern, ist es wichtig, die gewonnenen Erkenntnisse zu dokumentieren und zu berücksichtigen. Bei diesem Prozess handelt es sich um eine Feedback-Schleife, die systematisch Erkenntnisse aus Gamedays erfasst und zur Verbesserung von Systemen, Prozessen und Teamfähigkeiten nutzt.

Um leichter reale Szenarien reproduzieren zu können, in denen Systemkomponenten oder Services möglicherweise unerwartet ausfallen, können Sie simulierte Fehler als Gameday-Übung einbauen. Die Teams können die Resilienz und Fehlertoleranz ihrer Systeme testen und ihre Reaktions- und Wiederherstellungsprozesse in einer kontrollierten Umgebung simulieren.

In AWS können Ihre Gamedays mit Replikaten Ihrer Produktionsumgebung unter Verwendung von Infrastructure as Code durchgeführt werden. Auf diese Weise können Sie Ihre Tests in einer sicheren Umgebung durchführen, die Ihrer Produktionsumgebung sehr ähnlich ist. Ziehen Sie die Verwendung von [AWS Fault Injection Service](#) in Betracht, um verschiedene Ausfallszenarien zu erstellen. Verwenden Sie Services wie [Amazon CloudWatch](#) und [AWS X-Ray](#), um das Systemverhalten an Gamedays zu überwachen. Verwenden Sie [AWSSystems Manager](#) für die Verwaltung und Ausführung von Playbooks und [AWS Step Functions](#) zur Orchestrierung von wiederkehrenden Workflows am Gameday.

Implementierungsschritte

- Einrichtung eines Programms für den Gameday: Entwickeln Sie ein strukturiertes Programm, das die Häufigkeit, den Umfang und die Ziele der Gamedays festlegt. Binden Sie wichtige Stakeholder und Fachexperten in die Planung und Durchführung dieser Übungen ein.
- Vorbereitung des Gamedays:

1. Identifizieren Sie die wichtigsten geschäftskritischen Services, um die es bei dem Gameday gehen soll. Katalogisieren Sie die Mitarbeiter, Prozesse und Technologien, die diese Services unterstützen, und ordnen Sie sie zu.
 2. Legen Sie die Agenda für den Gameday fest und bereiten Sie die beteiligten Teams auf die Teilnahme an dem Event vor. Bereiten Sie Ihre Automatisierungsservices so vor, dass sie die geplanten Szenarien simulieren und die entsprechenden Wiederherstellungsprozesse ausführen. AWS-Services wie [AWS Fault Injection Service](#), [AWS Step Functions](#) und [AWS Systems Manager](#) können Ihnen dabei helfen, verschiedene Aspekte von Gamedays zu automatisieren, beispielsweise das Einschleusen von Fehlern und das Einleiten von Wiederherstellungsaktionen.
- Ausführung Ihrer Simulation: Führen Sie am Gameday das geplante Szenario aus. Beobachten und dokumentieren Sie, wie Mitarbeiter, Prozesse und Technologien auf das simulierte Ereignis reagieren.
 - Durchführung von Prüfungen nach der Übung: Führen Sie nach dem Gameday eine Retrospektive durch, um die gewonnenen Erkenntnisse zu überprüfen. Identifizieren Sie Bereiche mit Verbesserungspotenzial und alle Maßnahmen, die zur Verbesserung der betrieblichen Resilienz erforderlich sind. Dokumentieren Sie Ihre Ergebnisse und verfolgen Sie alle notwendigen Änderungen, um Ihre Resilienzstrategien und Ihre Bereitschaft zu verbessern.

Ressourcen

Zugehörige bewährte Methoden:

- [REL12-BP01 Untersuchen von Fehlern mit Playbooks](#)
- [REL12-BP04 Testen der Ausfallsicherheit mit Chaos-Engineering](#)
- [OPS04-BP01 Ermitteln wichtiger Leistungskennzahlen](#)
- [OPS07-BP03 Verwenden von Runbooks zur Durchführung von Verfahren](#)
- [OPS10-BP01 Verwenden eines Prozesses für die Bewältigung von Ereignissen, Vorfällen und Problemen](#)

Zugehörige Dokumente:

- [Was ist AWS GameDay?](#)

Zugehörige Videos:

- [AWS re:Invent 2023 – Practice like you play: How Amazon scales resilience to new heights](#)

Zugehörige Beispiele:

- [AWS Workshop – Den Sturm überstehen: Widerstandsfähige Systeme durch kontrolliertes Chaos](#)
- [Gestalten Ihres eigenen Gamedays zur Förderung der betrieblichen Resilienz](#)

REL 13. Was ist bei der Planung der Notfallwiederherstellung zu beachten?

Sicherungen und redundante Workload-Komponenten sind der Ausgangspunkt Ihrer Strategie für die Notfallwiederherstellung. [RTO und RPO sind Ihre Ziele](#) für die Wiederherstellung Ihrer Workload. Legen Sie diese entsprechend den geschäftlichen Anforderungen fest. Implementieren Sie eine Strategie, um diese Ziele zu erreichen. Berücksichtigen Sie dabei Standorte und Funktionen von Workload-Ressourcen und -Daten. Die Wahrscheinlichkeit von Unterbrechungen und die Kosten von Wiederherstellungen sind ebenfalls wichtige Faktoren bei der Ermittlung des Unternehmenswerts, den Notfallwiederherstellungen von Workloads bieten.

Best Practices

- [REL13-BP01 Definieren von Wiederherstellungszielen bei Ausfällen und Datenverlusten](#)
- [REL13-BP02 Verwenden von definierten Wiederherstellungsstrategien, um die Wiederherstellungsziele zu erreichen](#)
- [REL13-BP03 Testen der Implementierung der Notfallwiederherstellung zur Validierung](#)
- [REL13-BP04 Verwalten der Konfigurationsabweichungen am Standort oder in der Region der Notfallwiederherstellung](#)
- [REL13-BP05 Automatisieren der Wiederherstellung](#)

REL13-BP01 Definieren von Wiederherstellungszielen bei Ausfällen und Datenverlusten

Ausfälle können sich auf verschiedene Weise auf Ihr Unternehmen auswirken. Erstens können Ausfälle zu Betriebsunterbrechungen (Ausfallzeiten) führen. Zweitens können Ausfälle dazu führen, dass Daten verloren gehen, inkonsistent oder veraltet sind. Definieren Sie für jeden Workload ein Recovery Time Objective (RTO) und Recovery Point Objective (RPO). Das Recovery Time Objective (RTO) ist die maximal zulässige Verzögerung zwischen der Unterbrechung und der Wiederherstellung des Services. Das Recovery Point Objective (RPO) ist die maximal zulässige Zeitspanne seit dem letzten Datenwiederherstellungspunkt.

Gewünschtes Ergebnis: Für jeden Workload gibt es ein bestimmtes RTO und RPO, basierend auf technischen Überlegungen und geschäftlichen Auswirkungen.

Typische Anti-Muster:

- Sie verfügen nicht über festgelegte Wiederherstellungsziele.
- Sie wählen willkürliche Wiederherstellungsziele aus.
- Sie wählen Wiederherstellungsziele aus, die nicht strikt genug sind und die Geschäftsziele nicht erfüllen.
- Sie haben die Auswirkungen von Ausfallzeiten und Datenverlusten nicht bewertet.
- Sie wählen unrealistische Wiederherstellungsziele aus (beispielsweise sofortige Wiederherstellung oder kein Datenverlust), die für Ihre Workload-Konfiguration möglicherweise nicht erreichbar sind.
- Sie wählen Wiederherstellungsziele aus, die strikter sind als die tatsächlichen Geschäftsziele. Dies erzwingt Wiederherstellungsimplementierungen, die kostspieliger und komplizierter sind als für den Workload erforderlich.
- Sie wählen Wiederherstellungsziele aus, die nicht mit denen eines abhängigen Workloads vereinbar sind.
- Sie berücksichtigen gesetzliche Vorschriften und Compliance-Anforderungen nicht.

Vorteile der Einführung dieser bewährten Methode: Durch die Festlegung von RTOs und RPOs für Ihre Workloads definieren Sie klare und messbare Ziele für die Wiederherstellung auf der Grundlage Ihrer Geschäftsanforderungen. Sobald Sie diese Ziele festgelegt haben, können Sie Pläne für die Notfallwiederherstellung erstellen, die auf die Erreichung dieser Ziele zugeschnitten sind.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Erstellen Sie eine Matrix oder ein Arbeitsblatt, um eine bessere Planung der Notfallwiederherstellung zu ermöglichen. Erstellen Sie in Ihrer Matrix verschiedene Workload-Kategorien oder -stufen basierend auf den Auswirkungen der Workloads auf das Unternehmen (z. B. kritisch, hoch, mittel und gering) und den zugehörigen RTOs und RPOs, die Sie für jeden einzelnen Workload anstreben. Die folgende Matrix enthält ein Beispiel, dem Sie folgen können (beachten Sie, dass Ihre RTO- und RPO-Werte hiervon abweichen können):

Matrix der Notfallwiederherstellung						
		Wiederherstellungszeitpunkt				
		< 1 Minute	< 1 Stunde	< 6 Stunden	< 1 Tag	+ 1 Tag
Wiederherstellungsdauer	< 10 Minuten	Kritisch	Kritisch	Hoch	Mittel	Mittel
	< 2 Stunden	Kritisch	Hoch	Mittel	Mittel	Niedrig
	< 8 Stunden	Hoch	Mittel	Mittel	Niedrig	Niedrig
	< 24 Stunden	Mittel	Mittel	Niedrig	Niedrig	Niedrig
	24 + Stunden	Mittel	Niedrig	Niedrig	Niedrig	Niedrig

Beispielmatrix für die Notfallwiederherstellung

Sie müssen für jeden Workload die Auswirkungen von Ausfallzeiten und Datenverlusten auf Ihr Unternehmen ermitteln und verstehen. Die Auswirkungen werden in der Regel umso größer, je länger die Ausfallzeiten sind und je größer der Datenverlust ist, die Form der Auswirkung kann jedoch je nach Art des Workloads unterschiedlich sein. So kann es beispielsweise sein, dass Ausfallzeiten von bis zu einer Stunde geringe Auswirkungen haben, die Auswirkungen danach aber schnell zunehmen. Die Auswirkungen können viele verschiedene Formen annehmen. So kann es etwa finanzielle Auswirkungen (z. B. Umsatzeinbußen), Auswirkungen auf den Ruf (einschließlich Verlust von Kundenvertrauen), betriebliche Auswirkungen (z. B. verspätete Auszahlung von Gehältern oder verringerte Produktivität) und regulatorische Risiken geben. Weisen Sie den Workload nach Abschluss des Vorgangs der entsprechenden Stufe zu.

Berücksichtigen Sie bei der Analyse der Auswirkungen eines Fehlers die folgenden Fragen:

1. Wie lange kann der Workload maximal nicht verfügbar sein, bevor dies inakzeptable Auswirkungen auf das Unternehmen hat?
2. Wie stark werden die Auswirkungen einer Unterbrechung des Workloads auf das Unternehmen sein und welche Art von Auswirkungen wird es geben? Berücksichtigen Sie alle Arten von Auswirkungen, einschließlich finanzieller, den Ruf betreffender, betrieblicher und regulatorischer Auswirkungen.
3. Wie viele Daten können maximal verlorengehen oder nicht wiederherstellbar sein, bevor dies inakzeptable Auswirkungen auf das Unternehmen hat?

4. Können verlorene Daten aus anderen Quellen wiederhergestellt werden (sogenannte abgeleitete Daten)? Wenn ja, sollten Sie auch die RPOs aller Quelldaten berücksichtigen, die zur Neuerstellung der Workload-Daten verwendet werden.
5. Was sind die Wiederherstellungsziele und Verfügbarkeitserwartungen für Workloads, von denen dieser Workload abhängt (Downstream)? Die Ziele Ihres Workloads müssen unter Berücksichtigung der Wiederherstellungsmöglichkeiten seiner nachgelagerten Abhängigkeiten erreichbar sein. Erwägen Sie mögliche Behelfslösungen oder Abhilfemaßnahmen für nachgelagerte Abhängigkeiten, die die Wiederherstellungsfähigkeit dieses Workloads verbessern können.
6. Was sind die Wiederherstellungsziele und Verfügbarkeitserwartungen für Workloads, die von diesem Workload abhängen (Upstream)? Angesichts der Ziele vorgelagerter Workloads muss dieser Workload möglicherweise über striktere Wiederherstellungsmöglichkeiten verfügen, als es zunächst den Anschein hat.
7. Gibt es je nach Art des Vorfalls unterschiedliche Wiederherstellungsziele? Sie könnten beispielsweise unterschiedliche RTOs und RPOs haben, je nachdem, ob sich der Vorfall auf eine Availability Zone oder eine gesamte Region auswirkt.
8. Ändern sich Ihre Wiederherstellungsziele während bestimmter Ereignisse oder zu bestimmten Zeiten des Jahres? Sie könnten beispielsweise unterschiedliche RTOs und RPOs für das Weihnachtsgeschäft, für Sportveranstaltungen, Sonderverkaufsaktionen und Produkteinführungen haben.
9. Wie stimmen die Wiederherstellungsziele mit den Strategien Ihrer Branche und Ihres Unternehmens für die Notfallwiederherstellung überein?
10. Gibt es rechtliche oder vertragliche Konsequenzen zu beachten? Sind Sie beispielsweise vertraglich verpflichtet, einen Service mit einem bestimmten RTO oder RPO bereitzustellen? Welche Strafen könnten Ihnen bei einer Nichteinhaltung drohen?
11. Müssen Sie die Datenintegrität wahren, um gesetzliche Vorschriften oder Compliance-Anforderungen zu erfüllen?

Das folgende Arbeitsblatt kann Ihnen bei der Bewertung der einzelnen Workloads helfen. Sie können dieses Arbeitsblatt Ihren spezifischen Bedürfnissen entsprechend ändern, indem Sie beispielsweise zusätzliche Fragen hinzufügen.

Schritt 2: primäre Fragen	Gilt für Workload?	Workload-RTO	Workload-RPO	RTO anpassen	RPO anpassen	Anleitungen
[1] Maximale Zeit, in der der Workload ausfallen kann						Gemessen Zeit seit Beginn des Ausfalls bis zur Wiederherstellung
[2] Maximale Datenmenge, die verloren gehen kann						Gemessen in Zeit seit dem letzten bekannten gut wiederherstellbaren Datensatz
[3a] Vorgelagerte Abhängigkeiten						Strengste nachgelagerte Wiederherstellungsziele eingeben
[3b] Nachgelagerte Abhängigkeiten						Am wenigsten strenge nachgelagerte Wiederherstellungsziele eingeben
[3a] Abgeglichen vorgelagerte Abhängigkeiten						Wenn der vorgelagerte Wert niedriger ist als aktuelle Werte und der nachgelagerte Wert größer ist, arbeiten Sie mit Abhängigkeiten, um auszugleichen und hier ausgeglichene Werte einzugeben.
[3b] Abgeglichen nachgelagerte Abhängigkeiten						Werte senken, um vorgelagerte Abhängigkeiten zu erfüllen oder die basierend auf nachgelagerten Abhängigkeitsfähigkeiten zu erhöhen
[3] Abhängigkeiten						
Schritt 2: zusätzliche Fragen						Geben Sie an, ob die Frage zutrifft. Falls nicht, überspringen Sie sie.
Basis-RTO-/RPO						Übertragen Sie die RTO- und RPO-Werte von oben nach hier unten.
[4] Art des Ausfalls	[] [X] [] [N]					Geben Sie Wiederherstellungsziele für Ereignisarten mit strengsten Anforderungen ein.
[5] Spezifische zeitbasierte Ziele	[] [X] [] [N]					Geben Sie Wiederherstellungsziele für Zeiten mit strengsten Anforderungen ein.
[6] Unterbrechungen bei Kunden	[] [X] [] [N]					Grafische Darstellung der betroffenen Kunden in Abhängigkeit von der Ausfallzeit oder dem Datenverlust. Verwenden Sie dies, um das maximal zulässige RTO und RPO auf der Grundlage der Kundenauswirkungen einzugeben.
[7] Auswirkungen auf den Ruf	[] [X] [] [N]					Mit dem Unternehmen arbeiten, um die maximale RTO und den maximalen RPO basierend auf der Auswirkung auf die Reputation zu bestimmen
[8] Betriebliche Auswirkungen	[] [X] [] [N]					Geben Sie das maximale RTO und RPO auf der Grundlage der betrieblichen Auswirkungen ein.
[9] Organisatorische Ausrichtung	[] [X] [] [N]					Geben Sie das maximale RTO und RPO für Workloads dieses Typs gemäß den LOB- und Organisationsanforderungen ein.
[10] Vertragliche Verpflichtungen	[] [X] [] [N]					Geben Sie das maximale RTO und RPO auf der Grundlage der vertraglichen Verpflichtungen ein.
[11] Gesetzliche Vorschriften	[] [X] [] [N]					Geben Sie das maximale RTO und RPO auf der Grundlage der geltenden gesetzlichen Bestimmungen ein.
Ziel basierend auf zusätzlichen Fragen						Nehmen Sie den Mindestwert (strengerer Wert) aus den Fragen 4–11 und geben Sie ihn hier ein.
Angepasstes Ziel						Wenn die Ziele in der obigen Zeile nicht erreicht werden können, arbeiten Sie mit den Beteiligten zusammen, um die Beschränkungen zu lockern, und geben Sie hier ein neues Minimum ein.
RTO/RPO angepasst						Geben Sie die Basis-RPO-/RTO-Werte oder das angepasste Ziel ein, je nachdem, welcher Wert niedriger ist.
Schritt 3						
Zuordnung zu vordefinierter Kategorie oder Stufe						Senken Sie beide Werte (machen Sie sie strenger), um sie an die nächstgelegene definierte Stufe anzupassen.

Arbeitsblatt

Implementierungsschritte

1. Ermitteln Sie die geschäftlichen Stakeholder und technischen Teams, die für die einzelnen Workloads verantwortlich sind, und setzen Sie sich mit ihnen in Verbindung.
2. Erstellen Sie Kategorien oder Stufen der Kritikalität für die Workload-Auswirkungen in Ihrem Unternehmen. Beispielkategorien sind u. a. „Kritisch“, „Hoch“, „Mittel“ und „Niedrig“. Wählen Sie für jede Kategorie ein Ihren Geschäftszielen und Anforderungen entsprechendes RTO und RPO.
3. Weisen Sie jedem Workload eine der im vorherigen Schritt erstellten Auswirkungskategorien zu. Berücksichtigen Sie für die Zuordnung eines Workloads zu einer Kategorie die Bedeutung des Workloads für das Unternehmen sowie die Auswirkungen von Unterbrechungen oder Datenverlusten und orientieren Sie sich an den obigen Fragen. So erhalten Sie für jeden Workload ein RTO und ein RPO.
4. Sehen Sie sich das im vorherigen Schritt ermittelte RTO und RPO für jeden Workload an. Beziehen Sie die geschäftlichen und technischen Teams für den Workload mit ein, um zu entscheiden, ob die Ziele angepasst werden sollten. Geschäftliche Stakeholder könnten beispielsweise zu dem Schluss kommen, dass strengere Ziele erforderlich sind. Alternativ könnten

technische Teams feststellen, dass die Ziele so geändert werden sollten, dass sie mit den verfügbaren Ressourcen und technologischen Einschränkungen erreichbar sind.

Ressourcen

Zugehörige bewährte Methoden:

- [REL09-BP04 Verifizieren der Sicherungsintegrität und -verfahren durch regelmäßiges Wiederherstellen der Daten](#)
- [REL12-BP01 Untersuchen von Fehlern mit Playbooks](#)
- [REL13-BP02 Verwenden von definierten Wiederherstellungsstrategien, um die Wiederherstellungsziele zu erreichen](#)
- [REL13-BP03 Testen der Implementierung der Notfallwiederherstellung zur Validierung](#)

Zugehörige Dokumente:

- [AWS Architecture Blog: Notfallwiederherstellung](#)
- [Notfallwiederherstellung von Workloads in AWS: Wiederherstellung in der Cloud \(AWS-Whitepaper\)](#)
- [Verwalten von Ausfallsicherheitsrichtlinien mit AWS Resilience Hub](#)
- [APN-Partner: Partner, die Sie bei der Notfallwiederherstellung unterstützen können](#)
- [AWS Marketplace: Für die Notfallwiederherstellung geeignete Produkte](#)

Zugehörige Videos:

- [AWS re:Invent 2018: Architecture Patterns for Multi-Region Active-Active Applications](#)
- [Notfallwiederherstellung von Workloads in AWS](#)

REL13-BP02 Verwenden von definierten Wiederherstellungsstrategien, um die Wiederherstellungsziele zu erreichen

Definieren Sie eine Notfallwiederherstellungsstrategie (Disaster Recovery, DR), die den Wiederherstellungszielen Ihrer Workloads entspricht. Wählen Sie eine Strategie aus, z. B. Backup und Wiederherstellung, Standby (aktiv/passiv) oder Aktiv/Aktiv.

Gewünschtes Ergebnis: Für jede Workload gibt es eine definierte und implementierte DR-Strategie, mit der die Workload die DR-Ziele erreichen kann. DR-Strategien zwischen Workloads nutzen wiederverwendbare Muster (wie die zuvor beschriebenen Strategien),

Typische Anti-Muster:

- Implementierung von inkonsistenten Wiederherstellungsprozeduren für Workloads mit ähnlichen DR-Zielen.
- Die DR-Strategie muss im Notfall Ad-hoc umgesetzt werden.
- Es gibt keinen Plan für die Notfallwiederherstellung.
- Abhängigkeit von Vorgängen auf der Steuerebene während der Wiederherstellung.

Vorteile der Nutzung dieser bewährten Methode:

- Durch die Nutzung definierter Wiederherstellungsstrategien können Sie verbreitet verwendete Tools und Testverfahren verwenden.
- Die Verwendung definierter Wiederherstellungsstrategien verbessert den Wissensaustausch zwischen den Teams und die Implementierung der Notfallwiederherstellung für ihre Workloads.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch. Ohne eine geplante, implementierte und getestete DR-Strategie ist es unwahrscheinlich, dass Sie Ihre Wiederherstellungsziele im Falle eines Notfalls erreichen.

Implementierungsleitfaden

Eine DR-Strategie beruht auf der Fähigkeit, Ihre Workload an einem Wiederherstellungsstandort bereitzustellen, wenn Ihr primärer Standort nicht mehr in der Lage ist, die Workload auszuführen. Die häufigsten Wiederherstellungsziele sind RTO und RPO, wie in [REL13-BP01 Definieren von Wiederherstellungszielen bei Ausfällen und Datenverlusten](#) besprochen.

Eine DR-Strategie, die mehrere Availability Zones (AZs) innerhalb eines einzigen AWS-Region umfasst, kann Katastrophenereignisse wie Brände, Überschwemmungen und größere Stromausfälle abfedern. Wenn es erforderlich ist, einen Schutz gegen ein unwahrscheinliches Ereignis zu implementieren, das verhindert, dass Ihre Workload in einer bestimmten AWS-Region ausgeführt werden kann, können Sie eine DR-Strategie verwenden, die mehrere Regionen nutzt.

Wenn Sie eine DR-Strategie für mehrere Regionen entwickeln, sollten Sie eine der folgenden Strategien wählen. Sie sind in aufsteigender Reihenfolge nach Kosten und Komplexität und in

absteigender Reihenfolge nach RTO und RPO aufgeführt. Die Wiederherstellungsregion bezieht sich auf eine andere AWS-Region als die primäre Region, die für Ihre Workload verwendet wird.

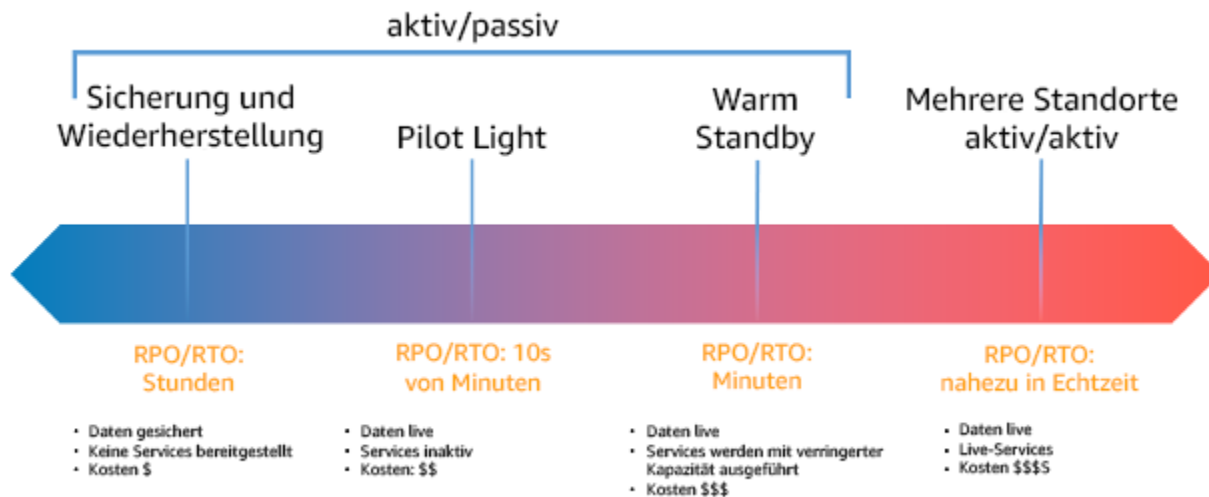


Abbildung 17: Strategien für die Notfallwiederherstellung (Disaster Recovery, DR)

- **Sicherung und Wiederherstellung** (RPO in Stunden, RTO in 24 Stunden oder weniger): Sichern Sie Ihre Daten und Anwendungen in der Wiederherstellungsregion. Die Verwendung automatisierter oder kontinuierlicher Backups ermöglicht eine zeitpunktbezogene Wiederherstellung (PITR), wodurch das RPO in einigen Fällen auf bis zu 5 Minuten gesenkt werden kann. Im Falle eines Notfalls stellen Sie Ihre Infrastruktur bereit (wobei Sie Infrastruktur als Code verwenden, um das RTO zu verkürzen), stellen Ihren Code bereit und stellen die gesicherten Daten wieder her, um eine Wiederherstellung nach einem Notfall in der Wiederherstellungsregion zu erfahren.
- **Pilot Light** (RPO in Minuten, RTO in 10-Minuten-Intervallen): Stellen Sie eine Kopie Ihrer Kern-Workload-Infrastruktur in der Wiederherstellungsregion bereit. Replizieren Sie Ihre Daten in die Wiederherstellungsregion und erstellen Sie dort Sicherungskopien der Daten. Ressourcen, die zur Unterstützung der Datenreplikation und -sicherung erforderlich sind, wie Datenbanken und Objektspeicher, sind immer eingeschaltet. Andere Elemente wie Anwendungsserver oder Serverless-Datenverarbeitung werden nicht bereitgestellt, sondern können bei Bedarf mit der erforderlichen Konfiguration und dem Anwendungscode erstellt werden.
- **Warm Standby** (RPO in Sekunden, RTO in Minuten): Eine herunterskalierte, aber voll funktionsfähige Version Ihrer Workload wird dauerhaft in der Wiederherstellungsregion ausgeführt. Geschäftskritische Systeme sind vollständig dupliziert und ständig aktiv, aber mit herunterskalierter Flotte. Die Daten werden repliziert und sind in der Wiederherstellungsregion live. Wenn eine

Wiederherstellung erforderlich ist, wird das System zur Bewältigung der Produktionslast schnell hochskaliert. Je höher die Skalierung des Warm Standby, desto geringer ist die Abhängigkeit von RTO und Steuerebene. Bei vollständiger Skalierung wird dies als Hot Standby bezeichnet.

- Regionsübergreifend (Multi-Site) Aktiv/Aktiv (RPO nahe Null, RTO potenziell Null): Ihre Workload wird auf mehrere AWS-Regionen verteilt und stellt aktiv Datenverkehr daraus aus. Bei dieser Strategie müssen Sie die Daten zwischen den Regionen synchronisieren. Mögliche Konflikte, die durch Schreibvorgänge auf denselben Datensatz in zwei verschiedenen regionalen Repliken verursacht werden, müssen vermieden oder behandelt werden, was sehr komplex sein kann. Die Datenreplikation ist nützlich für die Datensynchronisation und schützt Sie vor einigen Arten von Notfällen, aber sie schützt Sie nicht vor Datenbeschädigung oder -zerstörung, es sei denn, Ihre Lösung umfasst auch Optionen für eine zeitpunktbezogene Wiederherstellung.

Note

Der Unterschied zwischen Pilot Light und Warm Standby kann schwer zu überblicken sein. Beide beinhalten eine Umgebung in Ihrer Wiederherstellungsregion mit Kopien der Assets Ihrer Primärregion. Der Unterschied besteht darin, dass Pilot Light keine Anfragen bearbeiten kann, ohne dass zuvor zusätzliche Maßnahmen ergriffen werden, während Warm Standby den Datenverkehr (mit reduzierter Kapazität) sofort bearbeiten kann. Bei Pilot Light müssen Sie die Server einschalten, möglicherweise zusätzliche (nicht zum Kerngeschäft gehörende) Infrastruktur bereitstellen und die Leistung hochskalieren, während Sie bei Warm-Standby nur die Leistung hochskalieren müssen (alles ist bereits bereitgestellt und läuft). Wählen Sie je nach RTO- und RPO-Anforderungen zwischen diesen Varianten.

Wenn die Kosten eine Rolle spielen und Sie ähnliche RPO- und RTO-Ziele wie bei der Warm-Standby-Strategie erreichen möchten, könnten Sie cloudnative Lösungen wie AWS Elastic Disaster Recovery in Betracht ziehen, die den Pilot-Light-Ansatz verfolgen und bessere RPO- und RTO-Ziele bieten.

Implementierungsschritte

1. Bestimmen Sie eine DR-Strategie, die die Wiederherstellungsanforderungen für diese Workload erfüllt.

Die Wahl einer DR-Strategie ist eine Abwägung zwischen der Reduzierung von Ausfallzeiten und Datenverlusten (RTO und RPO) und den Kosten und der Komplexität der Implementierung

der Strategie. Sie sollten vermeiden, eine Strategie zu verfolgen, die strikter ist als nötig, da dies unnötige Kosten verursacht.

Im folgenden Diagramm hat das Unternehmen beispielsweise sein maximal zulässiges RTO sowie die Grenze der Ausgaben für seine Strategie zur Wiederherstellung von Diensten festgelegt. In Anbetracht der Ziele des Unternehmens erfüllen die DR-Strategien Pilot Light oder Warm Standby sowohl die RTO- als auch die Kostenkriterien.

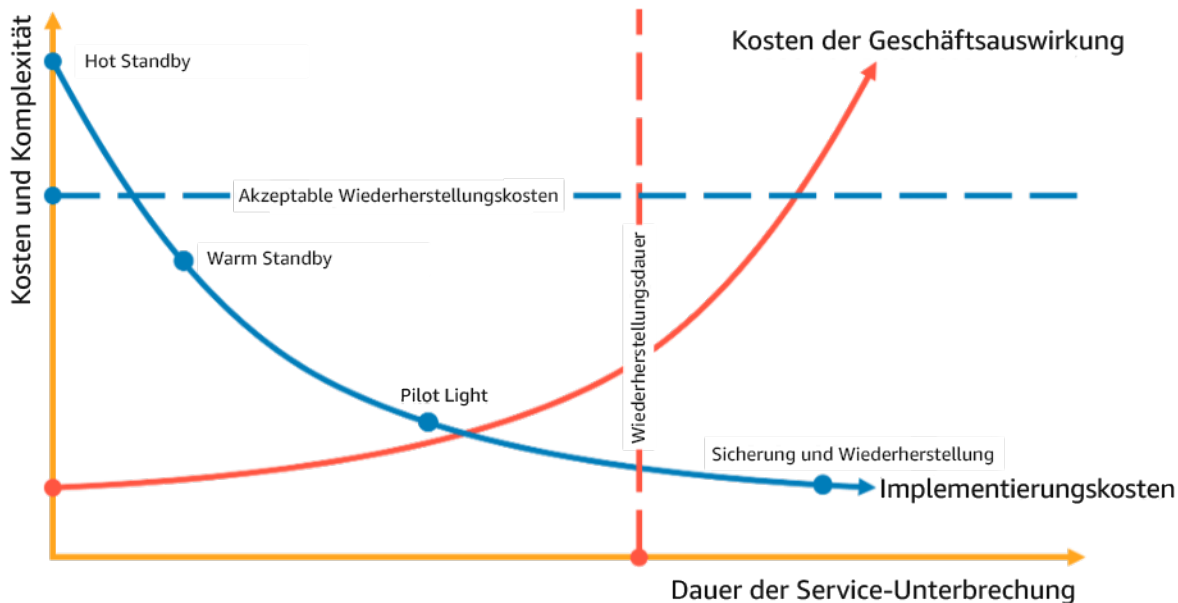


Abbildung 18: Auswahl einer DR-Strategie auf der Grundlage von RTO und Kosten

Weitere Informationen finden Sie unter [Geschäftsfortführungsplan](#).

2. Sehen Sie sich anhand der Muster an, wie die gewählte DR-Strategie umgesetzt werden kann.

In diesem Schritt geht es darum, zu verstehen, wie Sie die gewählte Strategie umsetzen wollen. Die Strategien werden durch die Verwendung von AWS-Regionen als primäre und Wiederherstellungsstandort erläutert. Sie können jedoch auch Availability Zones innerhalb einer einzigen Region als DR-Strategie verwenden, die Elemente mehrerer dieser Strategien nutzt.

In den folgenden Schritten können Sie die Strategie auf Ihre spezifische Workload anwenden.

Backup und Wiederherstellung

Backup und Wiederherstellung ist die einfachste Strategie zur Implementierung, erfordert jedoch mehr Zeit- und Arbeitsaufwand bei der Wiederherstellung der Workload, was zu einem höheren

RTO und RPO führt. Es ist eine gute Vorgehensweise, immer Sicherungskopien Ihrer Daten zu erstellen und diese auf einen anderen Standort (z. B. einen anderen AWS-Region) zu kopieren.

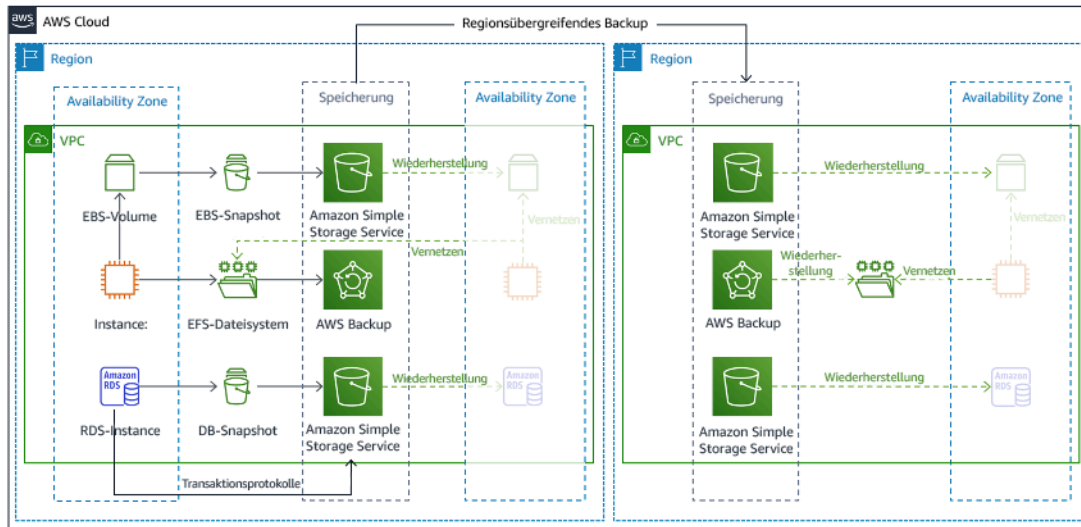


Abbildung 19: Backup- und Wiederherstellungsarchitektur

Weitere Informationen zu dieser Strategie finden Sie unter [Architektur für die Notfallwiederherstellung in AWS, Teil II: Sicherung und Wiederherstellung mit Rapid Recovery](#).

Pilot light

Beim Pilot-Light-Ansatz replizieren Sie die Daten von Ihrer primären Region in Ihre Wiederherstellungsregion. Die Kernressourcen, die für die Workload-Infrastruktur verwendet werden, werden in der Wiederherstellungsregion bereitgestellt, jedoch werden noch zusätzliche Ressourcen und Abhängigkeiten benötigt, um diesen Stack funktionsfähig zu machen. In Abbildung 20 werden zum Beispiel keine Datenverarbeitungs-Instances bereitgestellt.

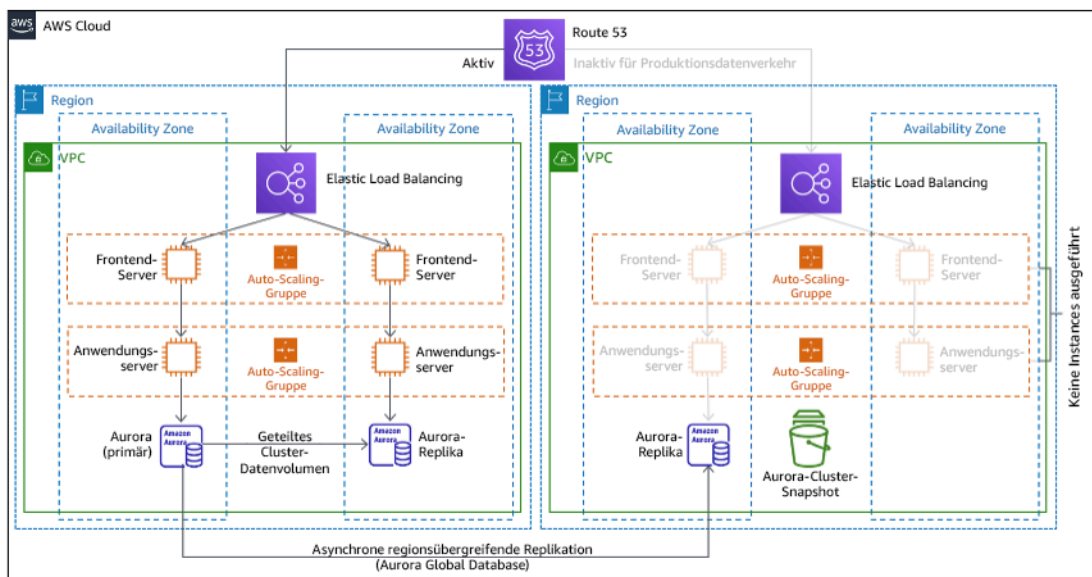


Abbildung 20: Pilot-Light-Architektur

Weitere Informationen zu dieser Strategie finden Sie unter [Architektur für die Notfallwiederherstellung in AWS, Teil III: Pilot Light und Warm Standby](#).

Warmer Bereitschaftsmodus

Beim Warm-Standby-Ansatz wird sichergestellt, dass eine herunterskalierte, aber voll funktionsfähige Kopie Ihrer Produktionsumgebung in einer anderen Region vorhanden ist. Dieser Ansatz erweitert das Konzept des Pilot Light und verkürzt die Zeit bis zur Wiederherstellung, da die Workload in einer anderen Region ständig präsent ist. Wenn die Wiederherstellungsregion mit voller Kapazität bereitgestellt wird, wird dies als Hot Standby bezeichnet.

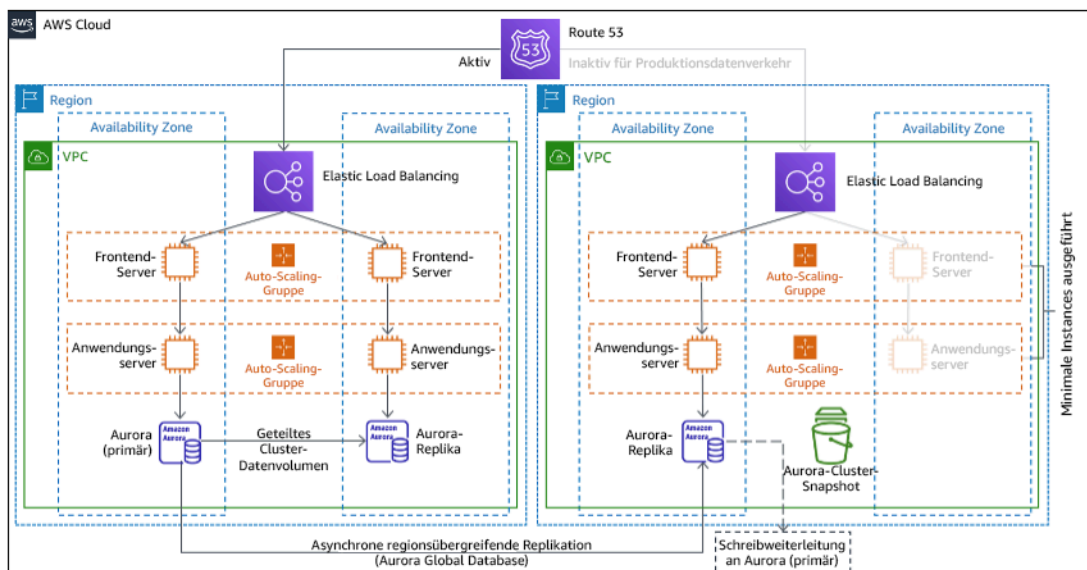


Abbildung 21: Warm-Standby-Architektur

Der Einsatz von Warm Standby oder Pilot Light erfordert ein Hochskalieren der Ressourcen in der Wiederherstellungsregion. Um zu überprüfen, ob bei Bedarf Kapazität verfügbar ist, sollten Sie die Nutzung von [Kapazitätsreservierungen](#) für EC2-Instances in Betracht ziehen. Wenn Sie AWS Lambda verwenden, kann die [bereitgestellte Parallelität](#) Laufzeitumgebungen bereitstellen, damit sie sofort auf die Aufrufe Ihrer Funktion reagieren können.

Weitere Informationen zu dieser Strategie finden Sie unter [Architektur für die Notfallwiederherstellung in AWS, Teil III: Pilot Light und Warm Standby](#).

Multi-Site Aktiv/Aktiv

Im Rahmen einer Multi-Site Aktiv/Aktiv-Strategie können Sie Ihre Workload in mehreren Regionen gleichzeitig ausführen. Multi-Site Aktiv/Aktiv bedient den Datenverkehr aus allen Regionen, in denen es eingesetzt wird. Diese Strategie kann zur Erhöhung der Verfügbarkeit oder bei der Bereitstellung einer Workload für eine globale Zielgruppe verwendet werden (um den Endpunkt näher an die Benutzer zu bringen und/oder um Stacks bereitzustellen, die für die Zielgruppe in dieser Region lokalisiert sind). Wenn die Workload in einer der AWS-Regionen, in denen sie bereitgestellt wird, nicht unterstützt werden kann, wird diese Region evakuiert und die verbleibenden Regionen werden zur Aufrechterhaltung der Verfügbarkeit genutzt. Multi-Site Aktiv/Aktiv ist die betrieblich komplexeste der DR-Strategien und sollte nur dann gewählt werden, wenn die Geschäftsanforderungen dies erfordern.

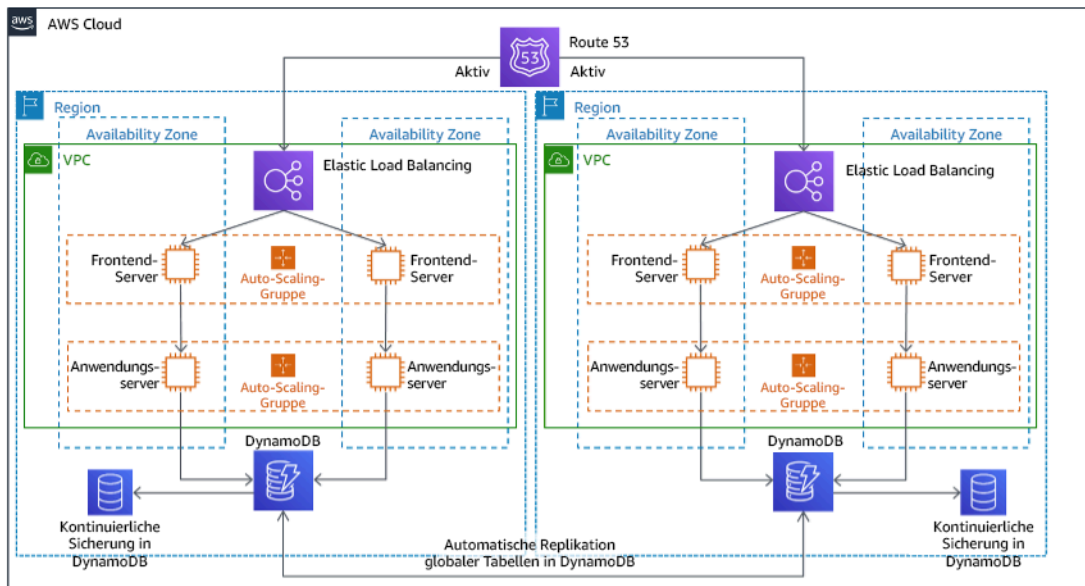


Abbildung 22: Multi-Site Aktiv/Aktiv-Architektur

Weitere Informationen zu dieser Strategie finden Sie unter [Architektur für die Notfallwiederherstellung in AWS, Teil IV: Multi-Site Aktiv/Aktiv](#).

AWS Elastic Disaster Recovery

Wenn Sie für die Notfallwiederherstellung die Pilot-Light- oder Warm-Standby-Strategie in Betracht ziehen, könnte AWS Elastic Disaster Recovery einen alternativen Ansatz mit verbesserten Vorteilen bieten. Elastic Disaster Recovery kann ein ähnliches RPO- und RTO-Ziel wie Warm Standby bieten, behält aber den kostengünstigen Ansatz von Pilot Light bei. Elastic Disaster Recovery repliziert Ihre Daten von Ihrer primären Region auf Ihre Wiederherstellungsregion und nutzt dabei die kontinuierliche Datensicherung, um ein RPO im Sekundenbereich und ein RTO im Minutenbereich zu erreichen. In der Wiederherstellungsregion werden nur die für die Replikation der Daten erforderlichen Ressourcen bereitgestellt, was die Kosten ähnlich wie bei der Pilot-Light-Strategie niedrig hält. Bei Verwendung von Elastic Disaster Recovery koordiniert und orchestriert der Service die Wiederherstellung von Datenverarbeitungs-Ressourcen, wenn die Initiierung als Teil eines Failover oder Drills erfolgt.

AWS Elastic Disaster Recovery (AWS DRS) – grundlegende Architektur

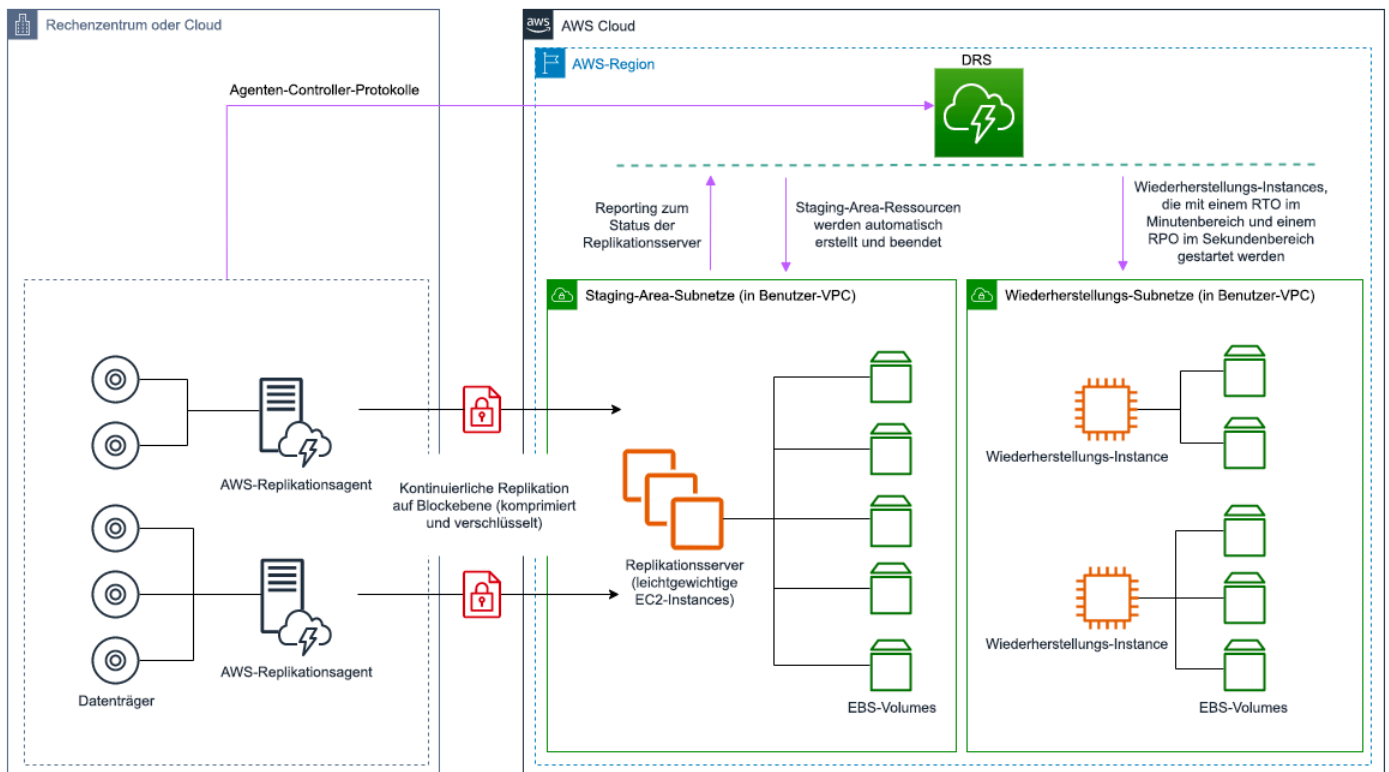


Abbildung 23: AWS Elastic Disaster Recovery-Architektur

Zusätzliche Methoden zum Schutz von Daten

Bei allen Strategien müssen Sie sich auch gegen einen Datennotfall wappnen. Kontinuierliche Datenreplikation schützt Sie vor einigen Arten von Notfällen, aber sie schützt Sie möglicherweise nicht vor Datenbeschädigung oder -zerstörung, es sei denn, Ihre Strategie umfasst auch die Versionsverwaltung gespeicherter Daten oder Optionen für eine zeitpunktbezogene Wiederherstellung. Sie müssen auch die replizierten Daten in der Wiederherstellungssite sichern, um zusätzlich zu den Replikaten zeitpunktgenaue Sicherungen zu erstellen.

Verwendung mehrerer Availability Zones (AZs) innerhalb einer einzigen AWS-Region

Wenn Sie mehrere AZs in einer einzigen Region verwenden, nutzt Ihre DR-Implementierung mehrere Elemente der oben genannten Strategien. Zunächst müssen Sie eine Hochverfügbarkeitsarchitektur (High Availability, HA) mit mehreren AZs erstellen, wie in Abbildung 23 dargestellt. Diese Architektur verwendet einen Multi-Site Aktiv/Aktiv-Ansatz, da die [Amazon](#)

[EC2-Instances](#) und der [Elastic Load Balancer](#) über Ressourcen verfügen, die in mehreren AZs bereitgestellt werden und Anfragen aktiv bearbeiten. Die Architektur demonstriert auch Hot Standby, d. h. bei einem Ausfall der primären [Amazon RDS](#)-Instance (oder der AZ selbst) wird die Standby-Instance zur primären Instance hochgestuft.

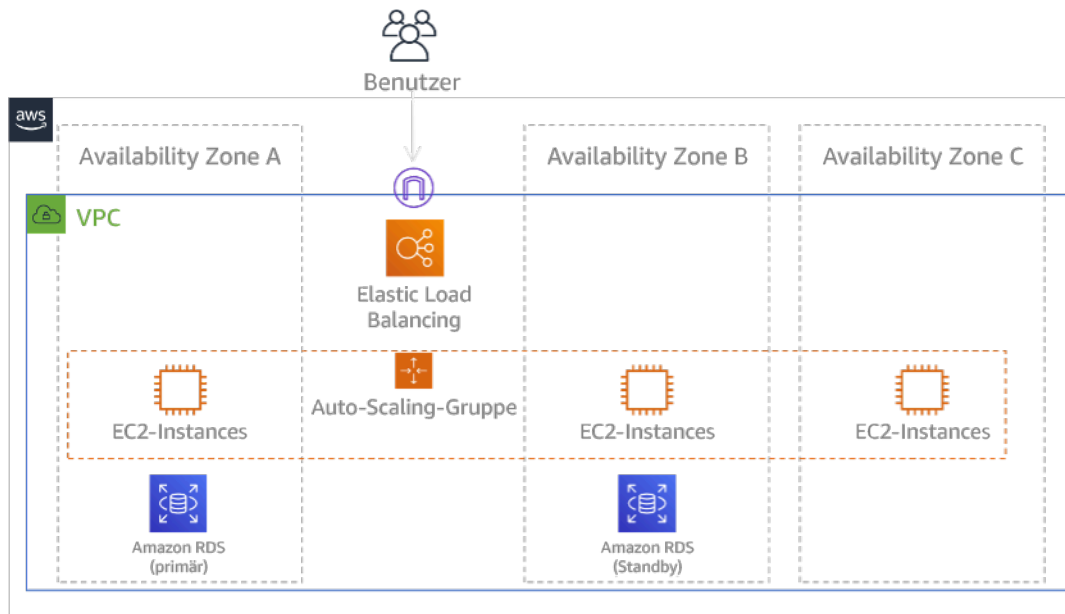


Abbildung 24: Multi-AZ-Architektur

Zusätzlich zu dieser HA-Architektur müssen Sie Backups aller Daten hinzufügen, die für die Ausführung Ihrer Workloads erforderlich sind. Dies ist besonders wichtig für Daten, die auf eine einzelne Zone beschränkt sind, etwa [Amazon-EBS-Volumes](#) oder [Amazon-Redshift-Cluster](#). Wenn eine AZ ausfällt, müssen Sie diese Daten in einer anderen AZ wiederherstellen. Wenn möglich, sollten Sie auch Datensicherungen auf einen anderen AWS-Region kopieren, um eine zusätzliche Sicherheit zu gewährleisten.

Ein weniger verbreiteter alternativer Ansatz für einzelne Regionen, die Multi-AZ-Notfallwiederherstellung, wird im Blogbeitrag [Entwickeln hoch resilienter Anwendungen mit Amazon Application Recovery Controller, Teil 1: Stack für eine einzelne Region](#) beschrieben. Hier besteht die Strategie darin, so viel Isolation wie möglich zwischen den AZs aufrechtzuerhalten, ähnlich wie bei den Regionen. Bei dieser alternativen Strategie können Sie sich für einen Aktiv/ Aktiv- oder Aktiv/Passiv-Ansatz entscheiden.

Note

Für einige Workloads gibt es gesetzliche Vorschriften im Hinblick auf die Datenresidenz. Wenn dies auf Ihre Workload in einer Region zutrifft, in der es derzeit nur eine AWS-

Region gibt, dann ist die Multi-Region für Ihre geschäftlichen Anforderungen nicht geeignet. Multi-AZ-Strategien bieten einen guten Schutz gegen die meisten Notfälle.

3. Bewerten Sie vor dem Failover (während des normalen Betriebs) die Ressourcen Ihrer Workloads und deren Konfiguration in der Wiederherstellungsregion.

Verwenden Sie für Infrastruktur- und AWS-Ressourcen Infrastruktur als Code, etwa [AWS CloudFormation](#) oder Tools von Drittanbietern wie Hashicorp Terraform. Um die Bereitstellung über mehrere Konten und Regionen hinweg mit einer einzigen Operation durchzuführen, können Sie [AWS CloudFormation StackSets](#) verwenden. Bei Multi-Site-Aktiv/Aktiv- und Hot-Standby-Strategien verfügt die in Ihrer Wiederherstellungsregion bereitgestellte Infrastruktur über dieselben Ressourcen wie Ihre Primärregion. Bei den Strategien Pilot Light und Warm Standby sind zusätzliche Maßnahmen erforderlich, um die Infrastruktur produktionsreif zu machen. Mithilfe von CloudFormation-[Parametern](#) und [bedingter Logik](#) können Sie mit einer [einzigen Vorlage](#) steuern, ob ein bereitgestellter Stack aktiv oder im Standby-Modus ist. Wenn Sie Elastic Disaster Recovery verwenden, repliziert und orchestriert der Service die Wiederherstellung von Anwendungskonfigurationen und Datenverarbeitungs-Ressourcen.

Alle DR-Strategien erfordern, dass Datenquellen innerhalb der AWS-Region gesichert werden und diese Backups anschließend in die Wiederherstellungsregion kopiert werden. [AWS Backup](#) bietet eine zentrale Ansicht, in der Sie Backups für diese Ressourcen konfigurieren, planen und überwachen können. Für Pilot Light, Warm Standby und Multi-Site Aktiv/Aktiv sollten Sie auch Daten aus der primären Region auf Datenressourcen in der Wiederherstellungsregion replizieren, etwa [Amazon Relational Database Service \(Amazon RDS\)](#)-DB-Instances oder [Amazon DynamoDB](#)-Tabellen. Diese Datenressourcen sind daher aktiv und bereit, Anfragen in der Wiederherstellungsregion zu bedienen.

Weitere Informationen darüber, wie AWS-Services in verschiedenen Regionen funktionieren, finden Sie in der Blogserie [Erstellen einer regionsübergreifenden Anwendung mit AWS](#).

4. Ermitteln und implementieren Sie, wie Sie Ihre Wiederherstellungsregion bei Bedarf (im Notfall) auf ein Failover vorbereiten.

Bei Multi-Site Aktiv/Aktiv bedeutet Failover, dass eine Region evakuiert wird und die verbleibenden aktiven Regionen genutzt werden. Im Allgemeinen sind diese Regionen bereit, Datenverkehr aufzunehmen. Bei den Strategien Pilot Light und Warm Standby müssen Ihre Wiederherstellungsmaßnahmen die fehlenden Ressourcen bereitstellen, z. B. die EC2-Instances in Abbildung 20, sowie alle anderen fehlenden Ressourcen.

Bei allen oben genannten Strategien müssen Sie möglicherweise schreibgeschützte Instances von Datenbanken zur primären Lese-/Schreib-Instance machen.

Bei der Sicherung und Wiederherstellung werden durch die Wiederherstellung von Daten aus der Sicherung Ressourcen für diese Daten wie EBS-Volumes, RDS-DB-Instances und DynamoDB-Tabellen erstellt. Außerdem müssen Sie die Infrastruktur wiederherstellen und Code bereitstellen. Sie können AWS Backup nutzen, um Daten in der Wiederherstellungsregion wiederherzustellen. Weitere Details finden Sie unter [REL09-BP01 Ermitteln und Sichern aller zu sichernden Daten oder Reproduzieren der Daten aus Quellen](#). Der Wiederaufbau der Infrastruktur umfasst die Erstellung von Ressourcen wie EC2-Instances zusätzlich zur [Amazon Virtual Private Cloud \(Amazon VPC\)](#), zu Subnetzen und zu benötigten Sicherheitsgruppen. Sie können einen Großteil des Wiederherstellungsprozesses automatisieren. Wie das geht, erfahren Sie in [diesem Blogbeitrag](#).

5. Ermitteln und implementieren Sie, wie Sie den Datenverkehr bei Bedarf (im Notfall) zum Failover umleiten.

Dieser Failover-Vorgang kann entweder automatisch oder manuell eingeleitet werden. Ein automatisch eingeleiteter Failover auf der Grundlage von Zustandsprüfungen oder Alarmen ist mit Vorsicht zu genießen, da ein unnötiger Failover (Fehlalarm) Kosten wie Nichtverfügbarkeit und Datenverlust verursacht. Daher wird häufig ein manuell initiiertes Failover verwendet. In diesem Fall sollten Sie die Schritte für den Failover dennoch automatisieren, sodass die manuelle Auslösung wie ein Knopfdruck wirkt.

Bei der Inanspruchnahme von AWS-Services gibt es mehrere Optionen für die Verwaltung des Datenverkehrs zu berücksichtigen. Eine Option ist die Verwendung von [Amazon Route 53](#). Mit Amazon Route 53 können Sie mehrere IP-Endpunkte in einem oder mehreren AWS-Regionen mit einem Route-53-Domainnamen verknüpfen. Um ein manuell eingeleitetes Failover zu implementieren, können Sie [Amazon Application Recovery Controller](#) verwenden, das eine hochverfügbare Datenebenen-API zur Umleitung des Datenverkehrs in die Wiederherstellungsregion bereitstellt. Verwenden Sie bei der Implementierung von Failover Vorgänge auf der Datenebene und vermeiden Sie solche auf der Steuerebene, wie in [REL11-BP04 Nutzen der Datenebene und nicht der Steuerebene während der Wiederherstellung](#) beschrieben.

Weitere Informationen zu diesen und anderen Optionen finden Sie in [diesem Abschnitt des Whitepapers zur Notfallwiederherstellung](#).

6. Entwerfen Sie einen Plan für das Failback Ihrer Workload.

Failback bedeutet, dass Sie den Workload-Betrieb in der primären Region wieder aufnehmen, nachdem ein Notfallereignis abgeklungen ist. Die Bereitstellung von Infrastruktur und Code für die primäre Region erfolgt im Allgemeinen in denselben Schritten wie ursprünglich, wobei Infrastruktur als Code und Code-Bereitstellungspipelines verwendet werden. Die Herausforderung beim Failback ist die Wiederherstellung von Datenspeichern und die Sicherstellung ihrer Konsistenz mit der in Betrieb befindlichen Wiederherstellungsregion.

Im ausgefallenen Zustand sind die Datenbanken in der Wiederherstellungsregion aktiv und verfügen über die aktuellen Daten. Ziel ist es dann, eine erneute Synchronisierung von der Wiederherstellungsregion mit der primären Region vorzunehmen, um sicherzustellen, dass diese auf dem neuesten Stand ist.

Einige AWS-Services werden das automatisch tun. Wenn Sie [globale Amazon DynamoDB-Tabellen](#) verwenden, setzt DynamoDB die Propagierung aller ausstehenden Schreibvorgänge fort, sobald die Tabelle wieder online ist, auch wenn sie in der primären Region nicht mehr verfügbar war. Wenn Sie [Amazon Aurora Global Database](#) und ein [verwaltetes geplantes Failover](#) verwenden, wird die vorhandene Replikationstopologie der globalen Aurora-Datenbank beibehalten. Daher wird die ehemalige Lese-/Schreib-Instance in der primären Region zu einem Replikat und erhält Aktualisierungen von der Wiederherstellungsregion.

In Fällen, in denen dies nicht automatisch geschieht, müssen Sie die Datenbank in der primären Region als Replikat der Datenbank in der Wiederherstellungsregion neu einrichten. In vielen Fällen bedeutet dies, dass die alte primäre Datenbank gelöscht und neue Replikate erstellt werden müssen.

Wenn Sie nach einem Failover in Ihrer Wiederherstellungsregion weiterarbeiten können, sollten Sie diese zur neuen Primärregion machen. Sie würden trotzdem alle oben genannten Schritte durchführen, um die ehemalige Primärregion in eine Wiederherstellungsregion zu verwandeln. Einige Unternehmen führen eine planmäßige Rotation durch und tauschen ihre Primär- und Wiederherstellungsregionen in regelmäßigen Abständen aus (z. B. alle drei Monate).

Alle für Failover und Failback erforderlichen Schritte sollten in einem Playbook festgehalten werden, das allen Teammitgliedern zur Verfügung steht und regelmäßig überprüft wird.

Wenn Sie Elastic Disaster Recovery verwenden, hilft der Service bei der Orchestrierung und Automatisierung des Failback-Prozesses. Weitere Informationen finden Sie unter [Durchführen eines Failbacks](#).

Aufwand für den Implementierungsplan: Hoch

Ressourcen

Zugehörige bewährte Methoden:

- [the section called “REL09-BP01 Ermitteln und Sichern aller zu sichernden Daten oder Reproduzieren der Daten aus Quellen”](#)
- [the section called “REL11-BP04 Nutzen der Datenebene und nicht der Steuerebene während der Wiederherstellung”](#)
- [the section called “REL13-BP01 Definieren von Wiederherstellungszielen bei Ausfällen und Datenverlusten”](#)

Zugehörige Dokumente:

- [AWS Architecture Blog: Notfallwiederherstellung](#)
- [Notfallwiederherstellung von Workloads in AWS: Wiederherstellung in der Cloud \(AWS-Whitepaper\)](#)
- [Optionen für die Notfallwiederherstellung in der Cloud](#)
- [Entwickeln einer Multi-Region-Serverless-Backend-Lösung, die aktiv/aktiv ist](#)
- [Multi-Region-Serverless-Backend – neu aufgelegt](#)
- [RDS: Regionsübergreifendes Replizieren von Lesereplikaten](#)
- [Route 53: Konfiguration des DNS-Failovers](#)
- [S3: Regionsübergreifende Replikation](#)
- [Was ist AWS Backup?](#)
- [Was ist Amazon Application Recovery Controller?](#)
- [AWS Elastic Disaster Recovery](#)
- [HashiCorp Terraform: Erste Schritte – AWS](#)
- [APN-Partner: Partner, die Sie bei der Notfallwiederherstellung unterstützen können](#)
- [AWS Marketplace: Für die Notfallwiederherstellung geeignete Produkte](#)

Zugehörige Videos:

- [Notfallwiederherstellung von Workloads in AWS](#)

- [AWS re:Invent 2018: Architecture Patterns for Multi-Region Active-Active Applications \(ARC209-R2\)](#)
- [Erste Schritte mit AWS Elastic Disaster Recovery | Amazon Web Services](#)

REL13-BP03 Testen der Implementierung der Notfallwiederherstellung zur Validierung

Testen Sie regelmäßig den Failover zu Ihrem Wiederherstellungsstandort, um zu überprüfen, ob er ordnungsgemäß funktioniert und ob das RTO und RPO eingehalten werden.

Typische Anti-Muster:

- Failover sollten nie in der Produktion getestet werden.

Vorteile der Nutzung dieser bewährten Methode: Durch regelmäßige Tests des Notfallwiederherstellungsplans ist gewährleistet, dass er bei Bedarf funktioniert und von Ihrem Team umgesetzt werden kann.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Vom Erstellen selten durchgeführter Wiederherstellungspfade wird abgeraten. So könnten Sie beispielsweise einen zweiten Datenspeicher unterhalten, der nur für Leseabfragen verwendet wird. Wenn Sie Daten in einen Datenspeicher schreiben und der primäre Datenspeicher einen Fehler ausgibt, können Sie einen Failover auf den zweiten Datenspeicher durchführen. Wenn Sie diesen Failover nicht regelmäßig testen, werden Sie möglicherweise feststellen, dass Ihre Annahmen zu den Möglichkeiten des sekundären Datenspeichers unzutreffend sind. Die Kapazität des sekundären Datenspeichers, die bei den letzten Tests möglicherweise noch ausreichend war, genügt möglicherweise nicht mehr den Anforderungen dieses Szenarios. Unsere Erfahrungen haben gezeigt, dass bei einer Wiederherstellung nach einem Fehler nur der Pfad funktioniert, den Sie regelmäßig testen. Daher ist es ratsam, mehrere Wiederherstellungspfade zu pflegen. Sie können Wiederherstellungsmuster erstellen und diese regelmäßig testen. Auch komplexe oder kritische Wiederherstellungspfade müssen regelmäßig mittels Fehlersimulationen in der Produktion durchgeführt werden, um sicherzustellen, dass sie funktionieren. In dem gerade besprochenen Beispiel sollten Sie regelmäßig und unabhängig von der Erfordernis einen Failover auf die Standby-Ressourcen durchführen.

Implementierungsschritte

1. Legen Sie die Workloads für die Wiederherstellung aus. Testen Sie regelmäßig Ihre Wiederherstellungspfade. Die Recovery-orientierte Datenverarbeitung identifiziert die Merkmale von Systemen, die die Wiederherstellung verbessern: Isolierung und Redundanz, systemweite Fähigkeit zur Rücknahme von Änderungen, Fähigkeit zur Überwachung und Bestimmung des Zustands, Fähigkeit zur Diagnose, automatisierte Wiederherstellung, modularer Aufbau und Fähigkeit zum Neustart. Testen Sie den Wiederherstellungspfad, um zu überprüfen, ob Sie die Wiederherstellung in der angegebenen Zeit und in dem angegebenen Zustand durchführen können. Dokumentieren Sie während dieser Wiederherstellung auftretende Probleme in Ihren Runbooks und suchen Sie vor dem nächsten Test nach Lösungen.
2. Verwenden Sie für Amazon EC2-basierte Workloads [AWS Elastic Disaster Recovery](#), um Drill-Instances für Ihre DR-Strategie zu implementieren und zu starten. AWS Elastic Disaster Recovery bietet die Möglichkeit, Drills effizient durchzuführen, sodass Sie sich auf ein Failover-Ereignis vorbereiten können. Sie können Ihre Instances mit Elastic Disaster Recovery außerdem regelmäßig zu Test- und Übungszwecken starten, ohne den Datenverkehr weiterleiten zu müssen.

Ressourcen

Zugehörige Dokumente:

- [APN-Partner: Partner, die Sie bei der Notfallwiederherstellung unterstützen können](#)
- [AWS Architecture Blog: Notfallwiederherstellung](#)
- [AWS Marketplace: Für die Notfallwiederherstellung geeignete Produkte](#)
- [AWS Elastic Disaster Recovery](#)
- [Notfallwiederherstellung von Workloads in AWS: Wiederherstellung in der Cloud \(AWS-Whitepaper\)](#)
- [AWS Elastic Disaster Recovery – Vorbereitung auf das Failover](#)
- [The Berkeley/Stanford Recovery-Oriented Computing Project](#)
- [Was ist AWS Fault Injection Simulator?](#)

Zugehörige Videos:

- [AWS re:Invent 2018: Architecture Patterns for Multi-Region Active-Active Applications](#)
- [AWS re:Invent 2019: Backup-and-Restore and Disaster-Recovery Solutions with AWS](#)

REL13-BP04 Verwalten der Konfigurationsabweichungen am Standort oder in der Region der Notfallwiederherstellung

Um ein erfolgreiches Verfahren für die Notfallwiederherstellung durchführen zu können, müssen Ihre Workloads in der Lage sein, den normalen Betrieb zeitnah wieder aufzunehmen, ohne dass es zu nennenswerten Funktions- oder Datenverlusten kommt, sobald die Umgebung für die Notfallwiederherstellung online gegangen ist. Um dieses Ziel zu erreichen, ist es unerlässlich, eine konsistente Infrastruktur sowie konsistente Daten und Konfigurationen zwischen Ihrer Umgebung für die Notfallwiederherstellung und der primären Umgebung aufrechtzuerhalten.

Gewünschtes Ergebnis: Die Konfiguration und die Daten Ihres Standorts für die Notfallwiederherstellung entsprechen denen des primären Standorts, wodurch bei Bedarf eine schnelle und vollständige Wiederherstellung möglich ist.

Typische Anti-Muster:

- Sie versäumen es, die Wiederherstellungsstandorte zu aktualisieren, wenn Änderungen an den primären Standorten vorgenommen werden. Dadurch kommt es zu veralteten Konfigurationen, die die Wiederherstellungsbemühungen behindern könnten.
- Sie lassen mögliche Einschränkungen wie Serviceunterschiede zwischen primärem Standort und Wiederherstellungsstandort, die zu unerwarteten Fehlern beim Failover führen können, außer Acht.
- Sie verlassen sich bei der Aktualisierung und Synchronisierung der Umgebung für die Notfallwiederherstellung auf manuelle Prozesse, was das Risiko von menschlichen Fehlern und Inkonsistenzen erhöht.
- Sie erkennen Konfigurationsabweichungen nicht und haben daher vor einem Vorfall fälschlicherweise das Gefühl, dass der Standort für die Notfallwiederherstellung bereit ist.

Vorteile der Einführung dieser bewährten Methode: Durch die Konsistenz zwischen der Umgebung für die Notfallwiederherstellung und der primären Umgebung erhöht sich die Wahrscheinlichkeit einer erfolgreichen Wiederherstellung nach einem Vorfall erheblich und das Risiko eines fehlgeschlagenen Wiederherstellungsvorgangs verringert sich.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Ein umfassender Ansatz für das Konfigurationsmanagement und die Failover-Bereitschaft kann Ihnen dabei helfen, sicherzustellen, dass der Standort für die Notfallwiederherstellung regelmäßig aktualisiert wird und bei einem Ausfall des primären Standorts zur Übernahme bereit ist.

Um Konsistenz zwischen Ihrer primären Umgebung und Ihrer Umgebung für die Notfallwiederherstellung zu erreichen, sollten Sie sicherstellen, dass Ihre Bereitstellungs-Pipelines Anwendungen sowohl an Ihren primären Standort als auch an Ihren Standort für die Notfallwiederherstellung verteilen. Führen Sie Änderungen an den Standorten für die Notfallwiederherstellung nach einem angemessenen Evaluierungszeitraum durch (sogenannte gestaffelte Bereitstellungen), um Probleme am primären Standort zu erkennen und die Bereitstellung zu stoppen, bevor sie sich ausbreiten. Implementieren Sie eine Überwachung, um Konfigurationsabweichungen zu erkennen und Änderungen sowie die Einhaltung von Vorschriften in Ihren Umgebungen nachzuverfolgen. Führen Sie automatisierte Problembehebungen am Standort für die Notfallwiederherstellung durch, um vollständige Konsistenz sicherzustellen und dafür zu sorgen, dass der Standort bei einem Vorfall übernahmebereit ist.

Implementierungsschritte

1. Stellen Sie sicher, dass die Region für die Notfallwiederherstellung die AWS-Services und -Features enthält, die für eine erfolgreiche Ausführung Ihres Plans für die Notfallwiederherstellung erforderlich sind.
2. Verwenden Sie Infrastructure as Code (IaC). Sorgen Sie dafür, dass Ihre Konfigurationsvorlagen für die Produktionsinfrastruktur und die Anwendungen korrekt sind, und wenden Sie die Vorlagen regelmäßig auf Ihre Umgebung für die Notfallwiederherstellung an. [AWS CloudFormation](#) kann Abweichungen zwischen den Vorgaben Ihrer CloudFormation-Vorlagen und der tatsächlichen Bereitstellung erkennen.
3. Konfigurieren Sie CI/CD-Pipelines, um Anwendungen und Infrastrukturupdates für alle Umgebungen bereitzustellen, einschließlich primärer Standorte und Standorte für die Notfallwiederherstellung. CI/CD-Lösungen wie [AWS CodePipeline](#) können den Bereitstellungsprozess automatisieren, wodurch das Risiko von Konfigurationsabweichungen verringert wird.
4. Staffeln Sie die Bereitstellungen zwischen der primären Umgebung und der Umgebung für die Notfallwiederherstellung. Auf diese Weise können Updates zunächst in der primären Umgebung bereitgestellt und getestet und so Probleme am primären Standort isoliert werden, bevor die Updates an den Standort für die Notfallwiederherstellung weitergegeben werden. Dieser Ansatz verhindert, dass Fehler gleichzeitig an die Produktion und den Standort für die Notfallwiederherstellung weitergegeben werden, und stellt die Integrität der Umgebung für die Notfallwiederherstellung sicher.
5. Überwachen Sie die Ressourcenkonfigurationen sowohl in der primären Umgebung als auch in der Umgebung für die Notfallwiederherstellung kontinuierlich. Lösungen wie [AWS Config](#) können

helfen, die Einhaltung von Konfigurationen durchzusetzen und Abweichungen zu erkennen. Auf diese Weise können konsistente Konfigurationen in allen Umgebungen aufrechterhalten werden.

6. Implementieren Sie Warnmechanismen, damit jegliche Konfigurationsabweichungen oder Unterbrechungen oder Verzögerungen der Datenreplikation nachverfolgt werden und eine entsprechende Benachrichtigung erfolgt.
7. Automatisieren Sie die Behebung erkannter Konfigurationsabweichungen.
8. Planen Sie regelmäßige Überwachungen und Konformitätsprüfungen ein, um die fortlaufende Abstimmung zwischen der primären Konfiguration und der Konfiguration für die Notfallwiederherstellung zu überprüfen. Regelmäßige Überprüfungen helfen Ihnen dabei, die Einhaltung definierter Regeln sicherzustellen und etwaige Unstimmigkeiten zu identifizieren, die behoben werden müssen.
9. Prüfen Sie, ob es Diskrepanzen in Bezug auf die von AWS bereitgestellte Kapazität, die Service Quotas, die Drosselungsgrenzen, Konfigurationen und Versionen gibt.

Ressourcen

Zugehörige bewährte Methoden:

- [REL01-BP01 Kenntnis von Service Quotas und Einschränkungen](#)
- [REL01-BP02 Verwalten von Service Quotas für mehrere Konten und Regionen](#)
- [REL01-BP04 Überwachen und Verwalten von Kontingenten](#)
- [REL13-BP03 Testen der Implementierung der Notfallwiederherstellung zur Validierung](#)

Zugehörige Dokumente:

- [Korrigieren von nicht konformen AWS-Ressourcen mit AWS-Config-Regeln](#)
- [AWS Systems Manager Automation](#)
- [AWS CloudFormation: Erkennen von nicht verwalteten Konfigurationsänderungen an Stacks und Ressourcen](#)
- [AWS CloudFormation: Ermitteln von Abweichungen im gesamten CloudFormation-Stack](#)
- [AWS Systems Manager Automation](#)
- [Notfallwiederherstellung von Workloads in AWS: Wiederherstellung in der Cloud \(AWS-Whitepaper\)](#)
- [Wie implementiere ich eine Lösung für die Verwaltung der Infrastrukturkonfiguration in AWS?](#)

- [Korrigieren von nicht konformen AWS-Ressourcen mit AWS-Config-Regeln](#)

Zugehörige Videos:

- [AWS re:Invent 2018: Architecture Patterns for Multi-Region Active-Active Applications \(ARC209-R2\)](#)

Zugehörige Beispiele:

- [CloudFormation Registry](#)
- [Quota Monitor for AWS](#)
- [Implementieren der automatischen Behebung von Abweichungen für AWS CloudFormation unter Verwendung von Amazon CloudWatch und AWS Lambda](#)
- [AWS Architecture Blog: Notfallwiederherstellung](#)
- [AWS Marketplace: Für die Notfallwiederherstellung geeignete Produkte](#)
- [Automatisierung sicherer, vollautomatischer Bereitstellungen](#)

REL13-BP05 Automatisieren der Wiederherstellung

Implementieren Sie getestete und automatisierte Wiederherstellungsmechanismen, die zuverlässig, beobachtbar und reproduzierbar sind, um das Risiko und die Auswirkungen von Ausfällen auf das Geschäft zu reduzieren.

Gewünschtes Ergebnis: Sie haben einen gut dokumentierten, standardisierten und gründlich getesteten Automatisierungs-Workflow für Wiederherstellungsprozesse implementiert. Ihre Wiederherstellungsautomatisierung behebt automatisch kleinere Probleme, bei denen das Risiko von Datenverlusten oder Nichtverfügbarkeit gering ist. Sie sind in der Lage, bei schwerwiegenden Vorfällen schnell Wiederherstellungsprozesse aufzurufen, das Problembehebungsverhalten während der Ausführung zu beobachten und die Prozesse zu beenden, wenn Sie gefährliche Situationen oder Fehler beobachten.

Typische Anti-Muster:

- Für Ihren Wiederherstellungsplan sind Sie auf Komponenten oder Mechanismen angewiesen, die ausgefallen oder beeinträchtigt sind.
- Ihre Wiederherstellungsprozesse erfordern manuelle Eingriffe, z. B. den Zugriff auf die Konsole (auch als ClickOps bezeichnet).

- Sie leiten automatisch Wiederherstellungsverfahren in Situationen ein, in denen ein hohes Risiko von Datenverlusten oder Nichtverfügbarkeit besteht.
- Sie beziehen keinen Mechanismus (wie z. B. ein Andon-Cord oder eine große rote Stopptaste) ein, mit dem Sie ein Wiederherstellungsverfahren abbrechen können, das nicht funktioniert oder zusätzliche Risiken birgt.

Vorteile der Nutzung dieser bewährten Methode:

- Höhere Zuverlässigkeit, Vorhersehbarkeit und Konsistenz der Wiederherstellungsvorgänge.
- Fähigkeit, strengere Ziele für die Wiederherstellung, einschließlich Recovery Time Objective (RTO) und Recovery Point Objective (RPO), zu erfüllen.
- Geringere Wahrscheinlichkeit, dass die Wiederherstellung während eines Vorfalls fehlschlägt.
- Geringeres Risiko von Fehlern im Zusammenhang mit manuellen Wiederherstellungsprozessen, die anfällig für menschliche Fehler sind.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Um die automatisierte Wiederherstellung zu implementieren, benötigen Sie einen umfassenden Ansatz, der AWS-Services und bewährte Methoden nutzt. Identifizieren Sie zunächst kritische Komponenten und potenzielle Fehlerquellen in Ihrem Workload. Entwickeln Sie automatisierte Prozesse, mit denen Sie Ihre Workloads und Daten nach Ausfällen ohne menschliches Eingreifen wiederherstellen können.

Entwickeln Sie Ihre Wiederherstellungsautomatisierung unter Berücksichtigung der Prinzipien von Infrastructure as Code (IaC). Auf diese Weise wird Ihre Wiederherstellungsumgebung mit der Quellumgebung konsistent und es ist eine Versionsverwaltung Ihrer Wiederherstellungsprozesse möglich. Um komplexe Wiederherstellungs-Workflows zu orchestrieren, sollten Sie Lösungen wie [AWS Systems Manager Automations](#) oder [AWS Step Functions](#) in Betracht ziehen.

Die Automatisierung von Wiederherstellungsprozessen bietet erhebliche Vorteile und kann Ihnen helfen, Ihr Recovery Time Objective (RTO) und Recovery Point Objective (RPO) leichter zu erreichen. Die Prozesse können jedoch auf unerwartete Situationen stoßen. Diese können zu einem Ausfall der Prozesse führen oder neue Risiken wie zusätzliche Ausfallzeiten und Datenverluste mit sich bringen. Um dieses Risiko zu minimieren, sollten Sie die Möglichkeit bieten, eine laufende

Wiederherstellungsautomatisierung schnell zu unterbrechen. Nach der Unterbrechung des Prozesses können Sie Nachforschungen anstellen und Korrekturmaßnahmen ergreifen.

Für unterstützte Workloads sollten Sie Lösungen wie AWS Elastic Disaster Recovery (AWSDRS) in Betracht ziehen, um ein automatisiertes Failover zu ermöglichen. AWS DRS repliziert Ihre Computer (einschließlich Betriebssystem, Systemstatuskonfiguration, Datenbanken, Anwendungen und Dateien) kontinuierlich in einen Staging-Bereich in Ihrem Ziel-AWS-Konto und in Ihrer bevorzugten Region. Bei einem Vorfall automatisiert AWS DRS die Konvertierung Ihrer replizierten Server in vollständig bereitgestellte Workloads in Ihrer Wiederherstellungsregion in AWS.

Die Wartung und Verbesserung der automatisierten Wiederherstellung ist ein fortlaufender Prozess. Testen und verfeinern Sie Ihre Wiederherstellungsverfahren kontinuierlich auf der Grundlage der gewonnenen Erkenntnisse und halten Sie sich über neue AWS-Services und -Features auf dem Laufenden, die Ihre Wiederherstellungsmöglichkeiten verbessern können.

Implementierungsschritte

1. Planen der automatisierten Wiederherstellung

- a. Führen Sie eine gründliche Überprüfung Ihrer Workload-Architektur, Ihrer Komponenten und Abhängigkeiten durch, um automatisierte Wiederherstellungsmechanismen zu identifizieren und zu planen. Unterteilen Sie die Abhängigkeiten Ihres Workloads in harte und weiche Abhängigkeiten. Harte Abhängigkeiten sind Abhängigkeiten, ohne die der Workload nicht funktionieren kann und für die kein Ersatz bereitgestellt werden kann. Weiche Abhängigkeiten sind Abhängigkeiten, die der Workload normalerweise nutzt, die aber durch temporäre Ersatzsysteme oder -prozesse ersetzt werden können oder die durch eine [Graceful Degradation](#) bewältigt werden können.
- b. Richten Sie Prozesse ein, um fehlende oder beschädigte Daten zu identifizieren und wiederherzustellen.
- c. Definieren Sie Schritte zur Bestätigung eines wiederhergestellten stabilen Zustands nach Abschluss der Wiederherstellungsmaßnahmen.
- d. Berücksichtigen Sie alle Maßnahmen, die erforderlich sind, um das wiederhergestellte System vollständig einsatzbereit zu machen, z. B. das Vorwärmen und das Auffüllen von Caches.
- e. Denken Sie an Probleme, die während des Wiederherstellungsprozesses auftreten könnten, und überlegen Sie, wie Sie diese erkennen und beheben können.
- f. Stellen Sie sich Szenarien vor, in denen kein Zugriff auf den primären Standort und die zugehörige Steuerungsebene möglich ist. Stellen Sie sicher, dass Wiederherstellungsaktionen unabhängig durchgeführt werden können, ohne auf den primären Standort angewiesen zu sein.

Ziehen Sie Lösungen wie [Amazon Application Recovery Controller \(ARC\)](#) in Betracht, um den Datenverkehr umzuleiten, ohne dass DNS-Einträge manuell mutiert werden müssen.

2. Entwicklung eines automatisierten Wiederherstellungsprozesses

- a. Implementieren Sie automatisierte Fehlererkennungs- und Failover-Mechanismen für eine Wiederherstellung ohne manuelle Eingriffe. Erstellen Sie Dashboards, beispielsweise mit [Amazon CloudWatch](#), um den Fortschritt und den Zustand automatisierter Wiederherstellungsverfahren zu melden. Schließen Sie Verfahren zur Validierung einer erfolgreichen Wiederherstellung ein. Stellen Sie einen Mechanismus bereit, um eine laufende Wiederherstellung abubrechen.
- b. Erstellen Sie [Playbooks](#) als Ausweichlösung für Fehler, die nicht automatisch behoben werden können, und berücksichtigen Sie Ihren [Plan für die Notfallwiederherstellung](#).
- c. Testen Sie die Wiederherstellungsprozesse, wie in [REL13-BP03](#) beschrieben.

3. Vorbereitung auf die Wiederherstellung

- a. Evaluieren Sie den Zustand Ihres Wiederherstellungsstandorts und stellen Sie wichtige Komponenten im Voraus bereit. Weitere Informationen finden Sie unter [REL13-BP04](#).
- b. Definieren Sie klare Rollen, Verantwortlichkeiten und Entscheidungsprozesse für Wiederherstellungsoperationen und beziehen Sie dabei die relevanten Stakeholder und Teams im gesamten Unternehmen ein.
- c. Definieren Sie die Bedingungen für die Einleitung Ihrer Wiederherstellungsprozesse.
- d. Erstellen Sie einen Plan, um den Wiederherstellungsprozess rückgängig zu machen und bei Bedarf, oder nachdem dies als sicher erachtet wird, auf Ihren primären Standort zurückzugreifen.

Ressourcen

Zugehörige bewährte Methoden:

- [REL07-BP01 Automatisiertes Abrufen oder Skalieren von Ressourcen](#)
- [REL11-BP01 Überwachen aller Komponenten der Workload auf Fehler](#)
- [REL13-BP02 Verwenden von definierten Wiederherstellungsstrategien, um die Wiederherstellungsziele zu erreichen](#)
- [REL13-BP03 Testen der Implementierung der Notfallwiederherstellung zur Validierung](#)
- [REL13-BP04 Verwalten der Konfigurationsabweichungen am Standort oder in der Region der Notfallwiederherstellung](#)

Zugehörige Dokumente:

- [AWS Architecture Blog: Notfallwiederherstellung](#)
- [Notfallwiederherstellung von Workloads in AWS: Wiederherstellung in der Cloud \(AWS-Whitepaper\)](#)
- [Orchestrieren der Automatisierung der Notfallwiederherstellung mit Amazon Route 53 ARC und AWS Step Functions](#)
- [Erstellen von AWS-Systems-Manager-Automation-Runbooks mithilfe von AWS CDK](#)
- [AWS Marketplace: Für die Notfallwiederherstellung geeignete Produkte](#)
- [AWS Systems Manager Automation](#)
- [AWS Elastic Disaster Recovery](#)
- [Verwenden von Elastic Disaster Recovery für Failover und Failback](#)
- [AWS Elastic Disaster Recovery – Ressourcen](#)
- [APN-Partner: Partner, die Sie bei der Notfallwiederherstellung unterstützen können](#)

Zugehörige Videos:

- [AWS re:Invent 2018: Architecture Patterns for Multi-Region Active-Active Applications \(ARC209-R2\)](#)
- [AWS re:Invent 2022: AWS On Air ft. AWS Failback for AWS Elastic Disaster Recovery](#)

Leistungseffizienz

Die Säule der Leistungseffizienz betrifft die Fähigkeit zur effizienten Nutzung von Cloud-Ressourcen, um die Leistungsanforderungen zu erfüllen, sowie die Möglichkeit zur Aufrechterhaltung dieser Effizienz bei Nachfrageänderungen und einer Weiterentwicklung der Technologien. Verbindliche Anleitungen zur Implementierung finden Sie im [Whitepaper „Säule der Leistungseffizienz“](#).

Bereiche für bewährte Methoden

- [Auswahl der Architektur](#)
- [Computer und Hardware](#)
- [Datenverwaltung](#)
- [Netzwerk und Bereitstellung von Inhalten](#)
- [Prozess und Kultur](#)

Auswahl der Architektur

Fragen

- [LEIST 1. Wie wählen Sie geeignete Cloud-Ressourcen und -Architekturen für Ihren Workload aus?](#)

LEIST 1. Wie wählen Sie geeignete Cloud-Ressourcen und -Architekturen für Ihren Workload aus?

Die optimale Lösung für eine bestimmte Workload variiert und Lösungen sind häufig eine Kombination mehrerer Ansätze. Well-Architected-Workloads nutzen mehrere Lösungen und ermöglichen verschiedene Funktionen zur Verbesserung der Leistung.

Best Practices

- [PERF01-BP01 Informieren über verfügbare Cloud-Services und -Features](#)
- [PERF01-BP02 Einholen von Rat beim Cloud-Anbieter oder einem geeigneten Partner, um mehr über Architekturmuster und bewährte Methoden zu erfahren](#)
- [PERF01-BP03 Kosten bei Architekturentscheidungen berücksichtigen](#)
- [PERF01-BP04 Evaluieren, wie sich Kompromisse auf Kunden und Architektureffizienz auswirken](#)
- [PERF01-BP05 Benutzungsrichtlinien und Referenzarchitekturen](#)
- [PERF01-BP06 Verwenden von Benchmarking, um architektonische Entscheidungen zu treffen](#)
- [PERF01-BP07 Verwenden eines datengesteuerten Ansatzes für architektonische Entscheidungen](#)

PERF01-BP01 Informieren über verfügbare Cloud-Services und -Features

Informieren Sie sich kontinuierlich über verfügbare Services und Konfigurationen, die Ihnen helfen, bessere architektonische Entscheidungen zu treffen und die Leistungseffizienz Ihrer Workload-Architektur zu verbessern.

Typische Anti-Muster:

- Sie verwenden die Cloud als gemeinsam genutztes Rechenzentrum.
- Sie modernisieren die Anwendung nach der Migration in die Cloud nicht.
- Sie verwenden nur einen Speichertyp für alle Objekte, die gespeichert werden müssen.
- Sie verwenden Instance-Typen, die am besten zu Ihren aktuellen Standards passen, bei Bedarf jedoch größer sind.

- Von Ihnen werden Technologien bereitgestellt und verwaltet, die als verwaltete Services verfügbar sind.

Vorteile der Nutzung dieser bewährten Methode: Wenn Sie neue Services und Konfigurationen in Betracht ziehen, können Sie möglicherweise die Leistung erheblich verbessern, die Kosten senken und den Aufwand für die Aufrechterhaltung der Workload optimieren. Es kann Ihnen auch dabei helfen, die Wertschöpfung für Cloud-fähige Produkte zu beschleunigen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

AWS veröffentlicht kontinuierlich neue Services und Features, mit denen die Leistung verbessert und die Kosten von Cloud-Workloads gesenkt werden können. Es ist entscheidend, mit diesen neuen Services und Features auf dem Laufenden zu bleiben, um die Leistungseffizienz in der Cloud aufrechtzuerhalten. Die Modernisierung der Workload-Architektur hilft Ihnen auch dabei, die Produktivität zu beschleunigen, Innovationen voranzutreiben und mehr Wachstumschancen zu erschließen.

Implementierungsschritte

- Inventarisieren Sie die Workload-Software und -Architektur für verwandte Services. Entscheiden Sie, über welche Produktkategorie Sie mehr erfahren möchten.
- Erkunden Sie die AWS-Angebote, um die relevanten Services und Konfigurationsoptionen zu identifizieren und kennenzulernen, mit denen Sie die Leistung verbessern und die Kosten und die betriebliche Komplexität reduzieren können.
 - [Amazon Web Services Cloud](#)
 - [AWS Academy](#)
 - [Neuerungen bei AWS](#)
 - [AWS-Blog](#)
 - [AWS Skill Builder](#)
 - [AWS-Veranstaltungen und -Webinare](#)
 - [AWS Training und Zertifizierungen](#)
 - [AWS-YouTube-Kanal](#)
 - [AWS-Workshops](#)
 - [AWS-Communitys](#)

- Verwenden Sie [Amazon Q](#), um relevante Informationen und Tipps zu Services zu erhalten.
- Verwenden Sie Sandbox- bzw. Nicht-Produktionsumgebungen, um neue Services zu erlernen und mit ihnen zu experimentieren, ohne dass zusätzliche Kosten anfallen.
- Informieren Sie sich kontinuierlich über neue Cloud-Services und -Features.

Ressourcen

Zugehörige Dokumente:

- [Übersicht über Amazon Web Services](#)
- [Amazon EC2 features](#)
- [Lernen Sie Schritt für Schritt mit einem Lehrplan für AWS-Partner](#)
- [AWS Training and Certification](#)
- [My learning path to become an AWS solutions architect](#)
- [AWS-Architekturzentrum](#)
- [AWS Partner Network](#)
- [AWS Lösungsbibliothek](#)
- [AWS Knowledge Center](#)
- [Moderne Anwendungen in AWS entwickeln](#)

Zugehörige Videos:

- [AWS re:Invent 2023 – What's new with Amazon EC2](#)
- [AWS re:Invent 2022 – Reduce your operational and infrastructure costs with Amazon ECS](#)
- [AWS re:Invent 2023 – Mit AWS die Effizienz, Agilität und Innovation der Cloud nutzen](#)
- [AWS re:Invent 2022 – ML-Modelle für Inferenz mit hoher Leistung und niedrigen Kosten bereitstellen](#)
- [This is my Architecture](#)

Zugehörige Beispiele:

- [AWS Beispiele](#)
- [AWS-SDK-Beispiele](#)

PERF01-BP02 Einholen von Rat beim Cloud-Anbieter oder einem geeigneten Partner, um mehr über Architekturmuster und bewährte Methoden zu erfahren

Greifen Sie bei Ihren architektonischen Entscheidungen auf die Ressourcen von Cloud-Unternehmen, wie etwa Dokumentation, Lösungsarchitekten, professionelle Services oder einen geeigneten Partner zurück. Diese Ressourcen helfen Ihnen dabei, Ihre Architektur zu überprüfen und zu verbessern, um so die Leistung zu optimieren.

Typische Anti-Muster:

- Sie verwenden AWS als gängigen Cloud-Anbieter.
- Sie verwenden AWS-Services auf eine Weise, für die sie nicht konzipiert wurden.
- Sie befolgen alle Anweisungen, ohne Ihren Geschäftskontext zu berücksichtigen.

Vorteile der Nutzung dieser bewährten Methode: Wenn Sie sich Rat bei einem Cloud-Anbieter oder einem geeigneten Partner einholen, können Sie die richtige Architektur für die Workload wählen und Entscheidungen mit größerer Zuversicht treffen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

AWS bietet eine breite Palette an Anleitungen, Dokumentation und Ressourcen, die Sie bei der Entwicklung und Verwaltung effizienter Cloud-Workloads unterstützen können. Die AWS-Dokumentation enthält Codebeispiele, Tutorials und detaillierte Serviceerklärungen. Zusätzlich zur Dokumentation bietet AWS Trainings- und Zertifizierungsprogramme, Lösungsarchitekten und professionelle Services, die Kunden dabei helfen können, verschiedene Aspekte von Cloud-Services zu entdecken und eine effiziente Cloud-Architektur in AWS zu implementieren.

Nutzen Sie diese Ressourcen, um Einblicke in wertvolles Wissen und bewährte Methoden zu gewinnen, Zeit zu sparen und bessere Ergebnisse in der AWS Cloud zu erzielen.

Implementierungsschritte

- Lesen Sie die AWS-Dokumentation und -Anleitungen und befolgen Sie die bewährten Methoden. Diese Ressourcen können Ihnen helfen, Services effektiv auszuwählen und zu konfigurieren und eine bessere Leistung zu erzielen.
- [AWS-Dokumentation](#) (wie Benutzerhandbücher und Whitepapers)

- [AWSBlog](#)
- [AWS Training und Zertifizierungen](#)
- [AWS-YouTube-Kanal](#)
- Nehmen Sie an AWS-Partnerveranstaltungen (wie AWS Global Summits, AWS re:Invent, Benutzergruppen und Workshops) teil, um von AWS-Experten mehr über bewährte Methoden für die Nutzung von AWS-Services zu lernen.
 - [Lernen Sie Schritt für Schritt mit einem Lehrplan für AWS-Partner](#)
 - [AWS-Veranstaltungen und -Webinare](#)
 - [AWS-Workshops](#)
 - [AWS-Communitys](#)
- Wenden Sie sich an AWS, wenn Sie zusätzliche Anleitungen oder Produktinformationen benötigen. AWS Solutions Architects und [AWS Professional Services](#) liefern Ratschläge für die Implementierung von Lösungen. [AWS Partner](#) bieten AWS-Fachwissen, damit Sie in Ihrem Unternehmen flexibel agieren und Innovationen nutzen können.
- Verwenden Sie [Support](#), wenn Sie technischen Support benötigen, um einen Service effektiv nutzen zu können. [Unsere Support-Pläne](#) bieten Ihnen die richtige Kombination aus Tools und Zugang zu Fachwissen, um die Grundlagen für Ihren Erfolg mit AWS zu legen, ohne dabei Themen wie Leistungsoptimierung, Risikomanagement und Kostenkontrolle zu vernachlässigen.

Ressourcen

Zugehörige Dokumente:

- [AWS-Architekturzentrum](#)
- [AWS Partner Network](#)
- [AWS-Lösungsbibliothek](#)
- [AWS Knowledge Center](#)
- [AWS Enterprise Support](#)

Zugehörige Videos:

- [This is my Architecture](#)
- [AWS re:Invent 2.023 - Advanced event-driven patterns with Amazon EventBridge](#)
- [AWS re:Invent 2023 – Implementierung verteilter Designmuster in AWS](#)

- [AWS re:Invent 2023 – Anwendungsarchitektur als Code](#)

Zugehörige Beispiele:

- [AWS Beispiele](#)
- [AWS-SDK-Beispiele](#)
- [AWS-Analytics-Referenzarchitektur](#)

PERF01-BP03 Kosten bei Architekturentscheidungen berücksichtigen

Berücksichtigen Sie die Kosten bei Ihren architektonischen Entscheidungen, um die Ressourcennutzung und Leistungseffizienz der Cloud-Workloads zu verbessern. Wenn Sie sich der Kostenauswirkungen der Cloud-Workload bewusst sind, ist es wahrscheinlicher, dass Sie effiziente Ressourcen nutzen und verschwenderische Methoden reduzieren.

Typische Anti-Muster:

- Sie verwenden nur eine Instance-Familie.
- Sie bewerten keine lizenzierten Lösungen verglichen mit Open-Source-Lösungen.
- Sie definieren keine Speicher-Lebenszyklusrichtlinien.
- Sie überprüfen keine neuen Dienste und Funktionen von AWS Cloud
- Sie nutzen nur Blockspeicher.

Vorteile der Nutzung dieser bewährten Methode: Wenn Sie die Kosten bei Ihrer Entscheidungsfindung berücksichtigen, können Sie effizientere Ressourcen einsetzen und andere Investitionen in Betracht ziehen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Die Kostenoptimierung von Workloads kann die Ressourcennutzung verbessern und Verschwendung bei einer Cloud-Workload vermeiden. Die Berücksichtigung der Kosten bei architektonischen Entscheidungen beinhaltet in der Regel die richtige Dimensionierung der Workload-Komponenten und die Schaffung von Elastizität. Dies führt zu einer verbesserten Leistungseffizienz von Cloud-Workloads.

Implementierungsschritte

- Legen Sie Kostenziele wie Budgetlimits für die Cloud-Workload fest.
- Identifizieren Sie die wesentlichen Komponenten (wie Instances und Speicher), die die Kosten der Workload erhöhen. Sie können [AWS Pricing Calculator](#) und [AWS Cost Explorer](#) verwenden, um die wichtigsten Kostentreiber in Ihrer Workload zu identifizieren.
- Informieren Sie sich über die [Preismodelle](#) in der Cloud, z. B. On-Demand, Reserved Instances, Savings Plans und Spot Instances.
- Verwenden Sie die [bewährten Methoden zur Kostenoptimierung von Well-Architected](#), um diese Schlüsselkomponenten im Hinblick auf die Kosten zu optimieren.
- Überwachen und analysieren Sie kontinuierlich die Kosten, um Möglichkeiten zur Kostenoptimierung in der Workload zu identifizieren.
 - Nutzen Sie [AWS -Budgets](#), um bei nicht akzeptablen Kosten Warnungsmeldungen zu erhalten.
 - Verwenden Sie [AWS Compute Optimizer](#) oder [AWS Trusted Advisor](#), um Empfehlungen zur Kostenoptimierung zu erhalten.
 - Nutzen Sie [AWS Cost Anomaly Detection](#), um das automatisierte Erkennen von Kostenanomalien mit Ursachenanalyse zu erhalten.

Ressourcen

Zugehörige Dokumente:

- [Was ist AWS Billing and Cost Management?](#)
- [Kostenoptimierung mit AWS](#)
- [Wahl einer AWS Kostenmanagementstrategie](#)
- [Ein Leitfaden für Anfänger zum AWS Kostenmanagement](#)
- [Eine detaillierte Übersicht über das Cost Intelligence Dashboard](#)
- [AWS -Architekturzentrum](#)
- [AWS -Lösungsbibliothek](#)
- [AWS Knowledge Center](#)

Zugehörige Videos:

- [This is my Architecture](#)

- [AWS re:Invent 2023 — Was ist neu bei AWS der Kostenoptimierung](#)
- [AWS re:Invent 2023 — Optimieren Sie Kosten und Leistung und verfolgen Sie die Fortschritte bei der Minderung](#)
- [AWS re:Invent 2023 — bewährte Methoden zur Optimierung der Speicherkosten AWS](#)
- [AWS re:Invent 2023 — Optimieren Sie die Kosten in Ihren Umgebungen mit mehreren Konten](#)

Zugehörige Beispiele:

- [AWS Compute Optimizer Demo-Code](#)
- [Workshop zur Kostenoptimierung](#)
- [Technische Playbooks zur Implementierung von Cloud Financial Management](#)
- [Startoptimierung: Optimierung der Anwendungsleistung für maximale Effizienz](#)
- [Workshop zur Serverless-Optimierung \(Leistung und Kosten\)](#)
- [Skalierung kostengünstiger Architekturen](#)

PERF01-BP04 Evaluieren, wie sich Kompromisse auf Kunden und Architektureffizienz auswirken

Ermitteln Sie beim Evaluieren von leistungsbezogenen Verbesserungen, welche gewählten Optionen sich auf Ihre Kunden und die Effizienz der Workloads auswirken. Wenn sich die Systemleistung beispielsweise bei Verwendung eines Schlüssel-Wert-Datenspeichers erhöht, sollten Sie unbedingt ermitteln, welche Auswirkungen sich bei einem dauerhaften Einsatz für die Kunden ergeben würden.

Typische Anti-Muster:

- Sie gehen davon aus, dass alle Leistungsgewinne implementiert werden sollten, auch wenn es Kompromisse für die Implementierung gibt.
- Änderungen an Workloads werden nur dann ausgewertet, wenn ein Leistungsproblem einen kritischen Punkt erreicht hat.

Vorteile der Nutzung dieser bewährten Methode: Wenn Sie potenzielle leistungsbezogene Verbesserungen bewerten, müssen Sie entscheiden, ob die Kompromisse für die Änderungen angesichts der Workload-Anforderungen akzeptabel sind. In einigen Fällen müssen Sie möglicherweise zusätzliche Kontrollen implementieren, um Kompromisse zu kompensieren.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Identifizieren Sie kritische Bereiche in der Architektur in Bezug auf Leistung und Kundenauswirkung. Stellen Sie fest, welche Verbesserungen möglich und welche Kompromisse damit verbunden sind und wie sich diese auf das System und das Benutzererlebnis auswirken. So lässt sich beispielsweise durch Caching von Daten die Leistung deutlich steigern. Es ist aber eine eindeutige Strategie erforderlich, mit der festgelegt wird, wie und wann Cache-Daten aktualisiert oder ungültig werden, um unerwünschtes Systemverhalten zu verhindern.

Implementierungsschritte

- Verstehen Sie Ihre Workload-Anforderungen und SLAs.
- Definieren Sie klare Bewertungsfaktoren. Faktoren können sich auf Kosten, Zuverlässigkeit, Sicherheit und Leistung der Workload beziehen.
- Wählen Sie die Architektur und Services, die Ihren Anforderungen entsprechen.
- Führen Sie Experimente und Machbarkeitsstudien (POCs) durch, um Kompromissfaktoren und Auswirkungen auf Kunden und Architektureffizienz zu bewerten. In der Regel verbrauchen hochverfügbare, leistungsstarke und sichere Workloads mehr Cloud-Ressourcen und bieten gleichzeitig ein besseres Kundenerlebnis. Machen Sie sich ein Bild von den Kompromissen in Bezug auf Komplexität, Leistung und Kosten Ihrer Workloads. In der Regel geht die Priorisierung von zwei der Faktoren auf Kosten des dritten.

Ressourcen

Zugehörige Dokumente:

- [Amazon Builders' Library](#)
- [Quick-KPIs](#)
- [Amazon CloudWatch RUM](#)
- [X-Ray-Dokumentation](#)
- [Resilienzmuster und Kompromisse verstehen, um eine effiziente Architektur in der Cloud zu entwickeln](#)

Zugehörige Videos:

- [Optimize applications through Amazon CloudWatch RUM](#)

- [AWS re:Invent 2023 – Kapazität, Verfügbarkeit, Kosteneffizienz: Wählen Sie drei Optionen aus](#)
- [AWS re:Invent 2023 – Erweiterte Integrationsmuster und Kompromisse für lose gekoppelte Systeme](#)

Zugehörige Beispiele:

- [Messen der Seitenladezeit mit Amazon CloudWatch Synthetics](#)
- [Webclient von Amazon CloudWatch RUM](#)

PERF01-BP05 Benutzungsrichtlinien und Referenzarchitekturen

Verwenden Sie interne Richtlinien und vorhandene Referenzarchitekturen bei der Auswahl von Services und Konfigurationen, um die Workload effizienter zu gestalten und zu implementieren.

Typische Anti-Muster:

- Sie erlauben eine Vielzahl von Technologien, was sich auf den Verwaltungsaufwand Ihres Unternehmens auswirken kann.

Vorteile der Nutzung dieser bewährten Methode: Durch Festlegung einer Richtlinie für die Architektur-, Technologie und Anbietersauswahl können Entscheidungen schnell getroffen werden.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Interne Richtlinien bei der Auswahl von Ressourcen und Architektur bieten Standards und Leitlinien, die bei Architekturentscheidungen zu beachten sind. Diese Richtlinien vereinfachen den Entscheidungsprozess bei der Auswahl des richtigen Cloud-Service und können zur Verbesserung der Leistungseffizienz beitragen. Stellen Sie die Workload mithilfe von Richtlinien oder Referenzarchitekturen bereit. Integrieren Sie die Services in Ihre Cloud-Bereitstellung. Überprüfen Sie anschließend anhand von Leistungstests, dass Sie die eigenen Leistungsanforderungen weiterhin erfüllen können.

Implementierungsschritte

- Verstehen Sie die Anforderungen der Cloud-Workload genau.
- Überprüfen Sie die internen und externen Richtlinien, um die relevantesten zu ermitteln.

- Verwenden Sie die entsprechenden Referenzarchitekturen, die von AWS bereitgestellt werden, oder die branchenweit anerkannten bewährten Methoden.
- Schaffen Sie ein Kontinuum, das aus Richtlinien, Standards, Referenzarchitekturen und präskriptiven Richtlinien für häufig auftretende Situationen besteht. Auf diese Weise können Ihre Teams schneller vorankommen. Passen Sie die Komponenten gegebenenfalls an die Branche an.
- Prüfen Sie diese Richtlinien und Referenzarchitekturen für die Workload in Sandbox-Umgebungen.
- Halten Sie up-to-date sich an Industriestandards und AWS Updates, um sicherzustellen, dass Ihre Richtlinien und Referenzarchitekturen zur Optimierung Ihrer Cloud-Workloads beitragen.

Ressourcen

Zugehörige Dokumente:

- [AWS -Architekturzentrum](#)
- [AWS Partner Network](#)
- [AWS -Lösungsbibliothek](#)
- [AWS Knowledge Center](#)
- [AWS Architektur-Blog](#)

Zugehörige Videos:

- [This is my Architecture](#)
- [AWS re:Invent 2022 — Steigern Sie den Wert Ihres Unternehmens mit SAP Referenzarchitektur AWS](#)

Zugehörige Beispiele:

- [AWS Beispiele](#)
- [AWS SDKBeispiele](#)

PERF01-BP06 Verwenden von Benchmarking, um architektonische Entscheidungen zu treffen

Führen Sie einen Benchmark-Vergleich für eine vorhandene Workload durch, um sich ein Bild über deren Leistung in der Cloud zu verschaffen, und treffen Sie architektonische Entscheidungen auf der Grundlage dieser Daten.

Typische Anti-Muster:

- Sie verlassen sich auf gängige Benchmarks, die für die Workload-Merkmale nicht aufschlussreich sind.
- Sie verlassen sich auf Kundenfeedback und Kundenwahrnehmung als einzige Benchmark.

Vorteile der Einführung dieser bewährten Methode: Durch Benchmarking Ihrer aktuellen Implementierung können Sie Leistungsverbesserungen messen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Kombinieren Sie Benchmarking mit synthetischen Tests, um die Leistung Ihrer Workload-Komponenten zu bewerten. Benchmarking lässt sich in der Regel schneller als Lasttests einrichten und dient zur Bewertung der Technologie einer bestimmten Komponente. Ein Benchmarking wird oft zu Beginn eines neuen Projekts durchgeführt, wenn Sie noch keine vollständige Lösung für einen Lasttest haben.

Sie können wahlweise eigene Benchmark-Tests erstellen oder einen branchenüblichen Standardtest wie etwa [TPC-DS](#) für das Benchmarking Ihrer Workloads verwenden. Branchen-Benchmarks sind zum Vergleich von Umgebungen nützlich. Benutzerdefinierte Benchmarks eignen sich zum Prüfen spezieller Arten von Vorgängen, die Sie in der Architektur ausführen möchten.

Beim Benchmarking ist es wichtig, die Testumgebung entsprechend vorzubereiten, um aussagekräftige Ergebnisse zu erzielen. Führen Sie denselben Benchmark-Test mehrmals aus, um sicherzustellen, dass alle Varianzen im Laufe der Zeit ermittelt wurden.

Da sich Benchmarks in der Regel schneller als Lasttests ausführen lassen, können Sie früher in der Bereitstellungspipeline eingesetzt werden und schneller Feedback zu Leistungsabweichungen liefern. Wenn Sie eine wesentliche Veränderung einer Komponente oder eines Services bewerten, können Sie schnell ermitteln, ob der Aufwand für die Korrektur gerechtfertigt ist. Die Verwendung von Benchmarking in Verbindung mit Lasttests ist wichtig, da letztere Auskunft über die Leistung der Workload in der Produktion geben.

Implementierungsschritte

- Planen und Definieren:
 - Definieren Sie die Ziele, Baselines, Testszenarien, Metriken (wie CPU-Auslastung, Latenz oder Durchsatz) und KPIs für Ihren Benchmark.

- Konzentrieren Sie sich auf die Benutzeranforderungen in Bezug auf das Benutzererlebnis und Faktoren wie Reaktionszeit und Barrierefreiheit.
- Identifizieren Sie ein Benchmarking-Tool, das für Ihre Workload geeignet ist. Sie können AWS-Services (wie [Amazon CloudWatch](#)) oder ein Drittanbieter-Tool verwenden, das mit der Workload kompatibel ist.
- Konfiguration und Verwendung:
 - Richten Sie Ihre Umgebung ein und konfigurieren Sie Ihre Ressourcen.
 - Implementieren Sie Überwachungs- und Protokollierungsfunktionen, um Testergebnisse zu erfassen.
- Benchmarking und Überwachung:
 - Führen Sie die Benchmark-Tests durch und überwachen Sie die Metriken während des Tests.
- Analyse und Dokumentation:
 - Dokumentieren Sie Ihren Benchmarking-Prozess und die entsprechenden Erkenntnisse.
 - Analysieren Sie die Ergebnisse, um Engpässe, Trends und Verbesserungsmöglichkeiten zu identifizieren.
 - Verwenden Sie die Testergebnisse, um die Architektur betreffende Entscheidungen zu fällen und die Workload anzupassen. Dies kann die Änderung von Services oder die Einführung neuer Features beinhalten.
- Optimierung und Wiederholung:
 - Passen Sie die Ressourcenkonfigurationen und -zuweisungen auf der Grundlage Ihrer Benchmarks an.
 - Testen Sie Ihre Workload nach der Anpassung erneut, um Ihre Verbesserungen zu überprüfen.
 - Dokumentieren Sie Ihre Erkenntnisse und wiederholen Sie den Prozess, um weitere Verbesserungsmöglichkeiten zu identifizieren.

Ressourcen

Zugehörige Dokumente:

- [AWS Architekturzentrum](#)
- [AWS Partner Network](#)
- [AWS Lösungsbibliothek](#)
- [AWS Knowledge Center](#)

- [Amazon CloudWatch RUM](#)
- [Amazon CloudWatch Synthetics](#)
- [Genomics workflows, Part 5: automated benchmarking](#)
- [Benchmark and optimize endpoint deployment in Amazon SageMaker AI JumpStart](#)

Zugehörige Videos:

- [AWS re:Invent 2023 – Benchmarking AWS Lambda cold starts](#)
- [Benchmarking stateful services in the cloud](#)
- [This is my Architecture](#)
- [Optimize applications through Amazon CloudWatch RUM](#)
- [Demo von Amazon CloudWatch Synthetics](#)

Zugehörige Beispiele:

- [AWS Beispiele](#)
- [AWS-SDK-Beispiele](#)
- [Verteilte Lasttests](#)
- [Messen der Seitenladezeit mit Amazon CloudWatch Synthetics](#)
- [Webclient von Amazon CloudWatch RUM](#)

PERF01-BP07 Verwenden eines datengesteuerten Ansatzes für architektonische Entscheidungen

Definieren Sie einen klaren, datengesteuerten Ansatz für architektonische Entscheidungen, um sicherzustellen, dass die richtigen Cloud-Services und -Konfigurationen verwendet werden, um Ihre spezifischen Geschäftsanforderungen zu erfüllen.

Typische Anti-Muster:

- Sie gehen davon aus, dass die aktuelle Architektur statisch ist und im Laufe der Zeit nicht aktualisiert werden sollte.
- Ihre architektonischen Entscheidungen basieren auf Vermutungen und Annahmen.
- Sie führen im Laufe der Zeit Änderungen an der Architektur ein, ohne sie zu begründen.

Vorteile der Nutzung dieser bewährten Methode: Durch einen klar definierten Ansatz für architektonische Entscheidungen verwenden Sie Daten, um das Workload-Design zu beeinflussen und im Laufe der Zeit fundierte Entscheidungen zu treffen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Nutzen Sie interne Erfahrungen und Kenntnisse im Zusammenhang mit der Cloud oder ziehen Sie externe Ressourcen heran, wie etwa veröffentlichte Anwendungsbeispiele oder Whitepapers, um Ressourcen und Services in der Architektur auszuwählen. Sie sollten über einen klar definierten Prozess verfügen, der das Experimentieren und Benchmarking mit den Services fördert, die in der Workload verwendet werden könnten.

Backlogs für kritische Workloads sollten nicht nur aus Benutzerszenarien bestehen, die für das Unternehmen und die Benutzer relevante Funktionen bereitstellen, sondern auch aus technischen Szenarien, die ein architektonisches System für die Workload bilden. Dieses System stützt sich auf neue technologische Fortschritte sowie neue Services und nimmt diese auf der Grundlage von Daten und entsprechender Begründung an. Dies stellt sicher, dass die Architektur zukunftssicher bleibt und nicht stagniert.

Implementierungsschritte

- Arbeiten Sie mit wichtigen Stakeholdern zusammen, um die Workload-Anforderungen zu definieren, einschließlich Überlegungen zu Leistung, Verfügbarkeit und Kosten. Berücksichtigen Sie Faktoren wie die Anzahl der Benutzer und das Nutzungsmuster für die Workload.
- Erstellen Sie ein Architektursystem oder einen Technologie-Backlog, der zusammen mit dem funktionalen Backlog priorisiert wird.
- Bewerten und beurteilen Sie verschiedene Cloud-Services (weitere Informationen finden Sie unter [PERF01-BP01 Informieren über verfügbare Cloud-Services und -Features](#)).
- Erkunden Sie verschiedene Architekturmuster wie Microservices oder Serverless, die Ihren Leistungsanforderungen entsprechen (weitere Informationen finden Sie unter [PERF01-BP02 Einholen von Rat beim Cloud-Anbieter oder einem geeigneten Partner, um mehr über Architekturmuster und bewährte Methoden zu erfahren](#)).
- Ziehen Sie andere Teams, Architekturdiagramme und Ressourcen zu Rate, z. B. AWS Solution Architects, [AWS-Architekturzentrum](#) und [AWS Partner Network](#), um die richtige Architektur für Ihre Workload auszuwählen.

- Definieren Sie Leistungsmetriken wie Durchsatz und Reaktionszeit, anhand derer Sie die Leistung der Workload bewerten können.
- Experimentieren Sie und verwenden Sie definierte Metriken, um die Leistung der ausgewählten Architektur zu validieren.
- Überwachen Sie kontinuierlich und nehmen Sie bei Bedarf Anpassungen vor, um die optimale Leistung der Architektur aufrechtzuerhalten.
- Dokumentieren Sie Ihre gewählte Architektur und Entscheidungen als Referenz für zukünftige Updates und Erkenntnisse.
- Überprüfen und aktualisieren Sie den Ansatz zur Architekturauswahl kontinuierlich auf der Grundlage von Erkenntnissen, neuen Technologien und Metriken, die auf eine notwendige Änderung oder ein Problem im aktuellen Ansatz hinweisen.

Ressourcen

Zugehörige Dokumente:

- [AWS-Lösungsbibliothek](#)
- [AWS Knowledge Center](#)
- [Architekturmodelle für die Erstellung von datengesteuerten End-to-End-Anwendungen in AWS](#)

Zugehörige Videos:

- [This is my Architecture](#)
- [AWS re:Invent 2021 – Das datengesteuerte Unternehmen: Von der Vision zum Mehrwert](#)
- [AWS re:Invent 2022 – Bereitstellung nachhaltiger, leistungsstarker Architekturen](#)
- [AWS re:Invent 2023 – Optimieren der Kosten und Leistung sowie Verfolgen der Fortschritte bei der Schadensbegrenzung](#)
- [AWS re:Invent 2022 – AWS-Optimierung: Umsetzbare Schritte für sofortige Ergebnisse](#)

Zugehörige Beispiele:

- [AWS Beispiele](#)
- [AWS SDK-Beispiele](#)

Computer und Hardware

Fragen

- [LEIST 2. Wie wählen und nutzen Sie Computing-Ressourcen für Ihre Workload?](#)

LEIST 2. Wie wählen und nutzen Sie Computing-Ressourcen für Ihre Workload?

Die optimale Datenverarbeitungsoption für eine bestimmte Workload kann sich je nach Anwendungsdesign, Nutzungsmustern und Konfigurationseinstellungen unterscheiden. Architekturen können verschiedene Datenverarbeitungsoptionen für verschiedene Komponenten verwenden und verschiedene Features zur Verbesserung der Leistung bieten. Die Wahl der falschen Datenverarbeitungslösung für eine Architektur kann die Leistungseffizienz schmälern.

Best Practices

- [PERF02-BP01 Auswählen der besten Datenverarbeitungsoptionen für die Workload](#)
- [PERF02-BP02 Verstehen verfügbarer Konfigurationen und Features für die Datenverarbeitung](#)
- [PERF02-BP03 Erfassung rechnerbezogener Metriken](#)
- [PERF02-BP04 Konfigurieren und richtiges Dimensionieren von Datenverarbeitungsressourcen](#)
- [PERF02-BP05 Dynamisches Skalieren von Datenverarbeitungsressourcen](#)
- [PERF02-BP06 Verwenden von optimierten hardwarebasierten Datenverarbeitungsbeschleunigern](#)

PERF02-BP01 Auswählen der besten Datenverarbeitungsoptionen für die Workload

Wenn Sie die für die Workload am besten geeignete Datenverarbeitungsoption auswählen, können Sie die Leistung verbessern, unnötige Infrastrukturkosten reduzieren und den Betriebsaufwand für die Aufrechterhaltung der Workload senken.

Typische Anti-Muster:

- Sie verwenden dieselbe Datenverarbeitungsoption, die On-Premises verwendet wurde.
- Ihnen fehlt es an Bewusstsein für Cloud-Datenverarbeitungsoptionen, -Features und -lösungen und wie diese Lösungen die Datenverarbeitungsleistung verbessern können.
- Sie stellen eine bestehende Datenverarbeitungsoption zu viel bereit, um Skalierungs- oder Leistungsanforderungen zu erfüllen, wenn eine alternative Datenverarbeitungsoption den Workload-Merkmalen besser entsprechen würde.

Vorteile der Nutzung dieser bewährten Methode: Durch die Ermittlung der Anforderungen an die Datenverarbeitung und deren Bewertung anhand der verfügbaren Optionen können Sie die Workload ressourceneffizienter gestalten.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Zur Optimierung der Cloud-Workloads im Hinblick auf Leistungseffizienz ist es wichtig, die am besten geeigneten Datenverarbeitungsoptionen für Ihren Anwendungsfall und Ihre Leistungsanforderungen auszuwählen. AWS bietet eine Vielzahl von Datenverarbeitungsoptionen, die auf unterschiedliche Workloads in der Cloud zugeschnitten sind. Sie können beispielsweise [Amazon EC2](#) verwenden, um virtuelle Server zu starten und zu verwalten, [AWS Lambda](#), um Code auszuführen, ohne Server bereitstellen oder verwalten zu müssen, [Amazon ECS](#) oder [Amazon EKS](#), um Container auszuführen und zu verwalten, oder [AWS Batch](#), um große Datenmengen parallel zu verarbeiten. Basierend auf Ihren Skalierungs- und Datenverarbeitungsanforderungen sollten Sie die optimale Datenverarbeitungslösung für Ihre Situation auswählen und konfigurieren. Sie können auch erwägen, mehrere Arten von Datenverarbeitungslösungen in einer einzigen Workload zu verwenden, da jede ihre eigenen Vor- und Nachteile hat.

Die folgenden Schritte führen Sie durch die Auswahl der richtigen Datenverarbeitungsoptionen, die Ihren Workload-Eigenschaften und Leistungsanforderungen entsprechen.

Implementierungsschritte

- Verstehen Sie Ihre Workload-Datenverarbeitungsanforderungen. Die zu berücksichtigenden wesentlichen Anforderungen umfassen Anforderungen an Datenverarbeitung, Datenverkehrsmuster, Datenzugriffsmuster, Skalierung und Latenz.
- Erfahren Sie mehr über verschiedene [AWS-Datenverarbeitungsservices](#) für Ihre Workload. Weitere Informationen finden Sie unter [PERF01-BP01 Informieren über verfügbare Cloud-Services und -Features](#). Hier finden Sie einige wichtige AWS-Datenverarbeitungsoptionen, ihre Eigenschaften und gängige Anwendungsfälle:

AWS-Service	Schlüsselmerkmale	Häufige Anwendungsfälle
Amazon Elastic Compute Cloud (Amazon EC2)	Verfügt über eine spezielle Option für Hardware, Lizenzanforderungen, eine große Auswahl an	Lift-and-Shift-Migrationen, monolithische Anwendung, hybride Umgebungen, Enterprise-Anwendungen

AWS-Service	Schlüsselmerkmale	Häufige Anwendungsfälle
	verschiedenen Instance-Familien, Prozessortypen und Beschleuniger der Datenverarbeitung	
Amazon Elastic Container Service (Amazon ECS) , Amazon Elastic Kubernetes Service (Amazon EKS)	Einfache Bereitstellung, konsistente Umgebungen, skalierbar	Microservices, Hybrid-Umgebungen
AWS Lambda	Serverless-Datenverarbeitungsservice , der Code als Reaktion auf Ereignisse ausführt und die zugrunde liegenden Ressourcen für die Datenverarbeitung automatisch verwaltet.	Microservices, ereignisgesteuerte Anwendungen
AWS Batch	Effiziente und dynamische Bereitstellung und Skalierung von Amazon Elastic Container Service (Amazon ECS) , Amazon Elastic Kubernetes Service (Amazon EKS) und AWS Fargate -Ressourcen der Datenverarbeitung, mit der Option, On-Demand- oder Spot Instances je nach Ihren Jobanforderungen zu verwenden	HPC, ML-Modelle trainieren
Amazon Lightsail	Vorkonfigurierte Linux- und Windows-Anwendung für die Ausführung kleiner Workloads	Einfache Webanwendungen, benutzerdefinierte Website

- Bewerten Sie die Kosten (wie stündliche Gebühr oder Datenübertragung) und den Verwaltungsaufwand (wie Patching und Skalierung), die mit jeder Datenverarbeitungsoption verbunden sind.
- Führen Sie Experimente und Benchmarking in einer Nicht-Produktionsumgebung durch, um herauszufinden, welche Datenverarbeitungsoption Ihre Workload-Anforderungen am besten erfüllt.
- Nachdem Sie experimentiert und die neue Datenverarbeitungslösung ermittelt haben, planen Sie die Migration und überprüfen Sie die Leistungsmetriken.
- Verwenden Sie AWS-Überwachungstools wie [Amazon CloudWatch](#) und Optimierungsservices wie [AWS Compute Optimizer](#), um die Datenverarbeitungsressourcen kontinuierlich auf der Grundlage realer Nutzungsmuster zu optimieren.

Ressourcen

Zugehörige Dokumente:

- [Cloud Computing mit AWS](#)
- [Amazon-EC2-Instance-Typen](#)
- [Amazon EKS Containers: Amazon EKS Worker Nodes](#)
- [Amazon ECS Containers: Amazon ECS Container Instances](#)
- [Funktionen: Lambda-Funktionskonfiguration](#)
- [Prescriptive Guidance for Containers](#)
- [Prescriptive Guidance for Serverless](#)

Zugehörige Videos:

- [AWS re:Invent 2023 – AWS Graviton: Das beste Preis-Leistungs-Verhältnis für Ihre AWS-Workloads](#)
- [AWS re:Invent 2023 – New Amazon Elastic Compute Cloud generative AI capabilities in AMS](#)
- [AWS re:Invent 2023 – What's new with Amazon Elastic Compute Cloud](#)
- [AWS re:Invent 2023 – Smart savings: Amazon Elastic Compute Cloud cost-optimization strategies](#)
- [AWS re:Invent 2021 – Powering next-gen Amazon Elastic Compute Cloud: Deep dive on the Nitro System](#)
- [AWS re:Invent 2019 – Optimieren von Leistung und Kosten für die Datenverarbeitung bei AWS](#)

- [AWS re:Invent 2.019 – Amazon Elastic Compute Cloud foundations](#)
- [AWS re:Invent 2022 – ML-Modelle für Inferenz mit hoher Leistung und niedrigen Kosten bereitstellen](#)
- [AWS re:Invent 2019 – Optimieren von Leistung und Kosten für die Datenverarbeitung bei AWS](#)
- [Grundlagen von Amazon EC2](#)
- [Stellen Sie ML-Modelle für Inference mit hoher Leistung und niedrigen Kosten bereit](#)

Zugehörige Beispiele:

- [Migration der Webanwendung zu Containern](#)
- [Ausführen eines Serverless-„Hello World“](#)
- [Workshop zu Amazon EKS](#)
- [Workshop zu Amazon EC2](#)
- [Effiziente und belastbare Workloads mit Amazon Elastic Compute Cloud Auto Scaling](#)
- [Migration zu AWS Graviton mit Container Services](#)

PERF02-BP02 Verstehen verfügbarer Konfigurationen und Features für die Datenverarbeitung

Informieren Sie sich über die verfügbaren Konfigurationsoptionen und Features für den Datenverarbeitungsservice, damit Sie die richtige Menge an Ressourcen bereitstellen und die Leistungseffizienz verbessern können.

Typische Anti-Muster:

- Sie bewerten keine Datenverarbeitungsoptionen oder verfügbaren Instance-Familien anhand der Workload-Merkmale.
- Sie stellen zu viele Datenverarbeitungsressourcen bereit, um Anforderungen von Nachfragespitzen zu erfüllen.

Vorteile der Nutzung dieser bewährten Methode: Machen Sie sich mit den AWS-Features und -Konfigurationen für die Datenverarbeitung vertraut, sodass Sie eine Datenverarbeitungslösung verwenden können, die für die Workload-Merkmale und -Anforderungen optimiert ist.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Jede Datenverarbeitungslösung verfügt über einzigartige Konfigurationen und Features, um unterschiedliche Workload-Merkmale und -Anforderungen zu unterstützen. Erfahren Sie, wie diese Optionen die Workload ergänzen, und finden Sie heraus, welche Konfigurationsoptionen am besten für Ihre Anwendung geeignet sind. Beispiele für diese Optionen sind Instance-Familien, Größen, Features (GPU, E/A), Bursting, Zeitüberschreitungen, Funktionsgrößen, Container-Instances und Gleichzeitigkeit. Wenn Ihre Workload die gleiche Rechenoption für mehr als vier Wochen verwendet hat und sie davon ausgehen, dass die Eigenschaften in Zukunft gleich bleiben, können Sie mithilfe von [AWS Compute Optimizer](#) herausfinden, ob Ihre aktuelle Datenverarbeitungsoption aus CPU- und Speicherebene für die Workloads geeignet ist.

Implementierungsschritte

- Verstehen Sie die Workload-Anforderungen (wie CPU-Bedarf, Arbeitsspeicher und Latenz).
- Lesen Sie die AWS-Dokumentation und die bewährten Methoden, um mehr über empfohlene Konfigurationsoptionen zu erfahren, mit denen Sie die Rechenleistung verbessern können. Hier finden Sie einige wichtige Konfigurationsoptionen, die Sie in Betracht ziehen sollten:

Konfigurationsoption	Beispiele
Instance-Typ	• Für Datenverarbeitung optimierte Instances eignen sich ideal für Workloads, die ein hohes vCPU-/Arbeitsspeicherverhältnis erfordern. • Arbeitsspeicheroptimierte Instances bieten große Mengen an Arbeitsspeicher, um arbeitsspeicherintensive Workloads zu unterstützen. • Speicheroptimierte Instances wurden für Workloads entworfen, die hohen, sequenziellen Lese- und Schreibzugriff (IOPS) auf lokalen Speicher erfordern.
Preismodell	• Mit On-Demand-Instances können Sie die Datenverarbeitungskapazität nach Sekunde oder Stunde ohne langfristige Verpflichtung

Konfigurationsoption	Beispiele
	tungen verwenden. Diese Instances eignen sich für Bursting über die Leistungsbasis hinaus. • Savings Plans bieten erhebliche Einsparungen gegenüber On-Demand-Instances im Austausch gegen die Verpflichtung, eine bestimmte Menge an Rechenleistung für einen Zeitraum von ein oder drei Jahren zu nutzen. • Spot Instances ermöglichen es Ihnen, ungenutzte Instance-Kapazitäten mit einem Rabatt für Ihre zustandslosen, fehlertoleranten Workloads zu nutzen.
Auto Scaling	Nutzen Sie die Auto-Scaling-Konfiguration zur Anpassung der Datenverarbeitungsressourcen an die Datenverkehrsmuster.
Dimensionierung	• Nutzen Sie Compute Optimizer zum Erhalt von Machine-Learning-gestützten Empfehlungen dazu, welche Datenverarbeitungskonfiguration am besten Ihren Datenverarbeitungsmerkmalen entspricht. • Mit AWS Lambda Power Tuning können Sie die beste Konfiguration für Ihre Lambda-Funktion auswählen.
Hardwarebasierte Computing-Beschleuniger	• Beschleunigte Computing-Instances führen Funktionen wie die Grafikverarbeitung oder Datenmusterzuordnung effizienter aus als CPU-basierte Alternativen. • Nutzen Sie für Machine-Learning-Workloads spezielle Hardware, die auf Ihre Workload abgestimmt ist, z. B. AWS Trainium, AWS Inferentia und Amazon EC2 DL1.

Ressourcen

Zugehörige Dokumente:

- [Cloud Computing mit AWS](#)
- [Amazon-EC2-Instance-Typen](#)
- [Processor State Control for Your Amazon EC2 Instance](#)
- [Amazon EKS Containers: Amazon EKS Worker Nodes](#)
- [Amazon ECS Containers: Amazon ECS Container Instances](#)
- [Funktionen: Lambda-Funktionskonfiguration](#)

Zugehörige Videos:

- [AWS re:Invent 2023 – AWS Graviton: Das beste Preis-Leistungs-Verhältnis für Ihre AWS-Workloads](#)
- [AWS re:Invent 2023 – New Amazon EC2 generative AI capabilities in AWS-Managementkonsole](#)
- [AWS re:Invent 2023 – What's new with Amazon EC2](#)
- [AWS re:Invent 2023 – Smart savings: Amazon EC2 cost-optimization strategies](#)
- [AWS re:Invent 2021 – Powering next-gen Amazon EC2: Deep dive on the Nitro System](#)
- [AWS re:Invent 2019 – Amazon EC2 foundations](#)
- [AWS re:Invent 2022 – Optimizing Amazon EKS for performance and cost on AWS](#)

Zugehörige Beispiele:

- [Demo-Code für Compute Optimizer](#)
- [Workshop zu Spot Instances von Amazon EC2](#)
- [Effiziente und belastbare Workloads mit Amazon EC2 AWS Auto Scaling](#)
- [Workshop für Graviton-Entwickler](#)
- [AWS für Microsoft-Workloads Immersion Day](#)
- [AWS für Linux-Workloads Immersion Day](#)
- [AWS Compute Optimizer-Demo-Code](#)
- [Workshop zu Amazon EKS](#)

PERF02-BP03 Erfassung rechnerbezogener Metriken

Erfassen und verfolgen Sie Datenverarbeitungsmetriken, um die Leistung der Rechenressourcen besser zu verstehen und deren Leistung und Auslastung zu verbessern.

Typische Anti-Muster:

- Sie suchen ausschließlich manuell mithilfe von Protokolldateien nach Metriken.
- Sie verwenden nur die Standardmetriken, die von der Überwachungssoftware aufgezeichnet wurden.
- Sie überprüfen Metriken nur dann, wenn ein Problem vorliegt.

Vorteile der Nutzung dieser bewährten Methode: Die Erfassung von Leistungsmetriken hilft Ihnen dabei, die Anwendungsleistung an den Geschäftsanforderungen auszurichten, um sicherzustellen, dass Sie Ihre Workload-Anforderungen erfüllen. Es kann Ihnen auch dabei helfen, die Ressourcenleistung und -nutzung in der Workload kontinuierlich zu verbessern.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Cloud-Workloads können große Mengen an Daten generieren, wie Metriken, Protokolle und Ereignisse. In der AWS Cloud ist das Sammeln von Metriken ein entscheidender Schritt zur Verbesserung von Sicherheit, Kosteneffizienz, Leistung und Nachhaltigkeit. AWS bietet eine breite Palette leistungsbezogener Kennzahlen mithilfe von Überwachungsdiensten wie [Amazon](#), CloudWatch um Ihnen wertvolle Einblicke zu bieten. Metriken wie CPU Auslastung, Speicherauslastung, Festplatten-I/O sowie eingehende und ausgehende Netzwerkdaten können Aufschluss über Auslastungsgrade oder Leistungsengpässe geben. Nutzen Sie diese Metriken im Rahmen eines datengestützten Ansatzes, der Ihnen die aktive Feinabstimmung und Optimierung der von der Workload genutzten Ressourcen ermöglicht. Im Idealfall sollten Sie alle Metriken zu Ihren Datenverarbeitungsressourcen auf einer einzigen Plattform erfassen und Aufbewahrungsrichtlinien implementieren, um Kosten- und Betriebsziele zu unterstützen.

Implementierungsschritte

- Identifizieren Sie, welche Leistungsmetriken für Ihre Workload relevant sind. Sie sollten Metriken zur Ressourcennutzung und zum Betrieb der Cloud-Workload (wie Reaktionszeit und Durchsatz) erfassen.
 - [EC2Amazon-Standardmetriken](#)

- [ECSAmazon-Standardmetriken](#)
- [EKSAAmazon-Standardmetriken](#)
- [Lambda-Standardmetriken](#)
- [EC2Speicher- und Festplattenmetriken von Amazon](#)
- Wählen Sie die richtige Protokollierungs- und Überwachungslösung für Ihre Workload aus und richten Sie sie ein.
 - [AWS -native Beobachtbarkeit](#)
 - [AWS Distribution für OpenTelemetry](#)
 - [Amazon Managed Service for Prometheus](#)
- Definieren Sie den erforderlichen Filter und die erforderliche Aggregation für die Metriken auf der Grundlage Ihrer Workload-Anforderungen.
 - [Quantifizieren Sie benutzerdefinierte Anwendungsmetriken mit Amazon CloudWatch Logs und Metrikfiltern](#)
 - [Sammeln Sie benutzerdefinierte Metriken mit CloudWatch strategischem Tagging von Amazon](#)
- Konfigurieren Sie Richtlinien zur Datenaufbewahrung für Ihre Metriken so, dass sie Ihren Sicherheits- und Betriebszielen entsprechen.
 - [Standardmäßige Datenspeicherung für Metriken CloudWatch](#)
 - [Standardmäßige Datenspeicherung für CloudWatch Protokolle](#)
- Erstellen Sie bei Bedarf Alarme und Benachrichtigungen für Ihre Metriken, damit Sie proaktiv auf leistungsbezogene Probleme reagieren können.
 - [Erstellen Sie mithilfe der CloudWatch Amazon-Anomalieerkennung Alarme für benutzerdefinierte Metriken](#)
 - [Erstellen Sie mit Amazon Metriken und Alarme für bestimmte Webseiten CloudWatch RUM](#)
- Verwenden Sie die Automatisierung, um die Kundendienstmitarbeiter für die Metrik- und Protokollaggregation einzusetzen.
 - [AWS Systems Manager Automatisierung](#)
 - [OpenTelemetrySammler](#)

Ressourcen

Zugehörige Dokumente:

- [Überwachung und Beobachtbarkeit](#)

- [Bewährte Methoden: Implementierung von Observability mit AWS](#)
- [CloudWatch Amazon-Dokumentation](#)
- [Erfassen Sie mit dem Agenten Metriken und Protokolle von EC2 Amazon-Instances und lokalen Servern CloudWatch](#)
- [Zugreifen auf Amazon CloudWatch Logs für AWS Lambda](#)
- [CloudWatch Logs mit Container-Instances verwenden](#)
- [Veröffentlichen von benutzerdefinierten Metriken](#)
- [AWS Answers: Zentralisierte Protokollierung](#)
- [AWS Dienste, die CloudWatch Metriken veröffentlichen](#)
- [Überwachung von Amazon EKS am AWS Fargate](#)

Zugehörige Videos:

- [AWS re:Invent 2023 — \[LAUNCH\] Anwendungsüberwachung für moderne Workloads](#)
- [AWS re:Invent 2023 — Implementierung der Anwendungsbeobachtbarkeit](#)
- [AWS re:Invent 2023 — Aufbau einer effektiven Strategie für Beobachtbarkeit](#)
- [AWS re:Invent 2023 — Nahtlose Beobachtbarkeit mit Distro für AWS OpenTelemetry](#)
- [Leistungsmanagement für Anwendungen aktiviert AWS](#)

Zugehörige Beispiele:

- [AWS Immersion Day für Linux-Workloads — Amazon CloudWatch](#)
- [Überwachung von ECS Amazon-Clustern und Containern](#)
- [Überwachung mit CloudWatch Amazon-Dashboards](#)
- [EKSAmazon-Werkstatt](#)

PERF02-BP04 Konfigurieren und richtiges Dimensionieren von Datenverarbeitungsressourcen

Konfigurieren und passen Sie die Größe der Datenverarbeitungsressourcen so an, dass sie den Leistungsanforderungen der Workloads entsprechen, und vermeiden Sie zu wenig oder zu stark ausgelastete Ressourcen.

Typische Anti-Muster:

- Sie ignorieren Ihre Workload-Leistungsanforderungen, was zu über- oder unterdimensionierten Datenverarbeitungsressourcen führt.
- Sie wählen nur die größte oder kleinste verfügbare Instance für alle Workloads aus.
- Sie verwenden nur eine Instance-Familie, um die Verwaltung zu vereinfachen.
- Sie ignorieren Empfehlungen von AWS Cost Explorer oder Compute Optimizer zur richtigen Dimensionierung.
- Sie bewerten die Workload nicht erneut auf die Eignung neuer Instance-Typen.
- Sie zertifizieren nur eine kleine Anzahl von Instance-Konfigurationen für Ihre Organisation.

Vorteile der Nutzung dieser bewährten Methode: Die richtige Dimensionierung der Datenverarbeitungsressourcen gewährleistet einen optimalen Betrieb in der Cloud, indem eine Über- und Unterdimensionierung von Ressourcen vermieden wird. Die richtige Dimensionierung der Datenverarbeitungsressourcen führt in der Regel zu einer besseren Leistung und einem besseren Kundenerlebnis bei gleichzeitiger Senkung der Kosten.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Die richtige Dimensionierung ermöglicht es Organisationen, ihre Cloud-Infrastruktur effizient und kostengünstig zu betreiben und gleichzeitig ihre Geschäftsanforderungen zu erfüllen. Eine übermäßige Bereitstellung von Cloud-Ressourcen kann zu zusätzlichen Kosten führen, während eine unzureichende Bereitstellung zu schlechter Leistung und einem negativen Kundenerlebnis führen kann. AWS bietet Tools wie [AWS Compute Optimizer](#) und [AWS Trusted Advisor](#), die historische Daten nutzen, um Empfehlungen zur richtigen Größe Ihrer Rechenressourcen zu geben.

Implementierungsschritte

- Wählen Sie einen Instance-Typ, der am besten zu Ihren Anforderungen passt:
 - [How do I choose the appropriate Amazon EC2 instance type for my workload?](#)
 - [Attributbasierte Auswahl von Instance-Typen für Amazon-EC2-Flotte](#)
 - [Erstellen einer Auto-Scaling-Gruppe mit attributbasierter Auswahl des Instance-Typs](#)
 - [Optimieren Ihrer Kubernetes-Datenverarbeitungskosten mit der Karpenter-Konsolidierung](#)
- Analysieren Sie die verschiedenen Leistungsmerkmale Ihrer Workload und bewerten Sie, wie sich diese auf Arbeitsspeicher, Netzwerk und CPU-Auslastung auswirken. Wählen Sie anhand dieser

Daten die für das Profil und die Leistungsziele der Workloads am besten geeigneten Ressourcen aus.

- Überwachen Sie Ihren Ressourcenverbrauch mithilfe von AWS-Überwachungstools wie Amazon CloudWatch.
- Wählen Sie die richtige Konfiguration für die Datenverarbeitungsressource aus.
 - Prüfen Sie für kurzzeitige Workloads die [Amazon-CloudWatch-Metriken der Instance](#) wie CPUUtilization, um festzustellen, ob die Instance zu wenig oder zu viel genutzt wird.
 - Prüfen Sie für stabile Workloads in regelmäßigen Intervallen AWS-Dimensionierungstools wie etwa AWS Compute Optimizer und AWS Trusted Advisor, um Möglichkeiten zur Optimierung und zur korrekten Dimensionierung der Datenverarbeitungsressource zu erkennen.
- Testen Sie Konfigurationsänderungen in einer Nicht-Produktionsumgebung, bevor Sie sie in einer Live-Umgebung implementieren.
- Bewerten Sie neue Datenverarbeitungsangebote und vergleichen Sie sie mit den Anforderungen Ihrer Workload.

Ressourcen

Zugehörige Dokumente:

- [Cloud Computing mit AWS](#)
- [Amazon-EC2-Instance-Typen](#)
- [Amazon ECS Containers: Amazon ECS Container Instances](#)
- [Amazon EKS Containers: Amazon EKS Worker Nodes](#)
- [Funktionen: Lambda-Funktionskonfiguration](#)
- [Processor State Control for Your Amazon EC2 Instance](#)

Zugehörige Videos:

- [Grundlagen von Amazon EC2](#)
- [AWS re:Invent 2023 – AWS Graviton: Das beste Preis-Leistungs-Verhältnis für Ihre AWS-Workloads](#)
- [AWS re:Invent 2023 – New Amazon EC2 generative AI capabilities in AWS-Managementkonsole](#)
- [AWS re:Invent 2023 – What's new with Amazon EC2](#)

- [AWS re:Invent 2023 – Smart savings: Amazon EC2 cost-optimization strategies](#)
- [AWS re:Invent 2021 – Powering next-gen Amazon EC2: Deep dive on the Nitro System](#)
- [AWS re:Invent 2019 – Amazon EC2 foundations](#)

Zugehörige Beispiele:

- [AWS Compute Optimizer-Demo-Code](#)
- [Workshop zu Amazon EKS](#)
- [Empfehlungen zur Dimensionierung](#)

PERF02-BP05 Dynamisches Skalieren von Datenverarbeitungsressourcen

Nutzen Sie die Elastizität der Cloud, um die Datenverarbeitungsressourcen dynamisch nach oben oder unten zu skalieren, um Ihren Bedürfnissen zu entsprechen und eine Über- oder Unterdimensionierung von Kapazitäten für die Workload zu vermeiden.

Typische Anti-Muster:

- Sie reagieren auf Alarme, indem Sie die Kapazität manuell erhöhen.
- Sie verwenden dieselben Dimensionierungsrichtlinien (in der Regel statische Infrastruktur) wie bei On-Premises.
- Sie belassen die erhöhte Kapazität nach dem Hochskalieren, anstatt wieder herunterzuskalieren.

Vorteile der Nutzung dieser bewährten Methode: Durch das Konfigurieren und Testen der Elastizität von Datenverarbeitungsressourcen können Sie Geld sparen, Leistungsbenchmarks einhalten und die Zuverlässigkeit verbessern, wenn sich der Datenverkehr ändert.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

AWS bietet Ihnen die Flexibilität, Ressourcen dynamisch durch verschiedene Skalierungsmechanismen nach oben oder unten zu skalieren, um Bedarfsänderungen gerecht zu werden. In Kombination mit Datenverarbeitungsmetriken ermöglicht eine dynamische Skalierung Workloads, automatisch auf Änderungen zu reagieren und die optimalen Datenverarbeitungsressourcen zu nutzen, um die Zielvorgabe zu erreichen.

Sie können verschiedene Ansätze nutzen, um das Angebot an Ressourcen auf die Nachfrage abzustimmen.

- Ansatz zur Zielverfolgung: Überwachen Sie Ihre Skalierungsmetriken und erhöhen oder verringern Sie die Kapazität automatisch Ihrem Bedarf entsprechend.
- Prädiktive Skalierung: Skalieren Sie in Erwartung täglicher und wöchentlicher Trends.
- Zeitplanbasierter Ansatz: Legen Sie Ihren eigenen Skalierungszeitplan entsprechend vorhersehbaren Laständerungen fest.
- Skalierung von Services: Wählen Sie Services (wie Serverless), die auf automatische Skalierung ausgelegt sind.

Sie müssen sicherstellen, dass Workload-Bereitstellungen sowohl Hoch- als auch Herunterskalierungsereignisse verarbeiten können.

Implementierungsschritte

- Datenverarbeitungs-Instances, Container und Funktionen bieten Mechanismen für Elastizität, sei es in Kombination mit AutoScaling oder als Feature des Service. Hier finden Sie einige Beispiele für automatische Skalierungsmechanismen:

Autoscaling-Mechanismus	Aktion
Amazon EC2 Auto Scaling	Zur Sicherstellung, dass die richtige Anzahl verfügbarer Amazon-EC2 -Instances vorhanden ist, um die Benutzerlast für Ihre Anwendung zu bewältigen.
Application Auto Scaling	Zur automatischen Skalierung der Ressourcen für einzelne AWS-Services über Amazon EC2 hinaus, wie AWS Lambda -Funktionen oder Amazon Elastic Container Service (Amazon ECS) -Services.
Kubernetes Cluster Autoscaler/Karpenter	Zur automatischen Skalierung von Kubernetes-Clustern.

- Skalieren wird häufig im Zusammenhang mit Datenverarbeitungs-Services wie Amazon-EC2-Instances oder AWS Lambda-Funktionen behandelt. Denken Sie auch daran, die Konfiguration von

nicht Daten verarbeitenden Services in Betracht zu ziehen, z. B. [AWS Glue](#), um die Nachfrage zu decken.

- Stellen Sie sicher, dass die Metriken für die Skalierung den Merkmalen der bereitgestellten Workload entsprechen. Wenn Sie eine Anwendung zur Video-Transkodierung bereitstellen, wird eine CPU-Auslastung von 100 % erwartet, weshalb dies nicht die Hauptmetrik sein sollte. Verwenden Sie stattdessen die Tiefe der Aufgabenwarteschlange für die Transkodierung. Sie können bei Bedarf eine [benutzerdefinierte Metrik](#) für Ihre Skalierungsrichtlinie verwenden. Beachten Sie bei der Wahl geeigneter Metriken die folgenden Hinweise für Amazon EC2:
 - Es muss sich um eine gültige Nutzungsmetrik handeln, die beschreibt, wie stark eine Instance genutzt wird.
 - Der Wert der Metrik muss sich proportional zur Anzahl der Instances in der Auto-Scaling-Gruppe erhöhen oder verringern.
- Achten Sie darauf, für Ihre Auto-Scaling-Gruppe eine [dynamische Skalierung](#) anstelle einer [manuellen Skalierung](#) zu verwenden. Außerdem empfiehlt es sich, bei der dynamischen Skalierung [Skalierungsrichtlinien zur Zielverfolgung](#) zu verwenden.
- Prüfen Sie, ob Workload-Bereitstellungen mit beiden Skalierungen (nach oben und unten) umgehen können. Sie können beispielsweise den [Aktivitätsverlauf](#) verwenden, um eine Skalierungsaktivität für eine Auto-Scaling-Gruppe zu überprüfen.
- Evaluieren Sie Ihre Workload auf vorhersagbare Muster und skalieren Sie proaktiv, wenn Sie vorhergesagte und geplante Änderungen der Nachfrage erwarten. Mit der prädiktiven Skalierung können Sie die Notwendigkeit einer Überbereitstellung von Kapazitäten vermeiden. Weitere Einzelheiten finden Sie unter [Prädiktive Skalierung mit Amazon EC2 Auto Scaling](#).

Ressourcen

Zugehörige Dokumente:

- [Cloud Computing mit AWS](#)
- [Amazon-EC2-Instance-Typen](#)
- [Amazon ECS Containers: Amazon ECS Container Instances](#)
- [Amazon EKS Containers: Amazon EKS Worker Nodes](#)
- [Funktionen: Lambda-Funktionskonfiguration](#)
- [Processor State Control for Your Amazon EC2 Instance](#)
- [Deep Dive on Amazon ECS Cluster Auto Scaling](#)

- [Vorstellung von Karpenter – Open-Source-Kubernetes-Cluster-Autoscaler mit hoher Leistung](#)

Zugehörige Videos:

- [AWS re:Invent 2023 – AWS Graviton: Das beste Preis-Leistungs-Verhältnis für Ihre AWS-Workloads](#)
- [AWS re:Invent 2023 – New Amazon EC2 generative AI capabilities in AWS Management Console](#)
- [AWS re:Invent 2023 – What's new with Amazon EC2](#)
- [AWS re:Invent 2023 – Smart savings: Amazon EC2 cost-optimization strategies](#)
- [AWS re:Invent 2021 – Powering next-gen Amazon EC2: Deep dive on the Nitro System](#)
- [AWS re:Invent 2019 – Amazon EC2 foundations](#)

Zugehörige Beispiele:

- [Beispiele für Amazon EC2 Auto Scaling-Gruppe](#)
- [Workshop zu Amazon EKS](#)
- [Skalieren von Amazon-EKS-Workloads durch die Ausführung auf IPv6](#)

PERF02-BP06 Verwenden von optimierten hardwarebasierten Datenverarbeitungsbeschleunigern

Verwenden Sie Hardwarebeschleuniger, um bestimmte Funktionen effizienter auszuführen als CPU-basierte Alternativen.

Typische Anti-Muster:

- Sie haben in der Workload keine Benchmark einer Allzweck-Instance verglichen mit einer speziell entwickelten Instance durchgeführt, die eine höhere Leistung und niedrigere Kosten bieten kann.
- Sie verwenden hardwarebasierte Datenverarbeitungsbeschleuniger für Aufgaben, die mithilfe von CPU-basierten Alternativen effizienter sein können.
- Sie überwachen die GPU-Nutzung nicht.

Vorteile der Nutzung dieser bewährten Methode: Durch die Verwendung hardwarebasierter Beschleuniger wie Grafikprozessoren (GPUs) und Field-Programmable Gate Arrays (FPGAs) können Sie bestimmte Verarbeitungsfunktionen effizienter ausführen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Beschleunigte Computing-Instances bieten Zugriff auf hardwarebasierte Datenverarbeitungsbeschleuniger wie GPUs und FPGAs. Diese Hardwarebeschleuniger führen bestimmte Funktionen wie die Grafikverarbeitung oder Datenmusterzuordnung effizienter aus als CPU-basierte Alternativen. Viele beschleunigte Workloads, wie Rendering, Transkodierung und Machine Learning, sind sehr variabel im Bezug auf die Ressourcennutzung. Betreiben Sie diese Hardware nur so lange wie nötig und nehmen Sie sie automatisch außer Betrieb, wenn sie nicht mehr benötigt wird, um die allgemeine Leistungseffizienz zu verbessern.

Implementierungsschritte

- Ermitteln Sie, welche [beschleunigten Computing-Instances](#) für Ihre Anforderungen geeignet sind.
- Nutzen Sie für Machine-Learning-Workloads spezielle Hardware, die auf Ihre Workload abgestimmt ist, z. B. [AWS Trainium](#), [AWS Inferentia](#) und [Amazon EC2 DL1](#). AWS Inferentia-Instances wie Inf2-Instances [bieten eine um bis zu 50 % bessere Leistung pro Watt als vergleichbare Amazon-EC2-Instances](#).
- Erfassen Sie Nutzungsmetriken für Ihre beschleunigten Computing-Instances. Sie können den CloudWatch-Agent beispielsweise verwenden, um Metriken wie `utilization_gpu` und `utilization_memory` für Ihre GPUs zu sammeln, wie im [Artikel zum Erfassen von NVIDIA-GPU-Metriken mit Amazon CloudWatch](#) beschrieben.
- Optimieren Sie Code, Netzwerkbetrieb und die Einstellungen von Hardwarebeschleunigern, um sicherzustellen, dass die zugrunde liegende Hardware optimal genutzt wird.
 - [Optimieren der GPU-Einstellungen](#)
 - [GPU-Überwachung und -Optimierung im Deep-Learning-AMI](#)
 - [Optimieren von E/A für die GPU-Leistungsoptimierung von Deep-Learning-Training in Amazon SageMaker AI](#)
- Verwenden Sie die aktuellen leistungsstarken Bibliotheken und GPU-Treiber.
- Automatisieren Sie die Freigabe nicht genutzter GPU-Instances.

Ressourcen

Zugehörige Dokumente:

- [Working with GPUs on Amazon Elastic Container Service](#)
- [GPU-Instances](#)

- [Instances mit AWS Trainium](#)
- [Instances mit AWS Inferentia](#)
- [Let's Architect! Erstellen von Architekturen mit benutzerdefinierten Chips und Beschleunigern](#)
- [Beschleunigte Datenverarbeitung](#)
- [Amazon EC2 VT1 Instances](#)
- [How do I choose the appropriate Amazon EC2 instance type for my workload?](#)
- [Choose the best AI accelerator and model compilation for computer vision inference with Amazon SageMaker AI](#)

Zugehörige Videos:

- AWS re:Invent 2021 – [How to select Amazon Elastic Compute Cloud GPU instances for deep learning](#)
- [AWS re:Invent 2022 – \[NEW LAUNCH!\] Introducing AWS Inferentia2-based Amazon EC2 Inf2 instances](#)
- [AWS re:Invent 2022 – Accelerate deep learning and innovate faster with AWS Trainium](#)
- [AWS re:Invent 2022 – Deep learning on AWS with NVIDIA: From training to deployment](#)

Zugehörige Beispiele:

- [Amazon SageMaker AI und NVIDIA GPU Cloud \(NGC\)](#)
- [Verwendung von SageMaker AI mit Trainium und Inferentia für optimierte Workloads für Deep-Learning-Training und Inferenzierung](#)
- [Optimierung von NLP-Modellen mit Instances für Amazon Elastic Compute Cloud Inf1 in Amazon SageMaker AI](#)

Datenverwaltung

Fragen

- [LEIST 3. Wie speichern und verwalten Sie die Daten in Ihrer Workload und wie greifen Sie darauf zu?](#)

LEIST 3. Wie speichern und verwalten Sie die Daten in Ihrer Workload und wie greifen Sie darauf zu?

Die optimale Datenverwaltungslösung für ein bestimmtes System hängt vom Datentyp (Block, Datei oder Objekt), den Zugriffsmustern (zufällig oder sequenziell), dem erforderlichen Durchsatz, der Zugriffshäufigkeit (online, offline, Archiv), der Aktualisierungshäufigkeit (WORM, dynamisch) sowie den Verfügbarkeits- und Lebensdauieranforderungen ab. Well-Architected-Workloads verwenden zweckgebundene Datenspeicher, die verschiedene Features zur Verbesserung der Leistung ermöglichen.

Best Practices

- [PERF03-BP01 Verwenden eines speziell entwickelten Datenspeichers, der die Datenzugriffs- und Speicheranforderungen am besten unterstützt](#)
- [PERF03-BP02 Bewerten verfügbarer Konfigurationsoptionen für den Datenspeicher](#)
- [PERF03-BP03 Erfassen und Aufzeichnen von Metriken zur Datenspeicherleistung](#)
- [PERF03-BP04 Implementieren von Strategien zur Verbesserung der Abfrageleistung im Datenspeicher](#)
- [PERF03-BP05 Implementieren von Datenzugriffsmustern, die Caching nutzen](#)

PERF03-BP01 Verwenden eines speziell entwickelten Datenspeichers, der die Datenzugriffs- und Speicheranforderungen am besten unterstützt

Machen Sie sich mit Datenmerkmalen (wie Freigabe, Größe, Cache-Größe, Zugriffsmuster, Latenz, Durchsatz und Persistenz von Daten) vertraut, um die richtigen, speziell entwickelten Datenspeicher (Speicher oder Datenbank) für die Workload auszuwählen.

Typische Anti-Muster:

- Sie halten an einem Datenspeicher fest, da es interne Erfahrungen und Wissen über eine bestimmte Datenbanklösung gibt.
- Sie gehen davon aus, dass für alle Workloads ähnliche Datenspeicher- und Zugriffsanforderungen gelten.
- Sie haben keinen Datenkatalog zur Inventarisierung Ihrer Datenbestände eingeführt.

Vorteile der Nutzung dieser bewährten Methode: Wenn Sie die Datenmerkmale und -anforderungen verstehen, können Sie die effizienteste und leistungsfähigste Speichertechnologie ermitteln, die für Ihre Workload-Anforderungen geeignet ist.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Stellen Sie bei der Auswahl und Implementierung von Datenspeicher sicher, dass die Abfrage-, Skalierungs- und Speichermerkmale die Workload-Datenanforderungen unterstützen. AWS bietet zahlreiche Datenspeicher- und Datenbanktechnologien, darunter Blockspeicher, Objektspeicher, Streaming-Speicher, Dateisystem-, relationale, Schlüsselwert-, Dokument-, In-Memory-, Graph-, Zeitreihen- und Ledger-Datenbanken. Jede Datenverwaltungslösung hat verfügbare Optionen und Konfigurationen, um Ihre Anwendungsfälle und Datenmodelle zu unterstützen. Wenn Sie die Merkmale und Anforderungen der Daten verstehen, können Sie sich von monolithischer Speichertechnologie und restriktiven Einheitsansätzen lösen und sich auf eine angemessene Datenverwaltung konzentrieren.

Implementierungsschritte

- Führen Sie eine Bestandsaufnahme der verschiedenen Datentypen durch, die in Ihrer Workload vorhanden sind.
- Verstehen und dokumentieren Sie Datenmerkmale und -anforderungen, einschließlich:
 - Datentyp (strukturiert, semistrukturiert, relational)
 - Datenvolumen und -wachstum
 - Lebensdauer von Daten: anhaltend, flüchtig, vorübergehend
 - Anforderungen an AKID (Atomarität, Konsistenz, Isolation, Dauerhaftigkeit)
 - Datenzugriffsmuster (leseintensiv oder schreibintensiv)
 - Latenz
 - Durchsatz
 - IOPS (Eingabe-/Ausgabevorgänge pro Sekunde)
 - Aufbewahrungszeitraum
- Erfahren Sie mehr über die verschiedenen Datenspeicher ([Speicher](#)- und [Datenbank](#)-Services), die für Ihre Workload in AWS verfügbar sind und Ihre Datenmerkmale erfüllen können (wie beschrieben unter [PERF01-BP01 Informieren über verfügbare Cloud-Services und -Features](#)). Einige Beispiele für AWS-Speichertechnologien und ihre Schlüsselmerkmale sind:

Typ	AWS-Services	Schlüsselmerkmale
Objektspeicher	Amazon S3	Unbegrenzte Skalierbarkeit, hohe Verfügbarkeit und mehrere Optionen für Barrierefreiheit. Für die Übertragung von Objekten in und aus Amazon S3 und den Zugriff auf diese Objekte können Sie einen Service wie z. B. Transfer Acceleration oder Zugangspunkte verwenden, um Ihren Standort, Ihre Sicherheitsanforderungen und Zugriffsmuster zu unterstützen.
Archivieren von Speichern	Amazon Glacier	Für die Datenarchivierung entwickelt.
Streaming-Speicher	Amazon Kinesis Amazon Managed Streaming für Apache Kafka (Amazon MSK)	Effiziente Erfassung und Speicherung von Streaming-Daten.
Gemeinsames Dateisystem	Amazon Elastic File System (Amazon EFS)	Bereitstellbares Dateisystem, auf das mehrere Arten von Datenverarbeitungslösungen zugreifen können.

Typ	AWS-Services	Schlüsselmerkmale
Gemeinsames Dateisystem	Amazon FSx	Basiert auf den aktuellen AWS-Datenverarbeitungslösungen und unterstützt vier gängige Dateisysteme: NetApp ONTAP, OpenZFS, Windows File Server und Lustre. Die Latenz, der Durchsatz und die IOPS von Amazon FSx variieren je nach Dateisystem und sollten bei der Auswahl des richtigen Dateisystems für Ihre Workload-Anforderungen berücksichtigt werden.
Blockspeicher	Amazon Elastic Block Store (Amazon EBS)	Skalierbarer, hochleistungsfähiger Blockspeicherservice für Amazon Elastic Compute Cloud (Amazon EC2). Amazon EBS umfasst SSD-gestützten Speicher für transaktions- und IOPS-intensive Workloads und HDD-gestützten Speicher für durchsatzintensive Workloads.

Typ	AWS-Services	Schlüsselmerkmale
Relationale Datenbank	Amazon Aurora , Amazon RDS , Amazon Redshift .	Sie unterstützt AKID-Transaktionen (Atomarität, Konsistenz, Isolation und Dauerhaftigkeit) und gewährleistet die referentielle Integrität sowie eine starke Datenkonsistenz. Bei zahlreichen herkömmlichen Anwendungen, Enterprise Resource Planning (ERP), Customer Relationship Management (CRM) und E-Commerce werden relationale Datenbanken zum Speichern der Daten verwendet.
Schlüssel-Werte-Datenbank	Amazon DynamoDB	Für gängige Zugriffsmuster optimiert, üblicherweise zum Speichern und Abrufen großer Datenmengen. Web-Apps mit hohem Datenverkehr, E-Commerce-Systeme und Gaming-Anwendungen sind typische Anwendungsfälle für Schlüssel-Werte-Datenbanken.

Typ	AWS-Services	Schlüsselmerkmale
Dokumentdatenbank	Amazon DocumentDB	Dies wurde entwickelt, um halbstrukturierte Daten als JSON-ähnliche Dokumente zu speichern. Mit diesen Datenbanken können Entwickler Anwendungen wie Content Management, Kataloge und Benutzerprofile schnell erstellen und aktualisieren.
In-Memory-Datenbanken	Amazon ElastiCache , Amazon MemoryDB für Redis	Dies wird für Anwendungen eingesetzt, die einen Echtzeitzugriff auf Daten, die niedrigste Latenz und den höchsten Durchsatz erfordern. Sie können In-Memory-Datenbanken für Anwendungs-Caching , Sitzungsmanagement, Gaming-Bestenlisten, ML-Feature-Store mit niedriger Latenz, Microservices-Messaging-System und einen Streaming-Mechanismus mit hohem Durchsatz verwenden

Typ	AWS-Services	Schlüsselmerkmale
Graphdatenbank	Amazon Neptune	Dies ist für Anwendungen gedacht, die in Millionen von Beziehungen zwischen hochgradig vernetzten Diagrammdatensätzen mit Millisekunden-Latenz navigieren und diese abfragen müssen. Viele Unternehmen verwenden Graphdatenbanken für Betrugserkennung, soziale Netzwerke und Empfehlungs-Engines.
Zeitreihendatenbank	Amazon Timestream	Diese erfasst, generiert und gewinnt auf effiziente Weise Einblicke aus Daten, die sich im Laufe der Zeit ändern. IoT-Anwendungen, DevOps und industrielle Telemetrie können Zeitreihendatenbanken nutzen.

Typ	AWS-Services	Schlüsselmerkmale
Wide-Column-Datenbanken	Amazon Keyspaces (für Apache Cassandra)	Es werden Tabellen, Zeilen und Spalten verwendet, aber im Gegensatz zu einer relationalen Datenbank können sich die Namen und das Format der Spalten von Zeile zu Zeile in derselben Tabelle unterscheiden. In der Regel werden Wide Column-Speicher in umfangreichen Branchen-Apps für Gerätewartung, Flottenverwaltung und Routenoptimierung eingesetzt.
Ledger	Amazon Quantum Ledger Database (Amazon QLDB)	Dies bietet eine zentrale und vertrauenswürdige Instanz für die Verwaltung einer skalierbaren, unveränderlichen und kryptografisch überprüfbaren Aufzeichnung von Transaktionen für jede Anwendung. Ledger-Datenbanken werden für Datensatzsysteme, Lieferketten, Registrierungen und sogar Banktransaktionen verwendet.

- Wenn Sie eine Datenplattform aufbauen, nutzen Sie eine [moderne Datenarchitektur](#) in AWS, um Ihren Data Lake, Ihr Data Warehouse und Ihre speziell entwickelten Datenspeicher zu integrieren.
- Die wichtigsten Fragen, die Sie bei der Auswahl eines Datenspeichers für Ihre Workload berücksichtigen müssen, lauten wie folgt:

Frage	Worauf Sie achten sollten
Wie sind die Daten strukturiert?	• Wenn die Daten nicht strukturiert sind, erwägen Sie einen Objektspeicher wie Amazon S3 oder eine NoSQL-Datenbank wie Amazon DocumentDB. • Erwägen Sie für Schlüssel-Werte-Daten DynamoDB, Amazon ElastiCache (Redis OSS) oder Amazon MemoryDB.
Welches Maß an referentieller Integrität ist erforderlich?	• Bei Fremdschlüsseinschränkungen können relationale Datenbanken wie Amazon RDS und Aurora diese Integritäts Ebene bieten. • Üblicherweise würden Sie innerhalb eines NoSQL-Datenmodells Ihre Daten in ein einzelnes Dokument oder eine Sammlung von Dokumenten denormalisieren, die in einer einzelnen Anfrage abgerufen werden können, anstatt Daten in Dokumenten oder Tabellen zusammenzufügen.
Ist AKID-Compliance (Atomarität, Konsistenz, Isolation, Dauerhaftigkeit) erforderlich?	• Wenn mit relationalen Datenbanken zusammenhängende AKID-Eigenschaften erforderlich sind, erwägen Sie eine relationale Datenbank wie Amazon RDS und Aurora. • Wenn eine starke Konsistenz für eine NoSQL-Datenbank erforderlich ist, können Sie strikt konsistente Lesevorgänge mit DynamoDB verwenden.

Frage	Worauf Sie achten sollten
Wie ändern sich die Speicheranforderungen im Laufe der Zeit? Wie beeinflusst dies die Skalierbarkeit?	• Serverless-Datenbanken wie DynamoDB und Amazon Quantum Ledger Database (Amazon QLDB) skalieren dynamisch. • Relationale Datenbanken haben oftmals Obergrenzen bei bereitgestelltem Speicher und müssen mithilfe von Mechanismen wie Sharding horizontal partitioniert werden, sobald sie diese Grenzen erreicht haben.
Wie hoch ist der Anteil der Leseabfragen im Verhältnis zu den Schreibabfragen? Könnte Caching die Leistung verbessern?	• Leseintensive Workloads könnten von einer Caching-Ebene, wie ElastiCache oder DAX profitieren, wenn es sich bei der Datenbank um DynamoDB handelt. • Lesevorgänge können auch zu Read Replicas mit relationalen Datenbanken ausgelagert werden, wie Amazon RDS.
Wird Speicher und Modifizierung (OLTP – Online Transaction Processing) oder Abruf und Berichterstattung (OLAP – Online Analytical Processing) eine höhere Priorität eingeräumt?	• Erwägen Sie für Read-as-is-Transaktionsverarbeitung mit hohem Durchsatz eine NoSQL-Datenbank wie DynamoDB. • Verwenden Sie Amazon RDS für hohen Durchsatz und komplexe Lesemuster (wie Join) mit Konsistenz. • Erwägen Sie für analytische Abfragen eine spaltenbasierte Datenbank wie Amazon Redshift oder das Exportieren von Daten zu Amazon S3 und das Durchführen von Analytik mithilfe von Athena oder Amazon Quick.

Frage	Worauf Sie achten sollten
Welches Ausmaß an Stabilität erfordern die Daten?	• Aurora repliziert Ihre Daten automatisch in drei Availability Zones innerhalb einer Region, was bedeutet, dass Ihre Daten hochbeständig sind und eine geringere Wahrscheinlichkeit von Datenverlust besteht. • DynamoDB wird automatisch in mehreren Availability Zones repliziert und bietet hohe Verfügbarkeit und Datenstabilität. • Amazon S3 bietet eine Langlebigkeit mit 11 Neunen. Viele Datenbankservices wie Amazon RDS und DynamoDB unterstützen das Exportieren von Daten zu Amazon S3 für Langzeitaufbewahrung und Archivierung.
Besteht der Wunsch, sich von kommerziellen Datenbank-Engines oder Lizenzkosten zu entfernen?	• Ziehen Sie Open-Source-Engines wie PostgreSQL und MySQL auf Amazon RDS oder Aurora in Erwägung. • Nutzen Sie AWS Database Migration Service und AWS Schema Conversion Tool zum Migrieren von kommerziellen Datenbank-Engines zu Open Source-Lösungen.
Was ist die Betriebserwartung an die Datenbank? Ist der Umstieg zu verwalteten Services eine Priorität?	• Das Verwenden von Amazon RDS anstatt von Amazon EC2 und DynamoDB oder Amazon DocumentDB anstatt eine NoSQL-Datenbank selbst zu hosten, kann den Betriebsaufwand verringern.

Frage	Worauf Sie achten sollten
Wie erfolgt derzeit der Zugriff auf die Datenbank? Handelt es sich nur um einen Anwendungszugriff oder gibt es Business-Intelligence (BI)-Benutzer und andere Standardanwendungen?	• Wenn Sie von externen Tools abhängig sind, müssen Sie möglicherweise mit der Datenbank, die unterstützt wird, die Kompatibilität aufrecht erhalten. Amazon RDS ist vollständig kompatibel mit den unterschiedlichen Engine-Versionen, die unterstützt werden, einschließlich Microsoft SQL Server, Oracle, MySQL und PostgreSQL.

- Führen Sie Experimente und Benchmarking in einer Nicht-Produktionsumgebung durch, um herauszufinden, welcher Datenspeicher Ihre Workload-Anforderungen erfüllen kann.

Ressourcen

Zugehörige Dokumente:

- [Typen von Amazon-EBS-Volumes](#)
- [Amazon-EC2-Speicher](#)
- [Amazon EFS: Amazon EFS-Performance](#)
- [Leistung von Amazon FSx für Lustre](#)
- [Leistung von Amazon FSx für Windows File Server](#)
- [Amazon Glacier: Glacier-Dokumentation](#)
- [Amazon S3: Überlegungen zu Anfragerate und Leistung](#)
- [Cloud-Speicher mit AWS](#)
- [Amazon EBS I/O Characteristics](#)
- [Cloud-Datenbanken mit AWS](#)
- [AWS Datenbank-Caching](#)
- [DynamoDB Accelerator](#)
- [Bewährte Methoden mit Amazon Aurora](#)
- [Leistung von Amazon Redshift](#)
- [Die zehn besten Leistungstipps für Amazon Athena](#)

- [Bewährte Methoden für Amazon Redshift Spectrum](#)
- [Bewährte Methoden für Amazon DynamoDB](#)
- [Choose between Amazon EC2 and Amazon RDS](#)
- [Bewährte Methoden für die Implementierung von Amazon ElastiCache](#)

Zugehörige Videos:

- [AWS re:Invent 2023: Improve Amazon Elastic Block Store efficiency and be more cost-efficient](#)
- [AWS re:Invent 2023: Optimizing storage price and performance with Amazon Simple Storage Service](#)
- [AWS re:Invent 2023: Building and optimizing a data lake on Amazon Simple Storage Service](#)
- [AWS re:Invent 2022: Building modern data architectures on AWS](#)
- [AWS re:Invent 2022: Building data mesh architectures on AWS](#)
- [AWS re:Invent 2023: Deep dive into Amazon Aurora and its innovations](#)
- [AWS re:Invent 2023: Advanced data modeling with Amazon DynamoDB](#)
- [AWS re:Invent 2022: Modernize apps with purpose-built databases](#)
- [Ausführliche Beschreibung von Amazon DynamoDB: Erweiterte Entwurfsmuster](#)

Zugehörige Beispiele:

- [AWS Workshop „Speziell entwickelte Datenbanken](#)
- [Databases for Developers](#)
- [AWS Modern Data Architecture Immersion Day](#)
- [Build a Data Mesh on AWS](#)
- [Amazon-S3-Beispiele](#)
- [Optimierung von Datenmustern mithilfe von Amazon Redshift Data Sharing](#)
- [Datenbankmigrationen](#)
- [MS SQL Server – AWS Database Migration Service \(AWS DMS\)-Replikationsdemo](#)
- [Praktischer Workshop zur Datenbankmodernisierung](#)
- [Beispiele zu Amazon Neptune](#)

PERF03-BP02 Bewerten verfügbarer Konfigurationsoptionen für den Datenspeicher

Machen Sie sich mit den verschiedenen Features und Konfigurationsoptionen vertraut, die für Ihre Datenspeicher verfügbar sind, und bewerten Sie sie, um Speicherplatz und Leistung für Ihre Workload zu optimieren.

Typische Anti-Muster:

- Sie verwenden nur einen Speichertyp, z. B. Amazon EBS, für alle Workloads.
- Sie verwenden bereitgestellte IOPS für alle Workloads, ohne reale Tests auf allen Speicherebenen durchzuführen.
- Ihnen fehlt das Bewusstsein für die Wahl der Konfigurationsoptionen der Datenverwaltungslösung.
- Sie verlassen sich ausschließlich auf das Vergrößern der Instance-Größe, ohne andere verfügbare Konfigurationsoptionen in Betracht zu ziehen.
- Sie testen die Skalierungsoptionen Ihres Datenspeichers nicht.

Vorteile der Nutzung dieser bewährten Methode: Indem Sie Datenspeicherkonfigurationen erkunden und mit ihnen experimentieren, können Sie möglicherweise Infrastrukturkosten senken, die Leistung verbessern und den Aufwand zur Verwaltung Ihrer Workloads verringern.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Für eine Workload können je nach Datenspeicher- und Zugriffsanforderungen ein oder mehrere Datenspeicher verwendet werden. Zur Optimierung der Leistungseffizienz und Kosten müssen Sie Datenzugriffsmuster auswerten, um die entsprechenden Datenspeicherkonfigurationen zu bestimmen. Während Sie die Datenspeicheroptionen erkunden, sollten Sie unterschiedliche Aspekte in Betracht ziehen. Dazu zählen Speicheroptionen, Arbeitsspeicher, Rechenvorgänge, Read Replica, Konsistenzanforderungen, Verbindungs-Pooling und Caching-Optionen. Experimentieren Sie mit diesen unterschiedlichen Konfigurationsoptionen, um Metriken zur Leistungseffizienz zu verbessern.

Implementierungsschritte

- Verstehen Sie die aktuellen Konfigurationen (wie Instance-Typ, Speichergröße oder Version der Datenbank-Engine) des Datenspeichers.
- Lesen Sie die AWS-Dokumentation und die bewährten Methoden, um mehr über empfohlene Konfigurationsoptionen zu erfahren, mit denen Sie die Leistung für den Datenspeicher verbessern

können. Die wichtigsten Datenspeicheroptionen, die Sie in Betracht ziehen sollten, sind die folgenden:

Konfigurationsoption	Beispiele
Auslagern von Lesevorgängen (wie Read Replicas und Caching)	• Bei DynamoDB-Tabellen können Sie Lesevorgänge mithilfe von DAX für Caching auslagern. • Sie können einen Cluster für Amazon ElastiCache (Redis OSS) erstellen und Ihre Anwendung so konfigurieren, dass sie zuerst aus dem Cache liest und dann auf die Datenbank zurückfällt, wenn das angeforderte Element nicht vorhanden ist. • Relationale Datenbanken wie Amazon RDS und Aurora sowie bereitgestellte NoSQL-Datenbanken wie Neptune und Amazon DocumentDB unterstützen alle das Hinzufügen von Read Replicas, um die Lesevorgänge der Workload auszulagern. • Serverless-Datenbanken wie DynamoDB skalieren automatisch. Stellen Sie sicher, dass Sie ausreichend Read Capacity Units (RCU) bereitstellen, um die Workload zu verarbeiten.

Konfigurationsoption	Beispiele
Skalieren von Schreibvorgängen (wie Partitionsschlüssel-Sharding oder Einführung einer Warteschlange)	• Bei relationalen Datenbanken können Sie die Größe der Instance erhöhen, um eine erhöhte Workload zu bewältigen, oder die bereitgestellten IOPs erhöhen, um einen erhöhten Durchsatz in den zugrunde liegenden Speicher zu ermöglichen. • Sie können vor Ihrer Datenbank auch eine Warteschlange einrichten, anstatt direkt in die Datenbank zu schreiben. Mithilfe dieses Musters können Sie die Datenerfassung von der Datenbank entkoppeln und die Flow-Rate steuern, sodass die Datenbank nicht überwältigt wird. • Das Batching Ihrer Schreibanforderungen, anstatt mehrere kurzlebige Transaktionen zu erstellen, kann Ihnen dabei helfen, den Durchsatz bei relationalen Datenbanken mit hohem Schreibvolumen zu verbessern. • Serverless-Datenbanken wie DynamoDB können den Schreibdurchsatz automatisch skalieren oder indem die bereitgestellten Kapazitätseinheiten für Schreibvorgänge (Write Capacity Units, WCU) abhängig vom Kapazitätsmodus angepasst werden. • Es können immer noch Probleme mit heißen Partitionen auftreten, wenn Sie die Durchsatzgrenzen für einen bestimmten Partitionsschlüssel erreichen. Dies kann verhindert werden, indem Sie einen Partitionsschlüssel auswählen, der gleichmäßiger verteilt ist, oder indem Sie die Schreibvorgänge des Partitionsschlüssels in Shards aufteilen.

Konfigurationsoption	Beispiele
Richtlinien zum Verwalten des Lebenszyklus von Datensätzen	• Mit Amazon-S3-Lebenszyklen können Sie Ihre Objekte während ihres gesamten Lebenszyklus verwalten. Wenn die Zugriffsmuster unbekannt oder nicht prognostizierbar sind oder sich ändern, können Sie Amazon S3 Intelligent-Tiering verwenden. Hiermit werden Zugriffsmuster überwacht und Objekte, auf die nicht zugegriffen wurde, automatisch in kostengünstigere Zugriffsebenen verschoben. Anhand der Metriken von Amazon S3 Storage Lens können Sie Optimierungsmöglichkeiten und Lücken im Lebenszyklusmanagement ermitteln. • Das Amazon-EFS-Lebenszyklusmanagement verwaltet den Dateispeicher für Ihre Dateisysteme automatisch.
Verbindungsmanagement und Pooling	• Amazon-RDS-Proxy kann mit Amazon RDS und Aurora verwendet werden, um Verbindungen mit der Datenbank zu verwalten. • Serverless-Datenbanken wie DynamoDB haben keine ihnen zugewiesenen Verbindungen, aber ziehen Sie die bereitgestellte Kapazität sowie automatische Skalierungsrichtlinien in Betracht, um Datenverkehrsspitzen zu bewältigen.

- Führen Sie Experimente und Benchmarking in einer Nicht-Produktionsumgebung durch, um herauszufinden, welche Konfigurationsoption Ihre Workload-Anforderungen erfüllen kann.
- Nachdem Sie experimentiert haben, planen Sie die Migration und überprüfen Sie die Leistungsmetriken.

- Verwenden Sie AWS-Tools zur Überwachung (wie [Amazon CloudWatch](#)) und Optimierung (wie [Amazon S3 Storage Lens](#)), um den Datenspeicher kontinuierlich anhand realer Nutzungsmuster zu optimieren.

Ressourcen

Zugehörige Dokumente:

- [Cloud-Speicher mit AWS](#)
- [Typen von Amazon-EBS-Volumes](#)
- [Amazon-EC2-Speicher](#)
- [Amazon EFS: Amazon EFS-Performance](#)
- [Leistung von Amazon FSx für Lustre](#)
- [Leistung von Amazon FSx für Windows File Server](#)
- [Amazon Glacier: Glacier-Dokumentation](#)
- [Amazon S3: Überlegungen zu Anfragerate und Leistung](#)
- [Amazon EBS I/O Characteristics](#)
- [Cloud-Datenbanken mit AWS](#)
- [AWS-Datenbank-Caching](#)
- [DynamoDB Accelerator](#)
- [Bewährte Methoden mit Amazon Aurora](#)
- [Leistung von Amazon Redshift](#)
- [Die zehn besten Leistungstipps für Amazon Athena](#)
- [Bewährte Methoden für Amazon Redshift Spectrum](#)
- [Bewährte Methoden für Amazon DynamoDB](#)

Zugehörige Videos:

- [AWS re:Invent 2023: Improve Amazon Elastic Block Store efficiency and be more cost-efficient](#)
- [AWS re:Invent 2023: Optimize storage price and performance with Amazon Simple Storage Service](#)
- [AWS re:Invent 2023: Building and optimizing a data lake on Amazon Simple Storage Service](#)
- [AWS re:Invent 2023: What's new with AWS file storage](#)

- [AWS re:Invent 2023: Dive deep into Amazon DynamoDB](#)

Zugehörige Beispiele:

- [AWS-Workshop „Speziell entwickelte Datenbanken“](#)
- [Databases for Developers](#)
- [AWS Modern Data Architecture Immersion Day](#)
- [Amazon EBS Autoscale](#)
- [Amazon-S3-Beispiele](#)
- [Amazon DynamoDB-Beispiele](#)
- [Beispiele von AWS-Datenbankmigration](#)
- [Workshop zur Modernisierung von Datenbanken](#)
- [Arbeiten mit Parametern auf Ihrem Amazon RDS für Postgress DB](#)

PERF03-BP03 Erfassen und Aufzeichnen von Metriken zur Datenspeicherleistung

Verfolgen und zeichnen Sie relevante Leistungsmetriken für Ihren Datenspeicher auf, um zu verstehen, wie Ihre Datenverwaltungslösungen funktionieren. Mithilfe dieser Metriken können Sie Ihren Datenspeicher optimieren, überprüfen, ob Ihre Workload-Anforderungen erfüllt werden, und sich einen klaren Überblick über die Workload-Leistung verschaffen.

Typische Anti-Muster:

- Sie suchen ausschließlich manuell mithilfe von Protokolldateien nach Metriken.
- Sie veröffentlichen Metriken nur in internen Tools, die von Ihrem Team verwendet werden, und Sie haben kein umfassendes Bild Ihrer Workload.
- Sie verwenden nur die Standardmetriken, die von der Überwachungssoftware Ihrer Wahl aufgezeichnet wurden.
- Sie überprüfen Metriken nur dann, wenn ein Problem vorliegt.
- Sie überwachen Metriken nur auf Systemebene und erfassen keine Datenzugriffs- und Nutzungsmetriken.

Vorteile der Nutzung dieser bewährten Methode: Das Einrichten einer Leistungsbasislinie hilft Ihnen dabei, das normale Verhalten und die Anforderungen von Workloads zu verstehen. Abnorme

Muster können schneller identifiziert und behoben werden, was die Leistung und Zuverlässigkeit des Datenspeichers erhöht.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Um die Leistung der Datenspeicher zu überwachen, müssen Sie mehrere Leistungsmetriken über einen bestimmten Zeitraum aufzeichnen. Auf diese Weise können Sie Anomalien erkennen und die Leistung anhand von Geschäftsmetriken messen, um sicherzustellen, dass Sie die Anforderungen Ihrer Workload erfüllen.

Metriken sollten das zugrunde liegende System, das den Datenspeicher unterstützt, sowie die Datenbankmetriken enthalten. Die Metriken des zugrunde liegenden Systems können die CPU-Auslastung, den Arbeitsspeicher, den verfügbaren Festplattenspeicher, Festplatten-E/A, das Cache-Trefferverhältnis und Metriken zum eingehenden und ausgehenden Netzwerkdatenverkehr umfassen, während die Datenspeichermetriken die Transaktionen pro Sekunde, die häufigsten Abfragen, die durchschnittlichen Abfrageraten, Antwortzeiten, die Indexauslastung, Tabellenschlösser, Abfragezeitüberschreitungen und die Anzahl offener Verbindungen enthält. Diese Daten sind von entscheidender Bedeutung, um festzustellen, wie leistungsfähig die Workload ist und wie die Datenverwaltungslösung genutzt wird. Nutzen Sie diese Metriken im Rahmen eines datengestützten Ansatzes, der Ihnen die Feinabstimmung und Optimierung der von der Workload genutzten Ressourcen ermöglicht.

Nutzen Sie Tools, Bibliotheken und Systeme zum Aufzeichnen von Messungen zur Datenbankleistung.

Implementierungsschritte

- Identifizieren Sie die wichtigsten Leistungsmetriken, die der Datenspeicher verfolgen soll.
 - [Metriken und Dimensionen von Amazon S3](#)
 - [Überwachen von Metriken in einer Amazon-RDS-Instance](#)
 - [Überwachen der DB-Last mit Performance Insights in Amazon RDS](#)
 - [Überblick über „Erweiterte Überwachung“](#)
 - [DynamoDB-Metriken und -Dimensionen](#)
 - [Überwachen von DynamoDB Accelerator](#)
 - [Überwachen von Amazon MemoryDB mit Amazon CloudWatch](#)
 - [Welche Metriken sollte ich überwachen?](#)

- [Überwachen der Amazon-Redshift-Cluster-Leistung](#)
- [Timestream-Metriken und Dimensionen](#)
- [Amazon-CloudWatch-Metriken für Amazon Aurora](#)
- [Protokollierung und Überwachung in Amazon Keyspaces \(für Apache Cassandra\)](#)
- [Überwachen von Amazon-Neptune-Ressourcen](#)
- Verwenden Sie eine zugelassene Protokollierungs- und Überwachungslösung, um diese Metriken zu erfassen. Mit [Amazon CloudWatch](#) lassen sich Metriken aus sämtlichen Ressourcen Ihrer Architektur erfassen. Sie können auch benutzerdefinierte Metriken erfassen und in Oberflächen-, Geschäfts- oder abgeleiteten Metriken veröffentlichen. Richten Sie mit CloudWatch oder mit Lösungen von Drittanbietern Alarme ein, die auf das Überschreiten von Schwellenwerten hinweisen.
- Prüfen Sie, ob die Datenspeicherüberwachung von einer Machine-Learning-Lösung profitieren kann, die Leistungsanomalien erkennt.
 - [Amazon DevOps Guru für Amazon RDS](#) ermöglicht einen Einblick in Leistungsprobleme und bietet Empfehlungen für Korrekturmaßnahmen.
- Konfigurieren Sie die Datenaufbewahrung in Ihrer Überwachungs- und Protokollierungslösung so, dass sie Ihren Sicherheits- und Betriebszielen entspricht.
 - [Standard-Datenaufbewahrung für CloudWatch-Metriken](#)
 - [Standard-Datenaufbewahrung für CloudWatch-Protokolle](#)

Ressourcen

Zugehörige Dokumente:

- [AWS-Datenbank-Caching](#)
- [Die zehn besten Leistungstipps für Amazon Athena](#)
- [Bewährte Methoden mit Amazon Aurora](#)
- [DynamoDB Accelerator](#)
- [Bewährte Methoden für Amazon DynamoDB](#)
- [Bewährte Methoden für Amazon Redshift Spectrum](#)
- [Leistung von Amazon Redshift](#)
- [Cloud-Datenbanken mit AWS](#)
- [Erkenntnisse zur Amazon-RDS-Leistung](#)

Zugehörige Videos:

- [AWS re:Invent 2022 – Performance monitoring with Amazon RDS and Aurora, featuring Autodesk](#)
- [Database Performance Monitoring and Tuning with Amazon DevOps Guru for Amazon RDS](#)
- [AWS re:Invent 2023 – What's new with AWS file storage](#)
- [AWS re:Invent 2023 – Dive deep into Amazon DynamoDB](#)
- [AWS re:Invent 2023 – Building and optimizing a data lake on Amazon S3](#)
- [AWS re:Invent 2023 – What's new with AWS file storage](#)
- [AWS re:Invent 2023 – Dive deep into Amazon DynamoDB](#)
- [Best Practices for Monitoring Redis Workloads on Amazon ElastiCache](#)

Zugehörige Beispiele:

- [Framework zur AWS-Datensatzerfassung und Sammlung von Metriken](#)
- [Workshop zur Amazon-RDS-Überwachung](#)
- [AWS-Workshop „Speziell entwickelte Datenbanken“](#)

PERF03-BP04 Implementieren von Strategien zur Verbesserung der Abfrageleistung im Datenspeicher

Implementieren Sie Strategien zur Datenoptimierung und Verbesserung der Datenabfrage, um mehr Skalierbarkeit und eine effizientere Leistung für Ihre Workloads zu erzielen.

Typische Anti-Muster:

- Sie partitionieren keine Daten in Ihrem Datenspeicher.
- Sie speichern Daten in nur einem Dateiformat in Ihrem Datenspeicher.
- Sie verwenden keine Indizes in Ihrem Datenspeicher.

Vorteile der Nutzung dieser bewährten Methode: Die Optimierung der Daten- und Abfrageleistung führt zu mehr Effizienz, niedrigeren Kosten und einer verbesserten Benutzererfahrung.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Daten- und Abfrageoptimierung sind wichtige Aspekte der Leistungseffizienz in einem Datenspeicher, da sie sich auf die Leistung und Reaktionsfähigkeit der gesamten Cloud-Workload auswirken. Nicht optimierte Abfragen können zu einem höheren Ressourcenverbrauch und Engpässen führen, wodurch die Gesamteffizienz eines Datenspeichers beeinträchtigt wird.

Die Datenoptimierung umfasst mehrere Techniken, um eine effiziente Datenspeicherung und einen effizienten Datenzugriff zu gewährleisten. Dies trägt auch dazu bei, die Abfrageleistung in einem Datenspeicher zu verbessern. Zu den wichtigsten Strategien gehören Datenpartitionierung, Datenkomprimierung und Datendenormalisierung, mit denen Daten sowohl für die Speicherung als auch für den Zugriff optimiert werden können.

Implementierungsschritte

- Verstehen und analysieren Sie die kritischen Datenabfragen, die in Ihrem Datenspeicher durchgeführt werden.
- Identifizieren Sie die langsamen Abfragen in Ihrem Datenspeicher und verwenden Sie Abfragepläne, um den aktuellen Status zu verstehen.
 - [Analysieren des Abfrageplans in Amazon Redshift](#)
 - [Verwenden von EXPLAIN und EXPLAIN ANALYZE in Athena](#)
- Implementieren Sie Strategien zur Verbesserung der Abfrageleistung. Einige der wichtigsten Strategien sind:
 - Nutzung eines [spaltenbasierten Dateiformats](#) (wie Parquet oder ORC).
 - Komprimieren von Daten im Datenspeicher, um Speicherplatz und E/A-Betrieb zu reduzieren.
 - Datenpartitionierung zur Aufteilung von Daten in kleinere Teile und zur Reduzierung der Zeit für das Scannen von Daten.
 - [Daten in Athena partitionieren](#)
 - [Partitionen und Datenverteilung](#)
 - Datenindizierung für die gemeinsamen Spalten in der Abfrage.
 - Verwenden Sie materialisierte Ansichten für häufige Abfragen.
 - [Materialisierte Ansichten verstehen](#)
 - [Erstellen von materialisierten Ansichten in Amazon Redshift](#)

- Wählen Sie den richtigen Verknüpfungsvorgang für die Abfrage aus. Wenn Sie zwei Tabellen verknüpfen, geben Sie die größere Tabelle auf der linken Seite der Verknüpfung und die kleinere Tabelle auf der rechten Seite der Verknüpfung an.
- Verteilte Caching-Lösung zur Verbesserung der Latenz und zur Reduzierung der Anzahl von Datenbank-E/A-Vorgängen.
- Regelmäßige Wartung wie [Bereinigung](#), Neuindizierung und [Ausführen von Statistiken](#).
- Experimentieren und testen Sie Strategien in einer Nicht-Produktionsumgebung.

Ressourcen

Zugehörige Dokumente:

- [Bewährte Methoden mit Amazon Aurora](#)
- [Leistung von Amazon Redshift](#)
- [Die zehn besten Leistungstipps für Amazon Athena](#)
- [AWS-Datenbank-Caching](#)
- [Bewährte Methoden für die Implementierung von Amazon ElastiCache](#)
- [Daten in Athena partitionieren](#)

Zugehörige Videos:

- [AWS re:Invent 2023 – Bewährte Methoden zur Kostenoptimierung für AWS-Speicher](#)
- [AWS re:Invent 2022 – Performance monitoring with Amazon RDS and Aurora, featuring Autodesk](#)
- [Optimize Amazon Athena Queries with New Query Analysis Tools](#)

Zugehörige Beispiele:

- [AWS Purpose Built Databases Workshop](#)

PERF03-BP05 Implementieren von Datenzugriffsmustern, die Caching nutzen

Implementieren Sie Zugriffsmuster, die vom Daten-Caching profitieren, damit häufig aufgerufene Daten schnell abgerufen werden können.

Typische Anti-Muster:

- Sie speichern Daten, die sich häufig ändern.
- Sie verlassen sich auf zwischengespeicherte Daten, als ob sie dauerhaft gespeichert und immer verfügbar wären.
- Sie berücksichtigen nicht die Konsistenz Ihrer zwischengespeicherten Daten.
- Sie überwachen die Effizienz Ihrer Caching-Implementierung nicht.

Vorteile der Nutzung dieser bewährten Methode: Das Speichern von Daten in einem Cache kann die Leselatenz, den Lesedurchsatz, die Benutzererfahrung und die Gesamteffizienz verbessern sowie die Kosten senken.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Ein Cache ist eine Software- oder Hardwarekomponente zum Speichern von Daten, damit zukünftige Abfragen derselben Daten schneller oder effizienter verarbeitet werden können. Die in einem Cache gespeicherten Daten können bei Verlust rekonstruiert werden, indem eine frühere Berechnung wiederholt wird oder die Daten aus einem anderen Datenspeicher abgerufen werden.

Das Caching von Daten kann eine der effektivsten Strategien sein, um die allgemeine Anwendungsleistung zu verbessern und die Belastung Ihrer zugrunde liegenden primären Datenquellen zu verringern. Daten können auf mehreren Ebenen in der Anwendung zwischengespeichert werden, z. B. innerhalb der Anwendung durch Remote-Aufrufe (so genanntes clientseitiges Caching) oder durch Verwendung eines schnellen sekundären Services zur Speicherung der Daten (so genanntes Remote-Caching).

Clientseitiges Caching

Beim clientseitigen Caching kann jeder Client (eine Anwendung oder ein Service, die bzw. der den Backend-Datenspeicher abfragt) die Ergebnisse seiner eindeutigen Abfragen lokal für einen bestimmten Zeitraum speichern. So kann die Anzahl der Anfragen an einen Datenspeicher im Netzwerk reduziert werden, da zuerst der lokale Client-Cache überprüft wird. Wenn die Ergebnisse nicht vorhanden sind, kann die Anwendung den Datenspeicher abfragen und diese Ergebnisse lokal speichern. Dieses Muster ermöglicht es jedem Client, Daten am nächstgelegenen Ort (dem Client selbst) zu speichern, was zur geringstmöglichen Latenz führt. Clients können auch weiterhin einige Abfragen bearbeiten, wenn der Backend-Datenspeicher nicht verfügbar ist, wodurch die Verfügbarkeit des Gesamtsystems erhöht wird.

Ein Nachteil dieses Ansatzes besteht darin, dass bei Beteiligung mehrerer Clients diese möglicherweise dieselben zwischengespeicherten Daten lokal speichern. Dies führt sowohl zu doppelten Speichervorgängen als auch zu Dateninkonsistenzen zwischen diesen Clients. So kann z. B. ein Client die Ergebnisse einer Abfrage zwischenspeichern und eine Minute später führt ein anderer Client dieselbe Abfrage aus und erhält ein anderes Ergebnis.

Remote-Caching

Um das Problem der doppelten Daten zwischen den Clients zu lösen, kann ein schneller externer Service oder ein Remote-Cache verwendet werden, um die abgefragten Daten zu speichern. Anstatt einen lokalen Datenspeicher zu überprüfen, prüft jeder Client den Remote-Cache, bevor er den Backend-Datenspeicher abfragt. Diese Strategie ermöglicht konsistentere Antworten zwischen den Clients, eine bessere Effizienz der gespeicherten Daten und ein höheres Volumen an zwischengespeicherten Daten, da der Speicherplatz unabhängig von den Clients skaliert wird.

Der Nachteil eines Remote-Caches besteht darin, dass das Gesamtsystem möglicherweise eine höhere Latenz aufweist, da ein zusätzlicher Netzwerk-Hop erforderlich ist, um den Remote-Cache zu überprüfen. Das clientseitige Caching kann in Kombination mit dem Remote-Caching verwendet werden, um ein mehrstufiges Caching zu implementieren und die Latenz zu verbessern.

Implementierungsschritte

- Identifizieren Sie Datenbanken, APIs und Netzwerkservices, die vom Caching profitieren könnten. Services, die hohe Lese-Workloads oder ein hohes Lese-Schreib-Verhältnis aufweisen oder deren Skalierung teuer ist, kommen für das Caching in Frage.
 - [Datenbank-Caching](#)
 - [Aktivieren von API-Caching für verbesserte Reaktionsfähigkeit](#)
- Identifizieren Sie die geeignete Caching-Strategie, die am besten zu Ihrem Zugriffsmuster passt.
 - [Caching-Strategien](#)
 - [AWS-Caching-Lösungen](#)
- Befolgen Sie die [bewährten Methoden für das Caching](#) für Ihren Datenspeicher.
- Konfigurieren Sie eine Cache-Invalidierungsstrategie, z. B. eine Time-to-Live (TTL), für alle Daten, die ein Gleichgewicht zwischen der Aktualität der Daten und der Verringerung der Auslastung des Backend-Datenspeichers herstellt.
- Aktivieren Sie Features wie automatische Verbindungswiederholungen, exponentielles Backoff, clientseitige Timeouts und Verbindungspooling beim Client, sofern verfügbar, um die Leistung und Zuverlässigkeit zu verbessern.

- [Bewährte Methoden: Redis-Clients und Amazon ElastiCache \(Redis OSS\)](#)
- Überwachen Sie die Cache-Trefferrate mit einem Ziel von mindestens 80 %. Niedrigere Werte können auf eine unzureichende Cache-Größe oder ein Zugriffsmuster hinweisen, das nicht vom Caching profitiert.
- [Welche Metriken sollte ich überwachen?](#)
- [Best Practices for Monitoring Redis Workloads on Amazon ElastiCache](#)
- [Monitoring best practices with Amazon ElastiCache \(Redis OSS\) using Amazon CloudWatch](#)
- Implementieren Sie die [Datenreplikation](#), um Lesevorgänge auf mehrere Instances auszulagern und die Leseleistung und Verfügbarkeit von Daten zu verbessern.

Ressourcen

Zugehörige Dokumente:

- [Verwendung von Amazon ElastiCache Well-Architected Lens](#)
- [Monitoring best practices with Amazon ElastiCache \(Redis OSS\) using Amazon CloudWatch](#)
- [Welche Metriken sollte ich überwachen?](#)
- [Whitepaper „Skalierbare Leistung mit Amazon ElastiCache“](#)
- [Caching-Herausforderungen und -Strategien](#)

Zugehörige Videos:

- [Amazon ElastiCache Learning Path](#)
- [Bewährte Methoden für erfolgreiches Design in Amazon ElastiCache](#)
- [AWS re:Invent 2020 – Bewährte Methoden für erfolgreiches Design in Amazon ElastiCache](#)
- [AWS re:Invent 2023 – \[LAUNCH\] Introducing Amazon ElastiCache Serverless](#)
- [AWS re:Invent 2022 – 5 hervorragende Methoden, um die Datenebene mit Redis neu zu gestalten](#)
- [AWS re:Invent 2021 – Deep dive on Amazon ElastiCache \(Redis OSS\)](#)

Zugehörige Beispiele:

- [Steigerung der MySQL-Datenbankleistung mit Amazon ElastiCache \(Redis OSS\)](#)

Netzwerk und Bereitstellung von Inhalten

Fragen

- [LEIST 4. Wie wählen und konfigurieren Sie Netzwerkressourcen in Ihrem Workload?](#)

LEIST 4. Wie wählen und konfigurieren Sie Netzwerkressourcen in Ihrem Workload?

Welche Netzwerklösung für eine Workload optimal ist, richtet sich nach der Latenz, dem erforderlichen Durchsatz, dem Jitter und der Bandbreite. Die Standortoptionen sind von den physischen Einschränkungen abhängig, z. B. von Benutzer- oder On-Premises-Ressourcen. Diese Einschränkungen können durch Edge-Standorte oder die Ressourcenplatzierung wettgemacht werden.

Best Practices

- [PERF04-BP01 Verstehen der Auswirkungen des Netzwerks auf die Leistung](#)
- [PERF04-BP02 Evaluieren verfügbarer Netzwerk-Features](#)
- [PERF04-BP03 Auswählen von entsprechend dedizierter Konnektivität oder VPN für Ihre Workload](#)
- [PERF04-BP04 Lastausgleich verwenden, um den Datenverkehr auf mehrere Ressourcen zu verteilen](#)
- [PERF04-BP05 Auswählen leistungsfördernder Netzwerkprotokolle](#)
- [PERF04-BP06 Wählen Sie den Standort Ihres Workloads basierend auf den Netzwerkanforderungen](#)
- [PERF04-BP07 Optimieren der Netzwerkkonfiguration basierend auf Metriken](#)

PERF04-BP01 Verstehen der Auswirkungen des Netzwerks auf die Leistung

Analysieren und verstehen Sie, wie sich netzwerkbezogene Entscheidungen auf Ihre Workload auswirken, sodass Sie eine effiziente Leistung und ein verbessertes Benutzererlebnis erzielen können.

Typische Anti-Muster:

- Der gesamte Datenverkehr fließt durch Ihre bestehenden Rechenzentren.
- Sie leiten den gesamten Datenverkehr durch zentrale Firewalls, anstatt cloudnative Netzwerksicherheitstools zu verwenden.

- Sie stellen AWS Direct Connect-Verbindungen bereit, ohne die tatsächlichen Nutzungsanforderungen zu verstehen.
- Sie berücksichtigen beim Definieren Ihrer Netzwerklösungen die Workload-Eigenschaften und den Verschlüsselungsaufwand nicht.
- Sie verwenden On-Premises-Konzepte und -Strategien für Netzwerklösungen in der Cloud.

Vorteile der Nutzung dieser bewährten Methode: Indem Sie verstehen, wie das Netzwerk die Workload-Leistung beeinflusst, können Sie potenzielle Engpässe erkennen, die Benutzererfahrung verbessern, die Zuverlässigkeit erhöhen und den Betriebsaufwand verringern, während sich die Workload verändert.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Das Netzwerk ist für die Verbindung zwischen Anwendungskomponenten, Cloud-Services, Edge-Netzwerken und On-Premises-Daten verantwortlich und kann daher die Workload-Leistung wesentlich beeinflussen. Die Benutzererfahrung kann nicht nur durch die Workload-Leistung, sondern auch durch Netzwerklatenz, Bandbreite, Protokolle, Standort, Netzwerküberlastungen, Jitter, Durchsatz und Routingregeln beeinträchtigt werden.

Sie haben eine dokumentierte Liste an Netzwerkanforderungen der Workload, einschließlich Latenz, Paketgröße, Routingregeln, Protokolle und unterstützender Datenverkehrsmuster. Sie überprüfen alle verfügbaren Netzwerklösungen und identifizieren, welcher Service den Netzwerkmerkmalen Ihrer Workload entspricht. Da cloudbasierte Netzwerke schnell geändert werden können, müssen Sie Ihre Netzwerkarchitektur im Laufe der Zeit weiterentwickeln, um die effiziente Leistung zu verbessern.

Implementierungsschritte:

- Definieren und dokumentieren Sie die Anforderungen an die Netzwerkleistung, einschließlich Metriken wie Netzwerklatenz, Bandbreite, Protokolle, Standorte, Datenverkehrsmuster (Spitzen und Frequenz), Durchsatz, Verschlüsselung, Überprüfung und Routingregeln.
- Erfahren Sie mehr über wichtige AWS-Netzwerk-Services wie [VPCs](#), [AWS Direct Connect](#), [Elastic Load Balancing \(ELB\)](#) und [Amazon Route 53](#).
- Erfassen Sie die folgenden wichtigen Netzwerkmerkmale:

Merkmale	Tools und Metriken
Grundlegende Netzwerkmerkmale	• VPC Flow-Protokolle • AWS Transit Gateway-Flow-Protokolle • AWS Transit Gateway Metriken • AWS PrivateLink Metriken
Merkmale von Anwendungsnetzwerken	• Elastic Fabric Adapter • AWS App Mesh Metriken • Metriken für Amazon API Gateway
Merkmale von Edge-Netzwerken	• Amazon-CloudFront-Metriken • Amazon-Route-53-Metriken • AWS Global Accelerator Metriken
Merkmale hybrider Netzwerke	• Direct Connect Metriken • AWS Site-to-Site VPN Metriken • AWS Client VPN Metriken • AWS Cloud-WAN-Metriken
Merkmale von Sicherheitsnetzwerken	• Metriken von AWS Shield, AWS WAF und AWS Network Firewall
Nachverfolgungsmerkmale	• AWS X-Ray • VPC Reachability Analyzer • Network Access Analyzer • Amazon Inspector • Amazon CloudWatch RUM

- Benchmarks für die Netzwerkleistung festlegen und testen:
- Führen Sie [Benchmark](#)-Tests für den Netzwerkdurchsatz durch, da einige Faktoren die Amazon-EC2-Netzwerkleistung beeinflussen können, wenn sich Instances in derselben VPC befinden. Messen Sie die Netzwerkbandbreite zwischen Linux-Instances von Amazon EC2 in der gleichen VPC.

- Führen Sie [Lasttests](#) durch, um mit Netzwerklösungen und -optionen zu experimentieren.

Ressourcen

Zugehörige Dokumente:

- [Application Load Balancer](#)
- [EC2: Enhanced Networking unter Linux](#)
- [EC2: Enhanced Networking unter Windows](#)
- [EC2: Platzierungsgruppen](#)
- [Enabling Enhanced Networking with the Elastic Network Adapter \(ENA\) on Linux Instances](#)
- [Network Load Balancer](#)
- [Netzwerkprodukte mit AWS](#)
- [Transit-Gateway](#)
- [Umstellung auf latenzbasiertes Routing in Amazon Route 53](#)
- [VPC-Endpunkte](#)

Zugehörige Videos:

- [AWS re:Invent 2023 – AWS-Netzwerkgrundlagen](#)
- [AWS re:Invent 2023 – Welche Vorteile bietet die Vernetzung für Ihre Anwendung?](#)
- [AWS re:Invent 2023 – Erweiterte VPC-Designs und neue Funktionen](#)
- [AWS re:Invent 2023 – Leitfaden für Entwickler von Cloud-Netzwerken](#)
- [AWS re:Invent 2019 – Konnektivität mit AWS und hybriden AWS-Netzwerkarchitekturen](#)
- [AWS re:Invent 2019 – Optimizing Network Performance for Amazon EC2 Instances](#)
- [AWS Summit Online – Verbessern der Leistung von globalen Netzwerken für Anwendungen](#)
- [AWS re:Invent 2020 – Bewährte Methoden für Netzwerke und Tipps für das Well-Architected Framework](#)
- [AWS re:Invent 2020 – Bewährte Methoden für AWS-Netzwerke in umfangreichen Migrationen](#)

Zugehörige Beispiele:

- [AWS Transit Gateway und skalierbare Sicherheitslösungen](#)

- [Workshops zu AWS-Netzwerken](#)
- [Praktischer Workshop zur Netzwerk-Firewall](#)
- [Beobachten und Diagnostizieren Ihres Netzwerks in AWS](#)
- [Finden und Beheben von Netzwerkfehlerkonfigurationen in AWS](#)

PERF04-BP02 Evaluieren verfügbarer Netzwerk-Features

Prüfen Sie die Netzwerk-Features in der Cloud, mit denen die Leistung unter Umständen verbessert werden kann. Messen Sie die Auswirkungen der Features anhand von Tests, Metriken und Analysen. Nutzen Sie beispielsweise die verfügbaren Features auf Netzwerkebene, um die Latenz, die Netzwerkentfernung oder den Jitter zu reduzieren.

Typische Anti-Muster:

- Sie bleiben innerhalb einer Region, da sich Ihre Firmenzentrale dort befindet.
- Sie verwenden Firewalls anstelle von Sicherheitsgruppen, um den Datenverkehr zu filtern.
- Sie unterbrechen TLS für die Überprüfung des Datenverkehrs, anstatt sich auf Sicherheitsgruppen, Endpunktrichtlinien und andere cloudnative Funktionen zu verlassen.
- Sie nutzen nur eine subnetzbasierte Segmentierung anstelle von Sicherheitsgruppen.

Vorteile der Nutzung dieser bewährten Methode: Wenn Sie alle Service-Features und Optionen evaluieren, kann dies die Workload-Leistung verbessern, die Infrastrukturkosten senken, den Verwaltungsaufwand für die Workload reduzieren und die allgemeine Sicherheit erhöhen. Dank der weltweiten Abdeckung von AWS können Sie Ihren Kunden stets das bestmögliche Netzwerkerlebnis bieten.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

AWS bietet Services wie [AWS Global Accelerator](#) und [Amazon CloudFront](#), die zur Verbesserung der Netzwerkleistung beitragen können, während die meisten AWS-Services über Produktfeatures (wie das Feature [Amazon S3 Transfer Acceleration](#)) zur Optimierung des Netzwerkverkehrs verfügen.

Sehen Sie sich die verfügbaren Konfigurationsoptionen für das Netzwerk an und finden Sie heraus, wie sich diese auf Ihre Workload auswirken. Die Leistungsoptimierung hängt davon ab, wie diese

Optionen mit Ihrer Architektur interagieren und welche Auswirkungen sie auf die gemessene Leistung und auf die Benutzererfahrung haben.

Implementierungsschritte

- Erstellen Sie eine Liste der Workload-Komponenten.
 - Erwägen Sie die Verwendung von [AWS Cloud WAN](#), um das Netzwerk Ihrer Organisation aufzubauen, zu verwalten und zu überwachen, wenn Sie ein einheitliches globales Netzwerk aufbauen.
 - Überwachen Sie Ihre globalen und Kernnetzwerke mit [Metriken für Amazon CloudWatch Logs](#). Nutzen Sie [Amazon CloudWatch RUM](#), das Erkenntnisse bietet, die dazu beitragen, das digitale Erlebnis der Benutzer zu identifizieren, zu verstehen und zu verbessern.
 - Zeigen Sie die aggregierte Netzwerklatenz zwischen AWS-Regionen und den Availability Zones sowie innerhalb der einzelnen Availability Zones mit [AWS Network Manager](#) an, um einen Einblick in das Verhältnis zwischen der Leistung Ihrer Anwendung und der Leistung des zugrunde liegenden AWS-Netzwerks zu erhalten.
 - Verwenden Sie ein vorhandenes CMDB-Tool (Configuration Management Database) oder einen Service wie [AWS Config](#), um eine Bestandsaufnahme Ihrer Workload und seiner Konfiguration zu erstellen.
- Wenn es sich um eine bestehende Workload handelt, ermitteln und dokumentieren Sie die Benchmark für Ihre Leistungsmetriken. Konzentrieren Sie sich dabei auf Engpässe und Bereiche mit Verbesserungspotenzial. Leistungsbezogene Netzwerkmetriken werden je nach geschäftlichen Anforderungen und Workload-Merkmalen für die einzelnen Workloads unterschiedlich sein. Für den Anfang könnte die Prüfung folgender Metriken für Ihre Workload wichtig sein: Bandbreite, Latenz, Paketverlust, Jitter und erneute Übertragungen.
- Wenn es sich um eine neue Workload handelt, führen Sie [Lasttests](#) durch, um Leistungsengpässe zu ermitteln.
- Prüfen Sie für die ermittelten Leistungsengpässe die Konfigurationsoptionen Ihrer Lösungen, um Möglichkeiten zur Leistungsverbesserung zu finden. Informieren Sie sich über die folgenden wichtigen Netzwerkoptionen und -Features:

Verbesserungsmöglichkeit	Lösung
Netzwerkpfad oder -routen	Verwenden Sie Network Access Analyzer , um Pfade oder Routen zu ermitteln.

Verbesserungsmöglichkeit	Lösung
Netzwerkprotokolle	Siehe PERF04-BP05 Auswählen leistungsfördernder Netzwerkprotokolle
Netzwerktopologie	Bewerten Sie Ihre betrieblichen und leistungsbezogenen Kompromisse zwischen VPC Peering und AWS Transit Gateway beim Verbinden mehrerer Konten. AWS Transit Gateway vereinfacht die Art und Weise, wie Sie all Ihre VPCs miteinander verbinden, die sich über Tausende von AWS-Konten-Netzwerken bis hin zu On-Premises-Netzwerken erstrecken können. Teilen Sie Ihr AWS Transit Gateway zwischen mehreren Konten mit AWS Resource Access Manager. Siehe PERF04-BP03 Auswählen von entsprechend dedizierter Konnektivität oder VPN für Ihre Workload

Verbesserungsmöglichkeit	Lösung
Netzwerksservices	AWS Global Accelerator ist ein Netzwerkservice, der die Leistung des Benutzerdatenverkehrs unter Verwendung der globalen Netzwerkinfrastruktur von AWS um bis zu 60 % verbessert. Amazon CloudFront kann die Leistung Ihrer Workload-Inhaltsbereitstellung und Latenz weltweit verbessern. Verwenden Sie Lambda@edge, um Funktionen auszuführen, die den Inhalt, den CloudFront bereitstellt, besser an die Benutzer anpassen, die Latenz reduzieren und die Leistung erhöhen. Amazon Route 53 bietet Optionen für latenzbasiertes Routing, Geolocation-Routing, Geoproximity-Routing und IP-basiertes Routing, mit denen Sie die Leistung Ihrer Workloads für eine globale Zielgruppe verbessern können. Ermitteln Sie, welche Routing-Option Ihre Workload-Leistung optimieren würde. Prüfen Sie dazu Ihren Workload-Datenverkehr und den Benutzerstandort bei der globalen Verteilung Ihrer Workload.

Verbesserungsmöglichkeit	Lösung
Speicherressourcen-Features	Durch das Feature Amazon S3 Transfer Acceleration profitieren externe Benutzer beim Hochladen von Daten in Amazon S3 von den Netzwerkoptimierungen von CloudFront. Dies erleichtert die Übertragung großer Datenmengen von Remotestandorten ohne spezielle Konnektivität zur AWS Cloud. Multi-Region-Zugangspunkte in Amazon S3 replizieren Inhalte in mehreren Regionen und vereinfachen die Workload durch die Bereitstellung eines Zugangspunkts. Bei Verwendung eines Multi-Region-Zugangspunkts können Sie Daten anfordern oder in Amazon S3 schreiben, wobei der Service den Bucket mit der geringsten Latenz ermittelt.

Verbesserungsmöglichkeit	Lösung
Compute-Ressourcen-Features	Die von Amazon-EC2-Instances, Container n und Lambda-Funktionen verwendeten Elastic-Network-Schnittstellen (ENI) sind pro Fluss begrenzt. Überprüfen Sie Ihre Platzierungsgruppen, um Ihren EC2-Netzwerkdurchsatz zu optimieren. Um Engpässe auf Pro-Fluss-Basis zu vermeiden, sollten Sie Ihre Anwendung so gestalten, dass mehrere Flüsse verwendet werden. Um Ihre datenverarbeitungsbezogenen Netzwerkmetriken zu überwachen und Einblicke in diese Metriken zu erhalten, verwenden Sie CloudWatch-Metriken und ethtool. Der ethtool-Befehl ist im ENA-Treiber enthalten und stellt zusätzliche netzwerkbezogene Metriken zur Verfügung, die als benutzerdefinierte Metrik in CloudWatch veröffentlicht werden können. Amazon Elastic Network Adapters (ENA) ermöglichen eine weitere Optimierung, indem sie für Ihre Instances innerhalb einer Cluster-Placement-Gruppe einen besseren Durchsatz bereitstellen. Elastic Fabric Adapter (EFA) ist eine Netzwerkschnittstelle für Amazon-EC2-Instances, mit der Sie Workloads, für die ein hohes Maß an Kommunikation zwischen Knoten anfällt, unter AWS bedarfsgesteuert ausführen können. Für Amazon EBS optimierte Instances verwenden einen optimierten Konfigurations-Stack und bieten zusätzliche, dediziert

Verbesserungsmöglichkeit	Lösung
	e Kapazität, um die Amazon EBS-I/O zu erhöhen.

Ressourcen

Zugehörige Dokumente:

- [Application Load Balancer](#)
- [EC2: Enhanced Networking unter Linux](#)
- [EC2: Enhanced Networking unter Windows](#)
- [EC2: Platzierungsgruppen](#)
- [Enabling Enhanced Networking with the Elastic Network Adapter \(ENA\) on Linux Instances](#)
- [Network Load Balancer](#)
- [Netzwerkprodukte mit AWS](#)
- [Umstellung auf latenzbasiertes Routing in Amazon Route 53](#)
- [VPC-Endpunkte](#)
- [VPC Flow Logs](#)

Zugehörige Videos:

- [AWS re:Invent 2023 – Sind Sie bereit für Neues? Gestaltung von Netzwerken für Wachstum und Flexibilität](#)
- [AWS re:Invent 2023 – Erweiterte VPC-Designs und neue Funktionen](#)
- [AWS re:Invent 2023 – Leitfaden für Entwickler von Cloud-Netzwerken](#)
- [AWS re:Invent 2022 – Ausführliche Beschreibung der AWS-Netzwerkinfrastruktur](#)
- [AWS re:Invent 2019 – Konnektivität mit AWS und hybriden AWS-Netzwerkarchitekturen](#)
- [AWS re:Invent 2018 – Optimizing Network Performance for Amazon EC2 Instances](#)
- [AWS Global Accelerator](#)

Zugehörige Beispiele:

- [AWS Transit Gateway und skalierbare Sicherheitslösungen](#)

- [Workshops zu AWS-Netzwerken](#)
- [Überwachung und Diagnose Ihres Netzwerks](#)
- [Finden und Beheben von Netzwerkfehlfunktionen in AWS](#)

PERF04-BP03 Auswählen von entsprechend dedizierter Konnektivität oder VPN für Ihre Workload

Wenn Hybrid-Konnektivität für die Verbindung von On-Premises- und Cloud-Ressourcen erforderlich ist, stellen Sie ausreichend Bandbreite bereit, um Ihre Leistungsanforderungen zu erfüllen. Schätzen Sie die Anforderungen an Bandbreite und Latenz für Ihre hybride Workload ab. Diese Zahlen dienen als Grundlage für die Größenanpassung.

Typische Anti-Muster:

- Sie evaluieren nur VPN-Lösungen für Ihre Netzwerk-Verschlüsselungsanforderungen.
- Sie bewerten keine Optionen für Sicherung oder redundante Verbindungen.
- Sie identifizieren nicht alle Workload-Anforderungen (Verschlüsselung, Protokoll, Bandbreite und Traffic-Bedarf).

Vorteile der Nutzung dieser bewährten Methode: Durch die Auswahl und Konfiguration geeigneter Konnektivitätslösungen wird die Zuverlässigkeit Ihrer Workloads erhöht und die Leistung maximiert. Indem Sie die Workload-Anforderungen identifizieren, im Voraus planen und hybride Lösungen evaluieren, verringern Sie teure physische Netzwerkänderungen sowie den Betriebsaufwand und verkürzen die Amortisationszeit.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Entwickeln Sie eine hybride Netzwerkarchitektur, die auf Ihren Bandbreitenanforderungen basiert. [Direct Connect](#) ermöglicht es Ihnen, Ihr On-Premises-Netzwerk privat mit AWS zu verbinden. Sie ist geeignet, wenn Sie eine hohe Bandbreite und eine geringe Latenz bei gleichbleibender Leistung benötigen. Eine VPN-Verbindung stellt eine sichere Verbindung über das Internet her. Sie wird verwendet, wenn lediglich eine temporäre Verbindung erforderlich ist, wenn die Kosten eine Rolle spielen, oder wenn bei der Verwendung von Direct Connect darauf gewartet wird, dass eine resiliente physische Netzwerkkonnektivität hergestellt wird.

Wenn Ihre Bandbreitenanforderungen hoch sind, könnten Sie mehrere Direct Connect oder VPN-Services in Betracht ziehen. Der Lastausgleich für den Datenverkehr kann über die Services hinweg

erfolgen. Allerdings empfehlen wir aufgrund der Latenz- und Bandbreitenunterschiede keinen Lastausgleich zwischen Direct Connect und VPN.

Implementierungsschritte

- Schätzen Sie die Anforderungen an Bandbreite und Latenz für Ihre bestehenden Anwendungen ab.
 - Für bestehende Workloads, die auf AWS umgestellt werden, nutzen Sie die Daten aus Ihren internen Systemen zur Überwachung des Netzwerks.
 - Bei neuen oder bestehenden Workloads, für die Sie keine Monitoring-Daten haben, beraten Sie sich mit den Besitzern der Produkte, um angemessene Metriken für die Leistung zu bestimmen und ein gutes Benutzererlebnis zu gewährleisten.
- Wählen Sie eine dedizierte Verbindung oder ein VPN als Konnektivitätsoption aus. Je nach den Anforderungen der Workload (Verschlüsselung, Bandbreite und Traffic-Bedarf) können Sie entweder AWS Direct Connect oder [Site-to-Site VPN](#) (oder beides) auswählen. Das folgende Diagramm kann Ihnen bei der Wahl der geeigneten Verbindungsart helfen.
- [AWS Direct Connect](#) liefert dedizierte Konnektivität für die AWS-Umgebung, von 50 Mbit/s bis zu 100 Gbit/s, entweder über dedizierte Verbindungen oder über gehostete Verbindungen. So erhalten Sie eine verwaltete und kontrollierte Latenz und bereitgestellte Bandbreite, damit sich Ihre Workload effizient mit anderen Umgebungen verbinden kann. Mit einem AWS Direct Connect-Partner können Sie eine End-to-End-Konnektivität aus mehreren Umgebungen nutzen und so ein erweitertes Netzwerk mit konsistenter Leistung bereitstellen. AWS bietet eine Skalierung der Bandbreite für Direct Connect-Verbindungen entweder über native 100 Gbit/s, Link Aggregation Group (LAG) oder BGP Equal-Cost Multipath (ECMP).
- Die AWS [Site-to-Site VPN](#) bietet einen verwalteten VPN-Service, der das IPsec (Internet Protocol Security) unterstützt. Wenn eine VPN-Verbindung erstellt wird, besteht die VPN-Verbindung aus zwei Tunneln, um eine hohe Verfügbarkeit zu gewährleisten.
- Folgen Sie der AWS-Dokumentation, um eine geeignete Verbindungsoption auszuwählen:
 - Wenn Sie sich für die Verwendung von Direct Connect entscheiden, wählen Sie die entsprechende Bandbreite für Ihre Konnektivität aus.
 - Wenn Sie eine AWS Site-to-Site VPN über mehrere Standorte hinweg nutzen, um eine Verbindung mit einer AWS-Region herzustellen, verwenden Sie eine [beschleunigte Site-to-Site-VPN-Verbindung](#), um die Netzwerkleistung zu verbessern.
 - Wenn Ihr Netzwerkdesign aus einer IPsec-VPN-Verbindung über [AWS Direct Connect](#) besteht, sollten Sie die Verwendung von privatem IP-VPN in Betracht ziehen, um die Sicherheit zu

verbessern und eine Segmentierung zu erzielen. [AWS Site-to-Site Private IP VPN](#) wird auf einer virtuellen Transit-Schnittstelle (VIF) bereitgestellt.

- [AWS Direct Connect SiteLink](#) ermöglicht die Schaffung redundanter Verbindungen mit niedriger Latenz zwischen Ihren Rechenzentren weltweit, indem Daten über den schnellsten Weg zwischen [AWS Direct Connect-Standorten](#) gesendet werden, unter Umgehung von AWS-Regionen.
- Überprüfen Sie Ihr Konnektivitäts-Setup, bevor Sie es in der Produktion einsetzen. Führen Sie Sicherheits- und Leistungstests durch, um sicherzustellen, dass das Setup Ihre Anforderungen an Bandbreite, Zuverlässigkeit, Latenz und Compliance erfüllt.
- Überwachen Sie regelmäßig die Leistung und Nutzung Ihrer Konnektivität und optimieren Sie sie bei Bedarf.

Flussdiagramm zur deterministischen Leistung

Ressourcen

Zugehörige Dokumente:

- [Netzwerkprodukte mit AWS](#)
- [AWS Transit Gateway](#)
- [VPC-Endpunkte](#)
- [Building a Scalable and Secure Multi-VPC AWS Network Infrastructure](#)
- [Client-VPN](#)

Zugehörige Videos:

- [AWS re:Invent 2023 – Aufbau einer hybriden Netzwerkkonnektivität mit AWS](#)
- [AWS re:Invent 2023 – Sichere Remote-Verbindung zu AWS](#)
- [AWS re:Invent 2022 – Optimizing performance with Amazon CloudFront](#)
- [AWS re:Invent 2019 – Konnektivität mit AWS und hybriden AWS-Netzwerkarchitekturen](#)
- [AWS re:Invent 2020 – AWS Transit Gateway Connect](#)

Zugehörige Beispiele:

- [AWS Transit Gateway und skalierbare Sicherheitslösungen](#)
- [AWS Workshops zu -Netzwerken](#)

PERF04-BP04 Lastausgleich verwenden, um den Datenverkehr auf mehrere Ressourcen zu verteilen

Verteilen Sie den Datenverkehr auf mehrere Ressourcen oder Services, damit Ihren Workload von der Elastizität der Cloud zu profitieren kann. Sie können den Lastausgleich auch nutzen, um die Terminierung von Verschlüsselung auszulagern. So lässt sich die Leistung und Zuverlässigkeit optimieren und der Datenverkehr effektiv verwalten und weiterleiten.

Typische Anti-Muster:

- Sie berücksichtigen bei der Wahl des Load-Balancer-Typs nicht die Anforderungen Ihrer Workload.
- Sie nutzen die Features des Load Balancers nicht zur Optimierung der Leistung.
- Die Workload ist direkt mit dem Internet verbunden, ohne dass ein Load Balancer zum Einsatz kommt.
- Sie leiten den gesamten Internetverkehr über vorhandene Load Balancer weiter.
- Sie nutzen einen generischen TCP-Lastausgleich und lassen die SSL-Verschlüsselung von den einzelnen Rechenknoten verarbeiten.

Vorteile der Nutzung dieser bewährten Methode: Ein Load Balancer verarbeitet die variierende Last des Anwendungsdatenverkehrs in einer einzigen oder in mehreren Availability Zones und ermöglicht eine hohe Verfügbarkeit, Auto Scaling sowie eine bessere Nutzung für Ihre Workload.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Load Balancer fungieren als Eingangspunkt für Ihre Workload und verteilen den Datenverkehr von dort aus auf Ihre Backend-Ziele – wie Computing-Instances oder Container –, um die Nutzung zu verbessern.

Die Wahl des richtigen Load-Balancer-Typs ist der erste Schritt zur Optimierung Ihrer Architektur. Starten Sie mit einer Auflistung Ihrer Workload-Merkmale wie Protokoll (z. B. TCP, HTTP, TLS oder WebSockets), Zieltyp (z. B. Instances, Container oder Serverless), Anwendungsanforderungen (z. B. langfristige Verbindungen, Benutzerauthentifizierung oder Stickiness) und Platzierung (z. B. Region, lokale Zone, Outposts oder Zonenisolierung).

AWS stellt für Ihre Anwendungen mehrere Modelle zur Verwendung des Lastausgleichs bereit. [Application Load Balancer](#) eignet sich optimal für den Lastausgleich von HTTP- und HTTPS-Datenverkehr. Sie profitieren dadurch von einer erweiterten Anfrageweiterleitung, die es Ihnen ermöglicht, moderne Anwendungsarchitekturen mit Microservices und Containern bereitzustellen.

[Network Load Balancer](#) eignet sich optimal für den Lastausgleich von TCP-Datenverkehr, wenn eine hohe Leistung erforderlich ist. Hiermit lassen sich mit konstant geringer Latenz Millionen Anforderungen pro Sekunde und plötzliche Datenverkehrsspitzen oder schwankende Datenverkehrsmuster verarbeiten.

[Elastic Load Balancing](#) ermöglicht die integrierte Zertifikatverwaltung und SSL/TLS-Entschlüsselung. Auf diese Weise können Sie die SSL-Einstellungen des Load Balancer flexibel zentral verwalten und CPU-intensive Arbeitsschritte für Ihre Workload auslagern.

Nachdem Sie sich für den richtigen Load Balancer entschieden haben, können Sie damit beginnen, seine Features zu nutzen, um die Belastung Ihres Backends durch den Datenverkehr zu verringern.

So können Sie beispielsweise sowohl mit Application Load Balancer (ALB) als auch mit Network Load Balancer (NLB) die SSL/TLS-Verschlüsselung auslagern, was die Möglichkeit bietet, den CPU-intensiven TLS-Handshake bei Ihren Zielen zu vermeiden und die Verwaltung der Zertifikate zu verbessern.

Wenn Sie SSL/TLS-Offloading in Ihrem Load Balancer konfigurieren, übernimmt dieser die Verschlüsselung des Datenverkehrs von und zu den Clients. Er leitet den Datenverkehr dann unverschlüsselt an Ihre Backends weiter, wodurch Ihre Backend-Ressourcen entlastet werden und die Reaktionszeit für die Clients verbessert wird.

Application Load Balancer kann außerdem HTTP/2-Datenverkehr ausliefern, ohne dass Sie ihn auf Ihren Zielen unterstützen müssen. Diese einfache Entscheidung kann die Reaktionszeit Ihrer Anwendung verbessern, da HTTP/2 TCP-Verbindungen effizienter nutzt.

Bei der Definition der Architektur sollten Sie die Anforderungen an die Latenz Ihrer Workload berücksichtigen. Wenn Sie beispielsweise eine latenzempfindliche Anwendung haben, können Sie sich für Network Load Balancer mit einer extrem niedrigen Latenz entscheiden. Alternativ können Sie Ihre Workload auch näher an Ihre Kunden heranbringen, indem Sie Application Load Balancer in [AWS Local Zones](#) oder sogar in [AWS Outposts](#) nutzen.

Eine weitere Überlegung für latenzempfindliche Workloads ist der zonenübergreifende Lastausgleich. Beim zonenübergreifenden Lastausgleich nimmt jeder Load-Balancer-Knoten eine Verteilung des Datenverkehrs auf die registrierten Ziele in allen zulässigen Availability Zones vor.

Verwenden Sie die Auto-Scaling-Integration für Ihren Load Balancer. Einer der Schlüssel für ein leistungsfähiges System ist die richtige Größenanpassung Ihrer Backend-Ressourcen. Zu diesem Zweck können Sie Load-Balancer-Integrationen für Backend-Zielressourcen nutzen. Mithilfe der Load-Balancer-Integration mit Auto-Scaling-Gruppen werden Ziele je nach Bedarf als Reaktion auf den eingehenden Datenverkehr zum Load Balancer hinzugefügt oder aus ihm entfernt. Load Balancer können für containerisierte Workloads außerdem mit [Amazon ECS](#) und [Amazon EKS](#) integriert werden.

- [Amazon ECS – Service-Lastausgleich](#)
- [Application Load Balancing in Amazon EKS](#)
- [Network Load Balancing in Amazon EKS](#)

Implementierungsschritte

- Definieren Sie Ihre Anforderungen an den Lastausgleich, einschließlich Datenverkehrsvolumen, Verfügbarkeit und Anwendungsskalierbarkeit.
- Wählen Sie den richtigen Load-Balancer-Typ für Ihre Anwendung.
 - Verwenden Sie den Application Load Balancer für HTTP/HTTPS-Workloads.
 - Verwenden Sie den Network Load Balancer für Nicht-HTTP-Workloads, die TCP oder UDP nutzen.
 - Verwenden Sie eine Kombination aus beidem ([ALB als Ziel des NLB](#)), wenn Sie die Features beider Produkte nutzen möchten. Dies ist zum Beispiel möglich, wenn Sie die statischen IP-Adressen von NLB zusammen mit dem HTTP-Header-basierten Routing von ALB verwenden möchten oder wenn Sie Ihre HTTP-Workload für einen [AWS PrivateLink](#) bereitstellen wollen.
- Einen vollständigen Vergleich der Load Balancer finden Sie unter [ELB-Produktvergleich](#).
- Verwenden Sie nach Möglichkeit SSL/TLS-Offloading.
 - Konfigurieren Sie HTTPS/TLS-Listeners mit [Application Load Balancer](#) und [Network Load Balancer](#), die in [AWS Certificate Manager](#) integriert sind.
 - Beachten Sie, dass einige Workloads aus Compliancegründen eine Ende-zu-Ende-Verschlüsselung benötigen können. In diesem Fall ist es erforderlich, die Verschlüsselung an den Zielen zuzulassen.
 - Bewährte Methoden für die Sicherheit finden Sie unter [SEC09-BP02 Erzwingen einer Verschlüsselung bei der Übertragung](#).
- Wählen Sie den richtigen Routing-Algorithmus (nur ALB) aus.

- Der Routing-Algorithmus kann einen entscheidenden Einfluss darauf haben, wie gut Ihre Backend-Ziele ausgelastet sind und wie sie die Leistung beeinflussen. ALB bietet beispielsweise [zwei Optionen für Routing-Algorithmen](#):
- Am wenigsten ausstehende Anfragen: Verwenden Sie diese Option, um eine bessere Verteilung der Last auf Ihre Backend-Ziele zu erreichen, wenn die Anfragen für Ihre Anwendung unterschiedlich komplex sind oder Ihre Ziele unterschiedliche Kapazitäten für die Verarbeitung haben.
- Round Robin: Verwenden Sie diese Option, wenn die Anfragen und Ziele ähnlich sind oder wenn Sie die Anfragen gleichmäßig auf die Ziele verteilen müssen.
- Ziehen Sie eine zonenübergreifende Verarbeitung oder Zonenisolierung in Betracht.
 - Verwenden Sie die deaktivierte zonenübergreifende Isolierung (Zonenisolierung), um die Latenz zu verbessern und Domains mit Zonenfehlern zu vermeiden. Sie ist standardmäßig in NLB deaktiviert und Sie können sie in [ALB pro Zielgruppe ausschalten](#).
 - Verwenden Sie die aktivierte zonenübergreifende Verarbeitung für eine höhere Verfügbarkeit und Flexibilität. Standardmäßig ist Cross-Zone für ALB aktiviert und [Sie können sie in NLB pro Zielgruppe aktivieren](#).
- Aktivieren Sie HTTP-Keep-Alives für Ihre HTTP-Workloads (nur ALB). Mit diesem Feature kann der Load Balancer Backend-Verbindungen wiederverwenden, bis die Keep-Alive-Zeit abgelaufen ist, wodurch sich Ihre HTTP-Anfrage- und Reaktionszeiten verbessern und die Auslastung der Ressourcen auf Ihren Backend-Zielen reduziert wird. Details zu dieser Funktion für Apache und Nginx finden Sie unter [Was sind die optimalen Einstellungen für die Verwendung von Apache oder NGINX als Backend-Server für ELB?](#)
- Aktivieren Sie die Überwachung für Ihren Load Balancer.
 - Aktivieren Sie die Zugriffsprotokolle für Ihren [Application Load Balancer](#) und [Network Load Balancer](#).
 - Die wichtigsten zu berücksichtigenden Elemente für ALB sind `request_processing_time`, `request_processing_time` und `response_processing_time`.
 - Die wichtigsten Elemente für NLB sind `connection_time` und `tls_handshake_time`.
 - Bereiten Sie sich darauf vor, die Protokolle bei Bedarf abfragen zu können. Sie können mittels Amazon Athena sowohl [ALB-Protokolle](#) als auch [NLB-Protokolle](#) abfragen.
 - Erstellen Sie Warnungen für leistungsbezogene Metriken wie [TargetResponseTime für ALB](#).

Ressourcen

Zugehörige Dokumente:

- [ELB-Produktvergleich](#)
- [AWS Global Infrastructure](#)
- [Verbesserung der Leistung und Senkung der Kosten durch Availability Zone-Affinität](#)
- [Step by step for Log Analysis with Amazon Athena](#)
- [Abfragen von Application-Load-Balancer-Protokollen](#)
- [Überwachen Ihrer Application Load Balancer](#)
- [Monitor your Network Load Balancer](#)
- [Um den Datenverkehr über die Instances in Ihrer Auto-Scaling-Gruppe zu verteilen, verwenden Sie Elastic-Load-Balancing](#)

Zugehörige Videos:

- [AWS re:Invent 2023 – Welche Vorteile bietet die Vernetzung für Ihre Anwendung?](#)
- [AWS re:Inforce 20: How to use Elastic Load Balancing to enhance your security posture at scale](#)
- [AWS re:Invent 2018: Elastic Load Balancing: Deep Dive and Best Practices](#)
- [AWS re:Invent 2021 – So wählen Sie den richtigen Load Balancer für Ihre AWS-Workloads aus](#)
- [AWS re:Invent 2019: Holen Sie das Beste aus Elastic Load Balancing für verschiedene Workloads heraus](#)

Zugehörige Beispiele:

- [Gateway Load Balancer](#)
- [CDK and CloudFormation samples for Log Analysis with Amazon Athena](#)

PERF04-BP05 Auswählen leistungsfördernder Netzwerkprotokolle

Treffen Sie Entscheidungen über Protokolle für die Kommunikation zwischen Systemen und Netzwerken auf Grundlage der Auswirkungen, die sich für die Leistung der Workload ergeben.

In Bezug auf die Erzielung eines höheren Durchsatzes besteht eine Beziehung zwischen der Latenz und der Bandbreite. Wenn Ihr File Transfer über TCP (Transmission Control Protocol) erfolgt,

verringern höhere Latenzen höchstwahrscheinlich den gesamten Durchsatz. Es gibt verschiedene Ansätze, dies mit der TCP-Optimierung und optimierten Übertragungsprotokollen zu lösen. Eine Lösung besteht jedoch in der Verwendung des User Datagram Protocol (UDP).

Typische Anti-Muster:

- Sie verwenden TCP unabhängig von den Leistungsanforderungen für alle Workloads.

Vorteile der Nutzung dieser bewährten Methode: Wenn Sie sicherstellen, dass ein geeignetes Protokoll für die Kommunikation zwischen Benutzern und Workload-Komponenten verwendet wird, können Sie das Benutzererlebnis für Ihre Anwendungen insgesamt verbessern. Das verbindungslose UDP ermöglicht zwar beispielsweise eine hohe Geschwindigkeit, bietet aber weder eine erneute Übertragung noch hohe Zuverlässigkeit. TCP ist ein Protokoll mit vollem Funktionsumfang, bringt jedoch einen größeren Overhead für die Verarbeitung der Pakete mit sich.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Wenn Sie in der Lage sind, verschiedene Protokolle für Ihre Anwendung auszuwählen, und Sie über Fachwissen in diesem Bereich verfügen, optimieren Sie Ihre Anwendungs- und Endbenutzererfahrung, indem Sie ein anderes Protokoll verwenden. Beachten Sie, dass dieser Ansatz mit erheblichen Schwierigkeiten verbunden ist und nur versucht werden sollte, wenn Sie Ihre Anwendung zuvor auf andere Weise optimiert haben.

Um die Leistung Ihrer Workload zu verbessern, sollten Sie in erster Linie die Anforderungen an die Latenz und den Durchsatz kennen und dann Netzwerkprotokolle auswählen, die die Leistung optimieren.

Wann sollten Sie TCP verwenden

TCP bietet eine zuverlässige Zustellung von Daten und kann für die Kommunikation zwischen Workload-Komponenten verwendet werden, bei denen die Zuverlässigkeit und die garantierte Zustellung von Daten wichtig sind. Viele webbasierte Anwendungen verlassen sich auf TCP-basierte Protokolle wie HTTP und HTTPS, um TCP-Sockets für die Kommunikation zwischen Anwendungskomponenten zu öffnen. E-Mail- und Dateidatenübertragung sind gängige Anwendungen, die auch TCP verwenden, da es sich um einen einfachen und zuverlässigen Übertragungsmechanismus zwischen Anwendungskomponenten handelt. Die Verwendung von TLS mit TCP kann zu einem gewissen Overhead bei der Kommunikation führen, was eine erhöhte Latenz

und einen verringerten Durchsatz zur Folge haben kann. Sie bietet jedoch den Vorteil der Sicherheit. Der Overhead entsteht vor allem durch den zusätzlichen Aufwand des Handshake-Prozesses, der mehrere Roundtrips in Anspruch nehmen kann. Sobald der Handshake abgeschlossen ist, ist der Overhead für die Ver- und Entschlüsselung der Daten relativ gering.

Wann sollten Sie UDP verwenden

UDP ist ein verbindungsloses Protokoll und eignet sich daher für Anwendungen, die eine schnelle, effiziente Übertragung benötigen, wie z. B. die Protokollierung, die Überwachung und VoIP-Daten. Ziehen Sie die Verwendung von UDP auch in Betracht, wenn Sie Workload-Komponenten haben, die auf kleine Abfragen von einer großen Anzahl von Clients reagieren, um eine optimale Leistung der Workload zu gewährleisten. Datagram Transport Layer Security (DTLS) ist die UDP-Entsprechung von Transport Layer Security (TLS). Bei der Verwendung von DTLS mit UDP entsteht der Overhead durch die Verschlüsselung und Entschlüsselung der Daten, da der Handshake-Prozess vereinfacht ist. DTLS fügt den UDP-Paketen außerdem einen geringen Overhead hinzu, da es zusätzliche Felder zur Angabe der Sicherheitsparameter und zur Erkennung von Manipulationen umfasst.

Wann sollten Sie SRD verwenden

Scalable Reliable Datagram (SRD) ist ein Netzwerktransportprotokoll, das für Workloads mit hohem Durchsatz optimiert ist, da es in der Lage ist, den Datenverkehr über mehrere Pfade zu verteilen und sich schnell von Paketverlusten oder Verbindungsfehlern zu erholen. SRD eignet sich daher am besten für HPC-Workloads (High Performance Computing), die einen hohen Durchsatz und eine geringe Latenz bei der Kommunikation zwischen Computing-Knoten erfordern. Dazu gehören z. B. parallele Verarbeitungsaufgaben wie Simulationen, Modellierung und Datenanalyse, bei denen große Mengen an Daten zwischen den Knoten übertragen werden müssen.

Implementierungsschritte

- Verwenden Sie die Services [AWS Global Accelerator](#) und [AWS Transfer Family](#), um den Durchsatz Ihrer Anwendungen für den File Transfer online zu verbessern. Der AWS Global Accelerator-Service hilft Ihnen, die Latenz zwischen Ihren Client-Geräten und Ihrer Workload auf AWS zu verringern. Mit AWS Transfer Family können Sie TCP-basierte Protokolle wie Secure Shell File Transfer Protocol (SFTP) und File Transfer Protocol over SSL (FTPS) verwenden, um Ihre Dateiübertragungen zu AWS-Speicherservices sicher zu skalieren und zu verwalten.
- Bestimmen Sie anhand der Netzwerklatenz, ob TCP für die Kommunikation zwischen Workload-Komponenten geeignet ist. Wenn die Netzwerklatenz zwischen Ihrer Client-Anwendung und dem Server hoch ist, kann der TCP-Drei-Wege-Handshake einige Zeit in Anspruch nehmen, was sich auf die Reaktionsfähigkeit Ihrer Anwendung auswirkt. Metriken wie Time to First Byte (TTFB) und

Round-Trip-Zeit (RTT) können zur Messung der Netzwerklatenz verwendet werden. Wenn Ihre Workload dynamische Inhalte für Benutzer bereitstellt, sollten Sie die Verwendung von [Amazon CloudFront](#) in Betracht ziehen. So wird eine dauerhafte Verbindung zu jeder Quelle für dynamische Inhalte hergestellt, um die Zeit für den Verbindungsaufbau zu vermeiden, die sonst jede Client-Anfrage verlangsamen würde.

- Die Verwendung von TLS mit TCP oder UDP kann aufgrund der Auswirkungen der Ver- und Entschlüsselung zu einer erhöhten Latenz und einem reduzierten Durchsatz für Ihre Workload führen. Ziehen Sie für solche Workloads das SSL/TLS-Offloading auf [Elastic Load Balancing](#) in Betracht, um die Leistung der Workload zu verbessern, indem Sie den Load Balancer die SSL/TLS-Verschlüsselung und -Entschlüsselung übernehmen lassen, anstatt dies den Backend-Instances zu überlassen. Dies kann dazu beitragen, die CPU-Auslastung der Backend-Instances zu reduzieren, was die Leistung verbessern und die Kapazität erhöhen kann.
- Verwenden Sie den [Network Load Balancer \(NLB\)](#), um Services bereitzustellen, die auf dem UDP-Protokoll basieren (wie die Authentifizierung und Autorisierung, die Protokollierung, DNS, IoT und das Streamen von Medien), um die Leistung und Zuverlässigkeit Ihrer Workload zu verbessern. Der NLB verteilt den eingehenden UDP-Datenverkehr auf mehrere Ziele, sodass Sie Ihre Workload horizontal skalieren, die Kapazität erhöhen und den Overhead eines einzelnen Ziels reduzieren können.
- Für Ihre HPC-Workloads (High Performance Computing) sollten Sie die Verwendung der Funktionalität [Elastic Network Adapter \(ENA\) Express](#) in Betracht ziehen, die das SRD-Protokoll nutzt, um die Leistung des Netzwerks zu verbessern, indem sie eine höhere Bandbreite für einen einzelnen Datenfluss (25 Gbit/s) und eine niedrigere Latenz (99,9 Perzentil) für den Netzwerkverkehr zwischen EC2-Instances bietet.
- Verwenden Sie den [Application Load Balancer \(ALB\)](#), um Ihren gRPC-Datenverkehr (Remote Procedure Calls) zwischen Workload-Komponenten oder zwischen gRPC-Clients und -Services zu routen und ein Load-Balancing durchzuführen. gRPC verwendet das TCP-basierte HTTP/2-Protokoll für den Transport und bietet Vorteile in Bezug auf die Leistung, wie z. B. einen geringeren Netzwerk-Footerprint, Komprimierung, effiziente binäre Serialisierung, Unterstützung zahlreicher Sprachen und bidirektionales Streaming.

Ressourcen

Zugehörige Dokumente:

- [Routing von UDP-Verkehr nach Kubernetes](#)
- [Application Load Balancer](#)

- [EC2: Enhanced Networking unter Linux](#)
- [EC2: Enhanced Networking unter Windows](#)
- [EC2: Platzierungsgruppen](#)
- [Enabling Enhanced Networking with the Elastic Network Adapter \(ENA\) on Linux Instances](#)
- [Network Load Balancer](#)
- [Netzwerkprodukte mit AWS](#)
- [Umstellung auf latenzbasiertes Routing in Amazon Route 53](#)
- [VPC-Endpunkte](#)

Zugehörige Videos:

- [AWS re:Invent 2022 – Scaling network performance on next-gen Amazon Elastic Compute Cloud instances](#)
- [AWS re:Invent 2022 – Grundlagen der Anwendungsnetzwerke](#)

Zugehörige Beispiele:

- [AWS Transit Gateway und skalierbare Sicherheitslösungen](#)
- [Workshops zu AWS-Netzwerken](#)

PERF04-BP06 Wählen Sie den Standort Ihres Workloads basierend auf den Netzwerkanforderungen

Evaluieren Sie Optionen für die Platzierung von Ressourcen, um die Latenz im Netzwerk zu verringern und den Durchsatz zu verbessern und so ein optimales Benutzererlebnis durch kürzere Seitenlade- und Datentransferzeiten zu gewährleisten.

Typische Anti-Muster:

- Sie konsolidieren alle Workload-Ressourcen an einem geografischen Standort.
- Sie haben sich für die Region entschieden, die Ihrem Standort, aber nicht dem Workload-Endbenutzer, am nächsten liegt.

Vorteile der Nutzung dieser bewährten Methode: Die Benutzererfahrung wird stark von der Latenz zwischen dem Benutzer und Ihrer Anwendung beeinflusst. Durch die Verwendung eines geeigneten

AWS-Regionen und AWS privaten globalen Netzwerks können Sie die Latenz reduzieren und Remote-Benutzern ein besseres Benutzererlebnis bieten.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Ressourcen, wie EC2 Amazon-Instances, werden in Availability Zones innerhalb [AWS-Regionen](#), [AWS Local Zones](#) oder [AWS Wavelength](#) Zonen platziert. [AWS Outposts](#) Die Auswahl dieses Standorts beeinflusst die Latenz des Netzwerks und den Durchsatz vom Standort des Benutzers aus. Edge-Services wie [Amazon CloudFront](#) [AWS Global Accelerator](#) können auch zur Verbesserung der Netzwerkleistung verwendet werden, indem sie entweder Inhalte an Edge-Standorten zwischenspeichern oder Benutzern einen optimalen Pfad zur Arbeitslast über das AWS globale Netzwerk bieten.

Amazon EC2 bietet Platzierungsgruppen zum Netzwerken an. Eine Platzierungsgruppe ist eine logische Gruppierung von Instances, um die Latenz zu verringern. Durch die Verwendung von Platzierungsgruppen mit unterstützten Instance-Typen und einem Elastic Network Adapter (ENA) können Workloads an einem 25-Gbit/s-Netzwerk mit geringer Latenz und reduziertem Jitter teilnehmen. Platzierungsgruppen werden für Workloads empfohlen, für die eine niedrige Netzwerklatenz bzw. ein hoher Durchsatz von Vorteil sind.

[Latenzempfindliche Dienste werden an Edge-Standorten über ein AWS globales Netzwerk wie Amazon bereitgestellt. CloudFront](#) Diese Edge-Standorte bieten in der Regel Dienste wie Content Delivery Network (CDN) und Domain Name System (DNS). Da sich diese Dienste an der Peripherie befinden, können Workloads mit geringer Latenz auf Anfragen nach Inhalten oder Lösungen reagieren. Es sind auch geografische Services wie das Geo-Targeting von Inhalten (Bereitstellung unterschiedlicher Inhalte gemäß dem Standort von Endbenutzern) oder die latenzbasierte Weiterleitung von Endbenutzern zur nächsten Region (minimale Latenz) verfügbar.

Verwenden Sie Edge-Services, um die Latenz zu reduzieren und das Caching von Inhalten zu ermöglichen. Konfigurieren Sie die Cachesteuerung DNS sowohl für als auch für HTTP/HTTPS, um den größtmöglichen Nutzen aus diesen Ansätzen zu ziehen.

Implementierungsschritte

- Erfassen Sie Informationen über den an den Netzwerkschnittstellen ein- und ausgehenden IP-Datenverkehr.
 - [Protokollierung von IP-Verkehr mithilfe von VPC Flow Logs](#)

- [Wie wird die Client-IP-Adresse aufbewahrt in AWS Global Accelerator](#)
- Analysieren Sie die Netzwerkzugriffsmuster in Ihrer Workload, um zu ermitteln, wie die Benutzer Ihre Anwendung verwenden.
 - Verwenden Sie Überwachungstools wie [Amazon CloudWatch](#) und [AWS CloudTrail](#), um Daten zu Netzwerkaktivitäten zu sammeln.
 - Analysen Sie die Daten, um das Netzwerkzugriffsmuster zu identifizieren.
- Wählen Sie Regionen für Ihre Workload-Bereitstellung auf der Grundlage der folgenden zentralen Elemente aus:
 - Standort Ihrer Daten: Für datenintensive Anwendungen (wie etwa Big Data oder Machine Learning) sollte der Anwendungscode so nahe wie möglich zu den Daten ausgeführt werden.
 - Standort Ihrer Benutzer: Wählen Sie für benutzerseitige Anwendungen eine Region (oder Regionen) in der Nähe der Benutzer der Workload.
 - Weitere Einschränkungen: Berücksichtigen Sie Einschränkungen wie Kosten und Compliance, wie unter [Relevante Aspekte bei der Wahl einer Region für Ihre Workloads](#) erläutert.
- Verwenden Sie [AWS Local Zones](#), um Workloads wie Video-Rendering auszuführen. Mit Local Zones können Sie von allen Vorteilen profitieren, die sich durch die Platzierung der Datenverarbeitungs- und Speicherressourcen in der Nähe Ihrer Endbenutzer ergeben.
- Verwenden Sie [AWS Outposts](#) für Workloads, die On-Premises verarbeitet werden müssen und die Sie nahtlos mit Ihren restlichen Workloads in AWS ausführen möchten.
- ultra-low-latency Für Anwendungen wie hochauflösendes Live-Videostreaming, High-Fidelity-Audio und Augmented Reality oder Virtual Reality (AR/VR) sind 5G-Geräte erforderlich. Beachten Sie bei solchen Anwendungen Folgendes: [AWS Wavelength](#) AWS Wavelength bettet AWS Rechen- und Speicherdienste in 5G-Netzwerke ein und bietet so eine mobile Edge-Computing-Infrastruktur für die Entwicklung, Bereitstellung und Skalierung von ultra-low-latency Anwendungen.
- Verwenden Sie lokale Zwischenspeicherung oder [AWS -Zwischenspeicherung](#) für häufig genutzte Ressourcen zur Verbesserung der Leistung, zur Verringerung von Datenverschiebungen und zur Reduzierung der Umweltauswirkungen.

Service	Wann sollte dies verwendet werden?
Amazon CloudFront	Wird verwendet, um statische Inhalte wie Bilder, Skripte und Videos sowie dynamische Inhalte wie API Antworten oder Webanwendungen zwischenspeichern.

Service	Wann sollte dies verwendet werden?
Amazon ElastiCache	Verwenden Sie dies für die Zwischenspeicherung von Inhalten für Webanwendungen.
DynamoDB Accelerator	Verwenden Sie dies für die Add-in-Speicher-Beschleunigung für Ihre DynamoDB-Tabellen.

- Nutzen Sie Services, die Ihnen dabei helfen können, Code näher an den Benutzern Ihrer Workload auszuführen:

Service	Wann sollte dies verwendet werden?
Lambda@Edge	Verwenden Sie dies für rechenintensive Anwendungen, die initiiert werden, wenn sich Objekte nicht im Zwischenspeicher befinden.
CloudFront Amazon-Funktionen	Wird für einfache Anwendungsfälle wie HTTP Anfragen oder Antwortmanipulationen verwendet, die durch kurzlebige Funktionen ausgelöst werden können.
AWS IoT Greengrass	Verwenden Sie dies für die Ausführung lokaler Rechenoperationen, Messaging sowie die Datenzwischenspeicherung für verbundene Geräte.

- Einige Anwendungen benötigen feste Zugangspunkte oder eine höhere Leistung. Bei diesen müssen First-Byte-Latenz der Jitter verringert und der Durchsatz erhöht werden. Diese Anwendungen können von Netzwerkdiensten profitieren, die statische Anycast-IP-Adressen und TCP Terminierung an Edge-Standorten bereitstellen. [AWS Global Accelerator](#) kann die Leistung Ihrer Anwendungen um bis zu 60% verbessern und ein schnelles Failover für Architekturen mit mehreren Regionen ermöglichen. AWS Global Accelerator stellt Ihnen statische Anycast-IP-Adressen zur Verfügung, die als fester Einstiegspunkt für Ihre in einer oder mehreren gehosteten Anwendungen dienen. AWS-Regionen Diese IP-Adressen ermöglichen es, dass der Datenverkehr so nah wie möglich an Ihren Benutzern in das AWS globale Netzwerk gelangt. AWS Global Accelerator reduziert die Zeit für den anfänglichen Verbindungsaufbau, indem eine TCP

Verbindung zwischen dem Client und dem AWS Edge-Standort hergestellt wird, der dem Client am nächsten ist. Prüfen Sie die Verwendung von AWS Global Accelerator , um die Leistung Ihrer TCP UDP -Workloads zu verbessern und ein schnelles Failover für Architekturen mit mehreren Regionen zu ermöglichen.

Ressourcen

Zugehörige bewährte Methoden:

- [COST07-BP02 Implementieren Sie Regionen auf der Grundlage der Kosten](#)
- [COST08-BP03 Implementieren Sie Dienste zur Senkung der Datenübertragungskosten](#)
- [REL10-BP01 Stellen Sie den Workload an mehreren Standorten bereit](#)
- [REL10-BP02 Wählen Sie die geeigneten Standorte für Ihre Bereitstellung an mehreren Standorten aus](#)
- [SUS01-BP01 Wählen Sie die Region auf der Grundlage von Geschäftsanforderungen und Nachhaltigkeitszielen](#)
- [SUS02-BP04 Optimieren Sie die geografische Verteilung von Workloads auf der Grundlage ihrer Netzwerkanforderungen](#)
- [SUS04-BP07 Minimierung der Datenbewegung zwischen Netzwerken](#)

Zugehörige Dokumente:

- [AWS Globale Infrastruktur](#)
- [AWS Local Zones und AWS Outposts Auswahl der richtigen Technologie für Ihren Edge-Workload](#)
- [Platzierungsgruppen](#)
- [AWS Local Zones](#)
- [AWS Outposts](#)
- [AWS Wavelength](#)
- [Amazon CloudFront](#)
- [AWS Global Accelerator](#)
- [AWS Direct Connect](#)
- [AWS Site-to-Site VPN](#)
- [Amazon Route 53](#)

Zugehörige Videos:

- [AWS Erklärvideo zu Local Zones](#)
- [AWS Outposts: Overview and How it Works](#)
- [AWS re:Invent 2023 — Eine Migrationsstrategie für Edge-Workloads und lokale Workloads](#)
- [AWS re:Invent 2021 — AWS Outposts: Das Erlebnis vor Ort umsetzen AWS](#)
- [AWS re:Invent 2020: AWS Wavelength: Führen Sie Apps mit extrem niedriger Latenz am 5G-Edge aus](#)
- [AWS re:Invent 2022 — AWS Local Zones: Entwicklung von Anwendungen für eine verteilte Kante](#)
- [AWS re:Invent 2021 — Websites mit niedriger Latenz mit Amazon erstellen CloudFront](#)
- [AWS re:Invent 2022 — Verbessern Sie Leistung und Verfügbarkeit mit AWS Global Accelerator](#)
- [AWS re:Invent 2022 — Bauen Sie Ihr globales Wide Area Network auf mit AWS](#)
- [AWS re:Invent 2020: Globales Verkehrsmanagement mit Amazon Route 53](#)

Zugehörige Beispiele:

- [AWS Global Accelerator Workshop zu kundenspezifischem Routing](#)
- [Verarbeitung von Rewrites und Redirects mit Edge-Funktionen](#)

PERF04-BP07 Optimieren der Netzwerkkonfiguration basierend auf Metriken

Treffen Sie anhand der erfassten und analysierten Daten fundierte Entscheidungen zum Optimieren Ihrer Netzwerkkonfiguration.

Typische Anti-Muster:

- Sie gehen davon aus, dass alle leistungsbezogenen Probleme auf Anwendungen zurückzuführen sind.
- Sie testen die Netzwerkleistung ausschließlich an einem Standort nahe der Stelle, an der Sie die Workload bereitgestellt haben.
- Sie verwenden Standardkonfigurationen für alle Netzwerkservices.
- Sie führen eine Überdimensionierung der Netzwerkkressourcen durch, um eine ausreichende Kapazität zu gewährleisten.

Vorteile der Nutzung dieser bewährten Methode: Das Sammeln der erforderlichen Metriken Ihres AWS-Netzwerks und die Implementierung von Tools zur Überwachung des Netzwerks bieten Ihnen die Möglichkeit, die Leistung des Netzwerks zu ermitteln und die Netzwerkkonfigurationen zu optimieren.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Niedrig

Implementierungsleitfaden

Die Überwachung des Datenverkehrs von und zu VPCs, Subnetzen oder Netzwerkschnittstellen ist für das Verständnis der Nutzung von AWS-Netzwerkressourcen und zur Optimierung von Netzwerkkonfigurationen entscheidend. Mit den folgenden AWS-Networking-Tools können Sie Informationen über die Nutzung des Datenverkehrs, den Netzwerkzugriff und die Protokolle genauer untersuchen.

Implementierungsschritte

- Identifizieren Sie die wichtigsten Leistungsmetriken wie Latenz oder Paketverlust, die erfasst werden müssen. AWS bietet mehrere Tools, die Ihnen bei der Erfassung dieser Metriken helfen können. Mit den folgenden Tools können Sie Informationen über die Nutzung des Datenverkehrs, den Netzwerkzugriff und die Protokolle genauer untersuchen:

AWS-Tool	Aktion
Amazon VPC IP Address Manager .	Nutzen Sie IPAM, um Ihre IP-Adressen für Ihre AWS- und On-Premises-Workloads zu planen, nachzuverfolgen und zu überwachen. Dies ist eine bewährte Methode zur Optimierung der Nutzung und Zuweisung von IP-Adressen.
VPC Flow-Protokolle	Nutzen Sie VPC Flow-Protokolle, um detaillierte Informationen über den Datenverkehr zu und von den Netzwerkschnittstellen in Ihren VPCs zu protokollieren. Mit VPC Flow-Protokollen können Sie restriktive oder freizügige Regeln für Sicherheitsgruppen diagnostizieren und die Richtung

AWS-Tool	Aktion
	des Datenverkehrs zu und von den Netzwerkschnittstellen ermitteln.
AWS Transit Gateway-Flow-Protokolle	Nutzen Sie AWS Transit Gateway-Flow-Protokolle, um Informationen über den IP-Datenverkehr zu und von Ihren Transit-Gateways zu erfassen.
DNS-Abfrageprotokollierung	Protokollieren Sie Informationen über von Route 53 empfangene öffentliche oder private DNS-Abfragen. Mit DNS-Protokollen können Sie DNS-Konfigurationen optimieren, indem Sie die angefragte Domain oder Subdomain bzw. die Route-53-Edge-Standorte, die auf DNS-Abfragen geantwortet haben, nachvollziehen.
Reachability Analyzer	Reachability Analyzer hilft Ihnen, die Erreichbarkeit von Netzwerken zu analysieren und zu debuggen. Reachability Analyzer ist ein Tool zur Konfigurationsanalyse, mit dem Sie Konnektivitätstests zwischen einer Quellressource und einer Zielressource in Ihren VPCs durchführen können. Mit diesem Tool können Sie überprüfen, ob Ihre Netzwerkkonfiguration der geplanten Konnektivität entspricht.

AWS-Tool	Aktion
Network Access Analyzer	Mit Network Access Analyzer können Sie den Netzwerkzugriff auf die eigenen Ressourcen analysieren. Mit Network Access Analyzer können Sie Ihre Anforderungen an den Netzwerkzugriff spezifizieren und potenzielle Netzwerkpfade identifizieren, die Ihren Anforderungen nicht entsprechen. Indem Sie Ihre entsprechende Netzwerkkonfiguration optimieren, können Sie den Zustand Ihres Netzwerks nachvollziehen und überprüfen sowie belegen, dass Ihr AWS-Netzwerk Ihre Complianceanforderungen erfüllt.
Amazon CloudWatch	Verwenden Sie Amazon CloudWatch und aktivieren Sie geeignete Metriken für Netzwerkoptionen. Stellen Sie sicher, dass Sie die richtige Netzwerk-Metrik für Ihre Workload auswählen. Sie können zum Beispiel Metriken für die VPC-Netzwerkadressennutzung, VPC-NAT-Gateways, AWS Transit Gateway, VPN-Tunnel, AWS Network Firewall, Elastic Load Balancing und AWS Direct Connect aktivieren. Die kontinuierliche Überwachung von Metriken ist eine gute Vorgehensweise, um den Status und die Nutzung Ihres Netzwerks zu beobachten und nachzuvollziehen. Sie hilft Ihnen, die Netzwerkkonfiguration auf der Basis Ihrer Beobachtungen zu optimieren.

AWS-Tool	Aktion
AWS Network Manager	Mit AWS Network Manager können Sie die Echtzeit- und historische Leistung des globalen AWS-Netzwerks für Betriebs- und Planungszwecke überwachen. Network Manager bietet aggregierte Netzwerklatenz zwischen AWS-Regionen und Availability Zones sowie innerhalb jeder Availability Zone, wodurch Sie besser verstehen können, wie Ihre Anwendungsleistung mit der Leistung des zugrunde liegenden AWS-Netzwerks zusammenhängt.
Amazon CloudWatch RUM	Verwenden Sie Amazon CloudWatch RUM, um die Metriken zu erfassen, die Ihnen die Erkenntnisse liefern, mit denen Sie die Benutzererfahrung identifizieren, verstehen und verbessern können.

- Identifizieren Sie mithilfe von VPC und AWS Transit Gateway Flow Logs Top-Talker und Muster des Anwendungsdatenverkehrs.
- Bewerten und optimieren Sie Ihre aktuelle Netzwerkarchitektur, einschließlich VPCs, Subnetze und Routing. Sie können beispielsweise bewerten, wie unterschiedliches VPC-Peering oder AWS Transit Gateway Ihnen helfen können, das Netzwerk in Ihrer Architektur zu verbessern.
- Bewerten Sie die Routingpfade in Ihrem Netzwerk, um sicherzustellen, dass immer der kürzeste Pfad zwischen Zielen verwendet wird. Network Access Analyzer kann Ihnen dabei helfen.

Ressourcen

Zugehörige Dokumente:

- [Öffentliche DNS-Abfrageprotokollierung](#)
- [Was ist IPAM?](#)
- [What is Reachability Analyzer?](#)
- [What is Network Access Analyzer?](#)

- [CloudWatch-Metriken für Ihre VPCs](#)
- [Die Leistung optimieren und die Kosten für Netzwerkanalytik mit VPC-Ablaufprotokollen im Apache-Parquet-Format senken](#)
- [Überwachen von globalen und Kernnetzwerken mit Amazon CloudWatch-Metriken](#)
- [Kontinuierliches Überwachen von Netzwerkdatenverkehr und -ressourcen](#)

Zugehörige Videos:

- [AWS re:Invent 2023 – Leitfaden für Entwickler von Cloud-Netzwerken](#)
- [AWS re:Invent 2023 – Sind Sie bereit für Neues? Gestaltung von Netzwerken für Wachstum und Flexibilität](#)
- [AWS re:Invent 2023 – Erweiterte VPC-Designs und neue Funktionen](#)
- [AWS re:Invent 2022 – Ausführliche Beschreibung der AWS-Netzwerkinfrastruktur](#)
- [AWS re:Invent 2020 – Bewährte Methoden für Netzwerke und Tipps für das AWSWell-Architected Framework](#)
- [AWS re:Invent 2020 – Überwachen des Netzwerkdatenverkehrs und Fehlerbehebung](#)

Zugehörige Beispiele:

- [Workshops zu AWS-Netzwerken](#)
- [AWS Network Monitoring](#)
- [Beobachten und Diagnostizieren Ihres Netzwerks in AWS](#)
- [Finden und Beheben von Netzwerkfehlkonfigurationen in AWS](#)

Prozess und Kultur

Fragen

- [LEIST 5. Wie tragen Ihre Unternehmenspraktiken und Ihre Unternehmenskultur zur Leistungseffizienz Ihrer Workload bei?](#)

LEIST 5. Wie tragen Ihre Unternehmenspraktiken und Ihre Unternehmenskultur zur Leistungseffizienz Ihrer Workload bei?

Bei der Architektur von Workloads gibt es Prinzipien und Methoden, die Sie übernehmen können, um effiziente und leistungsstarke Cloud-Workloads besser zu betreiben. Um eine Kultur zu schaffen, die die Leistungseffizienz von Cloud-Workloads fördert, sollten Sie diese Schlüsselprinzipien und -praktiken berücksichtigen:

Best Practices

- [PERF05-BP01 Festlegen wichtiger Leistungskennzahlen \(KPIs\) zum Messen des Zustands und der Leistung der Workload](#)
- [PERF05-BP02 Verwenden von Überwachungslösungen, um Bereiche mit kritischem Leistungsbedarf zu identifizieren](#)
- [PERF05-BP03 Definieren eines Prozesses zum Verbessern der Workload-Leistung](#)
- [PERF05-BP04 Belastungstest Ihr Workload](#)
- [PERF05-BP05 Verwenden von Automatisierung zur proaktiven Behebung leistungsbezogener Probleme](#)
- [PERF05-BP06 Behalten Sie Ihren Workload und Ihre Services up-to-date](#)
- [PERF05-BP07 Regelmäßiges Überprüfen von Metriken](#)

PERF05-BP01 Festlegen wichtiger Leistungskennzahlen (KPIs) zum Messen des Zustands und der Leistung der Workload

Identifizieren Sie die KPIs, die die Workload-Leistung quantitativ und qualitativ messen. Mithilfe von KPIs können Sie den Zustand und die Leistung einer Workload im Zusammenhang mit einem Geschäftsziel messen.

Typische Anti-Muster:

- Sie überwachen nur Metriken auf Systemebene, um Erkenntnisse über Ihre Workload zu gewinnen, und verstehen den geschäftlichen Einfluss dieser Metriken nicht.
- Sie gehen davon aus, dass Ihre KPIs bereits als standardmäßige Metrikdaten veröffentlicht und geteilt werden.
- Sie definieren keinen quantitativen, messbaren KPI.
- Sie richten KPIs nicht an Geschäftszielen oder -strategien aus.

Vorteile der Nutzung dieser bewährten Methode: Die Identifizierung spezifischer KPIs, die den Zustand und die Leistung der Workload widerspiegeln, hilft Teams dabei, sich auf ihre Prioritäten zu konzentrieren und erfolgreiche Geschäftsergebnisse zu definieren. Das Teilen dieser Metriken mit allen Abteilungen bietet Sichtbarkeit und die Ausrichtung an Grenzwerten, Erwartungen und Geschäftsauswirkungen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

KPIs helfen Business- und Entwicklungsteams, das Messen von Zielen und Strategien abzustimmen und festzustellen, wie diese Faktoren gemeinsam zu Geschäftsergebnissen beitragen. Beispielsweise könnte eine Website-Workload die Ladezeit der Seite als Indikator für die Gesamtleistung heranziehen. Diese Metrik wäre einer von mehreren Datenpunkten, mit denen das Benutzererlebnis gemessen wird. Zusätzlich zum Ermitteln der Grenzwerte für Seitenladezeiten sollten Sie das gewünschte Resultat dokumentieren bzw. das Geschäftsrisiko, wenn die ideale Leistung nicht erreicht wird. Die lange Ladezeit einer Seite betrifft Ihre Endbenutzer direkt, verringert die Bewertung ihres Benutzererlebnisses und kann zu einem Verlust von Kunden führen. Kombinieren Sie beim Definieren Ihrer KPI-Grenzwerte die Benchmarks der Branche und die Erwartungen Ihrer Endbenutzer. Beispielsweise, wenn die aktuelle Benchmark der Branche das Laden einer Webseite innerhalb von zwei Sekunden ist, Ihre Endbenutzer aber erwarten, dass eine Webseite innerhalb von einer Sekunde geladen wird, sollten Sie beim Einrichten des KPI beide Datenpunkte in Betracht ziehen.

Ihr Team muss Ihre Workload-KPIs mithilfe von detaillierten Echtzeitdaten und historischen Daten als Referenz evaluieren und Dashboards erstellen, die Metrikberechnungen für Ihre KPI-Daten durchführen, um Einblicke in Betrieb und Auslastung zu erhalten. KPIs sollten dokumentiert werden und Grenzwerte enthalten, die Geschäftsziele und -strategien unterstützen, und sie sollten den Metriken zugeordnet sein, die überwacht werden. KPIs sollten erneut aufgegriffen werden, wenn sich Geschäftsziele, Strategien oder Anforderungen von Endbenutzern ändern.

Implementierungsschritte

- **Identifizieren von Stakeholdern:** Identifizieren und dokumentieren Sie wichtige Geschäfts-Stakeholder, einschließlich Entwicklungs- und Betriebsteams.
- **Definition von Zielen:** Arbeiten Sie mit diesen Stakeholdern zusammen, um die Ziele Ihrer Workload zu definieren und zu dokumentieren. Berücksichtigen Sie die kritischen Leistungsaspekte Ihrer Workloads, wie Durchsatz, Reaktionszeit und Kosten, sowie Geschäftsziele wie die Benutzerzufriedenheit.

- Ansehen von in der Branche bewährter Methoden: Sehen Sie sich in der Branche bewährte Methoden an, um relevante KPIs zu identifizieren, die auf Ihre Workload-Ziele abgestimmt sind.
- Identifizieren von Metriken: Identifizieren Sie Metriken, die auf Ihre Workload-Ziele abgestimmt sind und Ihnen helfen können, Leistung und Geschäftsziele zu messen. Richten Sie KPIs basierend auf diesen Metriken ein. Beispiele für Metriken sind Messungen wie die durchschnittliche Reaktionszeit oder die Anzahl gleichzeitiger Benutzer.
- Definition und Dokumentation von KPIs: Verwenden Sie in der Branche bewährte Methoden und Ihre Workload-Ziele, um Ziele für Ihren Workload-KPI festzulegen. Verwenden Sie diese Informationen, um KPI-Schwellenwerte für Schweregrad oder Alarmebene festzulegen. Identifizieren und dokumentieren Sie das Risiko und die Auswirkungen, wenn ein KPI nicht erreicht wird.
- Implementieren der Überwachung: Verwenden Sie Überwachungstools wie [Amazon CloudWatch](#) oder [AWS Config](#), um Metriken zu sammeln und KPIs zu messen.
- Visuelle Kommunikation von KPIs: Verwenden Sie Dashboard-Tools wie [Amazon Quick](#), um KPIs zu visualisieren und mit den Stakeholdern zu kommunizieren.
- Analysieren und Optimieren: Überprüfen und analysieren Sie regelmäßig die KPIs, um Bereiche Ihres Workloads zu identifizieren, die verbessert werden müssen. Arbeiten Sie mit den Stakeholdern zusammen, um diese Verbesserungen umzusetzen.
- Wiederaufgreifen und verfeinern: Überprüfen Sie regelmäßig die Metriken und KPIs, um ihre Effektivität zu bewerten, insbesondere wenn sich die Geschäftsziele oder die Workload-Leistung ändern.

Ressourcen

Zugehörige Dokumente:

- [CloudWatch-Dokumentation](#)
- [Überwachung, Protokollierung und Leistung von AWS Partners](#)
- [AWS-Tools zur Beobachtbarkeit](#)
- [The Importance of Key Performance Indicators \(KPIs\) for Large-Scale Cloud Migrations](#)
- [How to track your cost optimization KPIs with the KPI Dashboard](#)
- [X-Ray-Dokumentation](#)
- [Verwenden von Amazon-CloudWatch-Dashboards](#)
- [Quick-KPIs](#)

Zugehörige Videos:

- [AWS re:Invent 2023 – Optimieren der Kosten und Leistung sowie Verfolgen der Fortschritte bei der Schadensbegrenzung](#)
- [AWS re:Invent 2023 – Manage resource lifecycle events at scale with AWS Health](#)
- [AWS re:Invent 2023 – Performance & efficiency at Pinterest: Optimizing the latest instances](#)
- [AWS re:Invent 2022 – AWS-Optimierung: Umsetzbare Schritte für sofortige Ergebnisse](#)
- [AWS re:Invent 2023 – Aufbau einer effektiven Beobachtbarkeitsstrategie](#)
- [AWS Summit SF 2022 – Full-stack observability and application monitoring with AWS](#)
- [AWS re:Invent 2023 – Scaling on AWS for the first 10 million users](#)
- [AWS re:Invent 2022 – How Amazon uses better metrics for improved website performance](#)
- [Creating an Effective Metrics Strategy for Your Business | AWS Events](#)

Zugehörige Beispiele:

- [Erstellen eines Dashboards mit Quick](#)

PERF05-BP02 Verwenden von Überwachungslösungen, um Bereiche mit kritischem Leistungsbedarf zu identifizieren

Ermitteln Sie die Bereiche, in denen sich durch Steigern der Workload-Leistung positive Auswirkungen auf die Effizienz oder den Kundenkomfort realisieren lassen. Beispiel: Eine Website mit zahlreichen Kundeninteraktionen kann von der Nutzung von Edge-Services profitieren, indem Inhalte näher bei den Kunden bereitgestellt werden.

Typische Anti-Muster:

- Sie gehen davon aus, dass standardmäßige Datenverarbeitungsmetriken wie CPU-Auslastung oder Arbeitsspeicherdruck ausreichen, um Leistungsprobleme zu erfassen.
- Sie verwenden nur die Standardmetriken, die von der Überwachungssoftware Ihrer Wahl aufgezeichnet wurden.
- Sie überprüfen Metriken nur dann, wenn ein Problem vorliegt.

Vorteile der Nutzung dieser bewährten Methode: Das eingehende Verständnis kritischer Bereiche hilft Workload-Eigentümern dabei, KPIs zu überwachen und Verbesserungen mit größeren Auswirkungen zu priorisieren.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Richten Sie durchgehende Nachverfolgung ein, um Datenverkehrsmuster, Latenz und kritische Leistungsbereiche zu identifizieren. Überwachen Sie Ihre Datenzugriffsmuster auf langsame Abfragen oder schlecht fragmentierte und partitionierte Daten. Identifizieren Sie problematische Workload-Bereiche mithilfe von Lasttests oder -überwachung.

Erhöhen Sie die Leistungseffizienz durch eingehendes Verständnis Ihrer Architektur, der Datenverkehrs- und der Datenzugriffsmuster und identifizieren Sie Ihre Latenz- und Verarbeitungszeiten. Identifizieren Sie potenzielle Engpässe, die sich bei zunehmenden Workloads auf den Kundenkomfort auswirken könnten. Nachdem Sie diese Bereiche untersucht haben, sollten Sie prüfen, welche Lösung Sie nutzen können, um diese Leistungsprobleme zu beseitigen.

Implementierungsschritte

- Richten Sie durchgehende Überwachung ein, um alle Workload-Komponenten und -Metriken zu erfassen. Hier finden Sie Beispiele für Überwachungslösungen in AWS.

Service	Aktion
Amazon CloudWatch Real-User Monitoring (RUM)	Zum Erfassen von Metriken zur Anwendungseistung aus realen clientseitigen und Frontend-Sitzungen.
AWS X-Ray	Zum Verfolgen des Datenverkehrs durch die Anwendungsebenen und zum Identifizieren der Latenz zwischen Komponenten und Abhängigkeiten. Verwenden Sie X-Ray-Service-Zuordnungen, um Beziehungen und Latenz zwischen Workload-Komponenten zu erkennen.

Service	Aktion
Amazon Relational Database Service – Performance Insights	Zum Anzeigen von Metriken zur Datenbankleistung und zum Identifizieren von Möglichkeiten zur Leistungsverbesserung.
Amazon RDS Enhanced Monitoring	Zum Anzeigen von Datenbank-BS-Leistungsmetriken.
Amazon DevOps Guru	Zum Erkennen ungewöhnlicher Betriebsmuster, damit Sie betriebliche Probleme identifizieren können, bevor sie sich auf Ihre Kunden auswirken.

- Führen Sie Tests durch, um Metriken zu generieren sowie Datenverkehrsmuster, Engpässe und kritische Leistungsbereiche zu identifizieren. Hier finden Sie einige Beispiele zum Durchführen von Tests:
 - Richten Sie [CloudWatch Synthetic Canaries](#) zur programmgesteuerten Nachahmung browserbasierter Benutzeraktivitäten mit Linux-Cron-Aufträgen oder Rate-Ausdrücken und zum Erhalt konsistenter Metriken in Zeitverlauf ein.
 - Verwenden Sie die Lösung für [verteilte Lasttests auf AWS](#), um Spitzendatenverkehr zu generieren oder Workloads mit der erwarteten Wachstumsrate zu testen.
- Evaluieren Sie die Metriken und die Telemetriedaten, um Ihre kritischen Leistungsbereiche zu identifizieren. Prüfen Sie diese Bereiche zusammen mit Ihrem Team und besprechen Sie Überwachung und Lösung zur Vermeidung von Engpässen.
- Experimentieren Sie mit Leistungsverbesserungen und messen Sie diese Änderungen anhand von Daten. Beispielsweise können Sie [CloudWatch Evidently](#) verwenden, um neue Verbesserungen und Leistungsauswirkungen auf Ihrer Workload zu testen.

Ressourcen

Zugehörige Dokumente:

- [What's new in AWS Observability at re:Invent 2023](#)
- [Amazon Builders' Library](#)
- [X-Ray-Dokumentation](#)
- [Amazon CloudWatch RUM](#)

- [Amazon DevOps Guru](#)

Zugehörige Videos:

- [AWS re:Invent 2023 – \[LAUNCH\] Application monitoring for modern workloads](#)
- [AWS re:Invent 2023 – Implementierung der Anwendungsbeobachtbarkeit](#)
- [AWS re:Invent 2023 – Aufbau einer effektiven Beobachtbarkeitsstrategie](#)
- [AWS Summit SF 2022 – Full-stack observability and application monitoring with AWS](#)
- [AWS re:Invent 2022 – AWS-Optimierung: Umsetzbare Schritte für sofortige Ergebnisse](#)
- [AWS re:Invent 2022 – Die Amazon Builders' Library: 25 Jahre operative Exzellenz von Amazon](#)
- [AWS re:Invent 2022 – How Amazon uses better metrics for improved website performance](#)
- [Visual Monitoring of Applications with Amazon CloudWatch Synthetics](#)

Zugehörige Beispiele:

- [Messen der Seitenladezeit mit Amazon CloudWatch Synthetics](#)
- [Webclient von Amazon CloudWatch RUM](#)
- [X-Ray-SDK für Python](#)
- [Distributed Load Testing auf AWS](#)

PERF05-BP03 Definieren eines Prozesses zum Verbessern der Workload-Leistung

Definieren Sie einen Prozess, mit dem sich neu verfügbare Services, Designmuster, Ressourcentypen und Konfigurationen bewerten lassen. Führen Sie beispielsweise vorhandene Leistungstests für neue Instance-Angebote durch, um zu ermitteln, welche Verbesserungen sich für Ihre Workload ergeben.

Typische Anti-Muster:

- Sie gehen davon aus, dass Ihre aktuelle Architektur statisch ist und im Laufe der Zeit nicht aktualisiert wird.
- Sie führen im Laufe der Zeit Änderungen an der Architektur ein, ohne sie begründen.

Vorteile der Nutzung dieser bewährten Methode: Durch einen definierten Prozess zum Ändern der Architektur erhalten Sie die Möglichkeit, die gesammelten Daten langfristig in die Gestaltung Ihrer Workload einfließen zu lassen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Für Ihre Workload gibt es einige wesentliche Einschränkungen. Dokumentieren Sie diese, damit Sie besser einschätzen können, durch welche Art von Innovation die Leistung Ihrer Workload gesteigert werden könnte. Ziehen Sie diese Informationen heran, wenn Sie von neuen verfügbaren Services oder Technologien erfahren, um Möglichkeiten zur Beseitigung von Einschränkungen oder Engpässen zu identifizieren.

Identifizieren Sie wesentliche Leistungseinschränkungen für Ihre Workload. Dokumentieren Sie die Leistungseinschränkungen Ihrer Workload, damit Sie besser einschätzen können, durch welche Art von Innovation die Leistung Ihrer Workload ggf. gesteigert werden kann.

Implementierungsschritte

- Identifizieren von KPIs: Identifizieren Sie Ihre Workload-Leistungs-KPIs, wie unter [PERF05-BP01 Festlegen wichtiger Leistungskennzahlen \(KPIs\) zum Messen des Zustands und der Leistung der Workload](#) zur Ermittlung Ihrer Workload-Baseline beschrieben.
- Implementieren der Überwachung: Verwenden Sie [AWS-Beobachtbarkeitstools](#), um Leistungskennzahlen zu erfassen und KPIs zu messen.
- Durchführen einer Analyse: Führen Sie eine eingehende Analyse durch, um die Bereiche (wie Konfiguration und Anwendungscode) in Ihrer Workload zu identifizieren, die leistungsschwach sind, wie beschrieben unter [PERF05-BP02 Verwenden von Überwachungslösungen, um Bereiche mit kritischem Leistungsbedarf zu identifizieren](#). Verwenden Sie Analyse- und Leistungstools, um die Strategie zur Leistungsverbesserung zu identifizieren.
- Validieren von Verbesserungen: Verwenden Sie Sandbox- oder Vorproduktionsumgebungen, um die Effektivität von Verbesserungsstrategien zu überprüfen.
- Implementieren von Änderungen: Implementieren Sie die Änderungen in der Produktion und überwachen Sie kontinuierlich die Leistung der Workload. Dokumentieren Sie die Verbesserungen und teilen Sie die Änderungen den Stakeholdern mit.
- Wiederaufgreifen und verfeinern: Überprüfen Sie regelmäßig Ihren Prozess zur Leistungsverbesserung, um Bereiche mit Verbesserungspotenzial zu identifizieren.

Ressourcen

Zugehörige Dokumente:

- [AWS Blog](#)
- [Neuerungen bei AWS](#)
- [AWS Skill Builder](#)

Zugehörige Videos:

- [AWS re:Invent 2022 – Bereitstellung nachhaltiger, leistungsstarker Architekturen](#)
- [AWS re:Invent 2023 – Optimieren der Kosten und Leistung sowie Verfolgen der Fortschritte bei der Schadensbegrenzung](#)
- [AWS re:Invent 2022 – AWS-Optimierung: Umsetzbare Schritte für sofortige Ergebnisse](#)
- [AWS re:Invent 2022 – Optimize your AWS workloads with best-practice guidance](#)

Zugehörige Beispiele:

- [AWS GitHub](#)

PERF05-BP04 Belastungstest Ihr Workload

Führen Sie für die Workload Lasttests durch, um sicherzustellen, dass sie die Produktionslast bewältigen kann, und identifizieren Sie Leistungsengpässe.

Typische Anti-Muster:

- Sie führen Lasttests für einzelne Teile der Workload durch, aber nicht für die gesamte Workload.
- Sie führen Lasttests in einer Infrastruktur durch, die sich von Ihrer Produktionsumgebung unterscheidet.
- Sie führen Lasttests nur für die erwartete Last durch und nicht für noch größere Lasten, um mögliche künftige Probleme besser vorherzusehen.
- Sie führen Belastungstests durch, ohne die [Amazon EC2 Testing Policy zu lesen](#) und ein Formular zur Einreichung von simulierten Ereignissen einzureichen. Dies führt dazu, dass Ihr Test nicht ausgeführt werden kann, da er wie ein denial-of-service Ereignis aussieht.

Vorteile der Nutzung dieser bewährten Methode: Die Messung der Leistung im Rahmen eines Lasttests gibt Aufschluss darüber, wo bei zunehmender Last mit Auswirkungen zu rechnen ist. Auf diese Weise können Sie erforderliche Änderungen vorhersehen, bevor sie sich auf Ihre Workload auswirken.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Niedrig

Implementierungsleitfaden

Lasttests in der Cloud sind ein Prozess zur Messung der Leistung einer Cloud-Workload unter realistischen Bedingungen mit erwarteter Benutzerlast. Dieser Prozess beinhaltet die Bereitstellung einer produktionsähnlichen Cloud-Umgebung, die Verwendung von Lasttest-Tools zur Lastgenerierung und die Analyse von Metriken, um die Fähigkeit Ihrer Workload zu bewerten, mit einer realistischen Last umzugehen. Verwenden Sie für Lasttests synthetische oder bereinigte Produktionsdaten und entfernen Sie sensible oder personenbezogene Informationen. Führen Sie automatisch Lasttests als Teil Ihrer Bereitstellungspipeline durch und vergleichen Sie die Ergebnisse mit vordefinierten Grenzwerten KPIs und Schwellenwerten. Dieser Prozess hilft Ihnen dabei, die erforderliche Leistung weiterhin zu erreichen.

Implementierungsschritte

- **Testziele definieren:** Identifizieren Sie die Leistungsaspekte Ihrer Workload, die Sie bewerten möchten, wie Durchsatz und Reaktionszeit.
- **Testtool auswählen:** Wählen und konfigurieren Sie das Lasttest-Tool, das zu Ihrer Workload passt.
- **Umgebung einrichten:** Richten Sie die Testumgebung auf der Grundlage Ihrer Produktionsumgebung ein. Sie können AWS Services verwenden, um Produktionsumgebungen auszuführen und Ihre Architektur zu testen.
- **Implementieren Sie Überwachung:** Verwenden Sie Überwachungstools wie [Amazon CloudWatch](#), um Kennzahlen für alle Ressourcen in Ihrer Architektur zu sammeln. Sie können auch benutzerdefinierte Metriken erfassen und veröffentlichen.
- **Szenarien definieren:** Definieren Sie die Szenarien und Parameter der Lasttests (wie Testdauer und Anzahl der Benutzer).
- **Belastungstests durchführen:** Führen Sie Testszenarien in großem Maßstab durch. Nutzen Sie das AWS Cloud , um Ihren Workload zu testen und herauszufinden, wo er nicht skaliert werden kann oder ob er nicht linear skaliert wird. Nutzen Sie beispielsweise Spot Instances, um kostengünstig Lasten zu erzeugen und Engpässe zu identifizieren, bevor diese in der Produktionsumgebung auftreten.

- **Testergebnisse analysieren:** Analysieren Sie die Ergebnisse, um Leistungsengpässe und verbesserungswürdige Bereiche zu identifizieren.
- **Erkenntnisse dokumentieren und teilen:** Dokumentieren Sie Erkenntnisse und Empfehlungen und erstellen Sie Berichte darüber. Teilen Sie diese Informationen mit Stakeholdern, um ihnen zu helfen, fundierte Entscheidungen über Strategien zur Leistungsoptimierung zu treffen.
- **Kontinuierliche Iteration:** Lasttests sollten in regelmäßigen Abständen durchgeführt werden, insbesondere nach einer Systemänderung oder einem Systemupdate.

Ressourcen

Zugehörige Dokumente:

- [Amazon CloudWatch RUM](#)
- [Amazon CloudWatch Synthetics](#)
- [Testen verteilter Lasten auf AWS](#)

Zugehörige Videos:

- [AWS Summit ANZ 2023: Mit AWS Distributed Load Testing mit Zuversicht beschleunigen](#)
- [AWS re:Invent 2022 — Weitere Skalierung AWS für Ihre ersten 10 Millionen Benutzer](#)
- [Problemlösung mit AWS Lösungen: Testen verteilter Lasten](#)
- [AWS re:Invent 2021 — Optimieren Sie Anwendungen mithilfe von Erkenntnissen für Endbenutzer mit Amazon CloudWatch RUM](#)
- [Demo von Amazon CloudWatch Synthetics](#)

Zugehörige Beispiele:

- [Verteilte Belastungstests auf AWS](#)

PERF05-BP05 Verwenden von Automatisierung zur proaktiven Behebung leistungsbezogener Probleme

Verwenden Sie wichtige Leistungskennzahlen (KPIs) in Kombination mit Überwachungs- und Warnsystemen, um eine proaktive Behandlung leistungsbezogener Probleme zu ermöglichen.

Typische Anti-Muster:

- Sie geben dem Betriebspersonal nur die Möglichkeit, betriebliche Änderungen an der Workload vorzunehmen.
- Sie lassen alle Alarme ohne proaktive Behebung zum Operations-Team filtern.

Vorteile der Nutzung dieser bewährten Methode: Die proaktive Behebung von Alarmaktionen ermöglicht es dem Support-Personal, sich auf die Elemente zu konzentrieren, die nicht automatisch umsetzbar sind. Dies hilft dem Betriebspersonal, alle Alarme zu bewältigen, ohne überfordert zu werden, und sich stattdessen auf die kritischen Alarme zu konzentrieren.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Niedrig

Implementierungsleitfaden

Verwenden Sie Alarme, um automatisierte Aktionen auszulösen und auf diese Weise Probleme nach Möglichkeit zu beheben. Leiten Sie den Alarm an die Personen weiter, die die richtigen Maßnahmen einleiten können, falls keine automatisierte Reaktion möglich ist. Beispielsweise können Sie ein System nutzen, das erwartete Werte wichtiger Leistungskennzahlen (KPIs) prognostiziert und bei Überschreiten bestimmter Schwellenwerte einen Alarm ausgibt. Denkbar ist auch ein Tool, das Bereitstellungen automatisch anhält oder zurücksetzt, wenn sich KPIs außerhalb der erwarteten Werte befinden.

Implementieren Sie Prozesse, die Ihnen Einblick in die Leistung gewähren, während Ihre Workload ausgeführt wird. Entwickeln Sie Dashboards für die Überwachung und legen Sie Leistungsnormen in Form von Grundwerten fest, um zu bestimmen, ob die Workload optimal funktioniert.

Implementierungsschritte

- Identifizierung eines Fehlerbehebungs-Workflows: Identifizieren und verstehen Sie das Leistungsproblem, das automatisch behoben werden kann. Verwenden Sie Überwachungslösungen von AWS wie [Amazon CloudWatch](#) oder AWS X-Ray, damit Sie die Ursache des Problems besser verstehen.
- Definition des Automatisierungsprozesses: Erstellen Sie einen schrittweisen Prozess zur Behebung des Problems, mit dem das Problem automatisch behoben werden kann.
- Konfiguration des Initiierungsereignisses: Konfigurieren Sie das Ereignis so, dass der Prozess zur Mängelbeseitigung automatisch eingeleitet wird. Sie können beispielsweise einen Auslöser definieren, der eine Instance automatisch neu startet, wenn sie einen bestimmten Schwellenwert für die CPU-Auslastung erreicht.

- Automatisierung der Mängelbeseitigung: Nutzen Sie AWS-Services und -Technologien, um den Prozess zur Mängelbeseitigung zu automatisieren. Beispielsweise bietet [AWS Systems Manager Automation](#) eine sichere und skalierbare Möglichkeit, den Prozess zur Mängelbeseitigung zu automatisieren. Achten Sie darauf, die Selbstheilungslogik zu verwenden, um Änderungen rückgängig zu machen, wenn das Problem nicht gelöst wurde.
- Testen des Workflows: Testen Sie den automatisierten Prozess zur Mängelbeseitigung in einer Vorproduktionsumgebung.
- Implementieren des Workflows: Implementieren Sie die automatisierte Mängelbeseitigung in der Produktionsumgebung.
- Entwicklung eines Playbooks: Entwickeln und dokumentieren Sie ein Playbook, in dem die Schritte für den Mängelbeseitigungsplan beschrieben werden, einschließlich der Initiierungseignisse, der Mängelbeseitigungslogik und der ergriffenen Maßnahmen. Stellen Sie sicher, dass alle Stakeholder entsprechend geschult werden, damit sie effektiv auf automatisierte Mängelbeseitigungseignisse reagieren können.
- Überprüfen und verfeinern: Bewerten Sie regelmäßig die Effektivität des automatisierten Mängelbeseitigungsworkflows. Passen Sie bei Bedarf die Initiierungseignisse und die Mängelbeseitigungslogik an.

Ressourcen

Zugehörige Dokumente:

- [CloudWatch-Dokumentation](#)
- [Überwachung, Protokollierung und Leistung von AWS Partner Network-Partnern](#)
- [X-Ray-Dokumentation](#)
- [Using Alarms and Alarm Actions in CloudWatch](#)
- [Build a Cloud Automation Practice for Operational Excellence: Best Practices from AWS Managed Services](#)
- [Automate your Amazon Redshift performance tuning with automatic table optimization](#)

Zugehörige Videos:

- [AWS re:Invent 2023 – Strategies for automated scaling, remediation, and smart self-healing](#)
- [AWS re:Invent 2023 – \[LAUNCH\] Application monitoring for modern workloads](#)
- [AWS re:Invent 2023 – Implementierung der Anwendungsbeobachtbarkeit](#)

- [AWS re:Invent 2021 – Intelligently automating cloud operations](#)
- [AWS re:Invent 2022 – Setting up controls at scale in your AWS environment](#)
- [AWS re:Invent 2022 – Automating patch management and compliance using AWS](#)
- [AWS re:Invent 2022 – How Amazon uses better metrics for improved website performance](#)
- [AWS re:Invent 2023 – Take a load off: Diagnose & resolve performance issues with Amazon RDS](#)
- [AWS re:Invent 2021 –{New Launch} Automatically detect and resolve issues with Amazon DevOps Guru](#)
- [AWS re:Invent 2023 – Centralize your operations](#)

Zugehörige Beispiele:

- [CloudWatch Logs – Konfigurieren von Alarmen](#)

PERF05-BP06 Behalten Sie Ihren Workload und Ihre Services up-to-date

Bleiben Sie up-to-date auf der Suche nach neuen Cloud-Diensten und -Funktionen, um effiziente Funktionen einzuführen, Probleme zu beheben und die allgemeine Leistungseffizienz Ihres Workloads zu verbessern.

Typische Anti-Muster:

- Sie gehen davon aus, dass Ihre aktuelle Architektur statisch ist und im Laufe der Zeit nicht aktualisiert wird.
- Sie haben keine Systeme oder regelmäßigen Besprechungen zur Prüfung, ob aktualisierte Software und Pakete mit Ihrer Workload kompatibel sind.

Vorteile der Einführung dieser bewährten Methode: Durch die Einrichtung eines Prozesses up-to-date zur Beibehaltung neuer Services und Angebote können Sie neue Funktionen und Fähigkeiten einführen, Probleme lösen und die Workload-Leistung verbessern.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Niedrig

Implementierungsleitfaden

Evaluieren Sie Möglichkeiten zur Verbesserung der Leistung, wenn neue Services, Entwurfsmuster und Produktfunktionen verfügbar sind. Ermitteln Sie anhand von Bewertungen, internen Diskussionen

oder externen Analysen, wie sich diese neuen Optionen positiv auf die Leistung oder Effizienz der Workload auswirken können. Definieren Sie einen Prozess zum Bewerten von Updates, neuen Features und Services, die für Ihre Workload relevant sind. Erstellen Sie beispielsweise Machbarkeitsstudien, die auf neuen Technologien aufbauen, oder beraten Sie sich mit einer internen Gruppe. Führen Sie beim Ausprobieren neuer Ideen oder Services Leistungstests durch, um die Auswirkungen auf die Leistung der Workload zu messen.

Implementierungsschritte

- **Inventarisierung Ihrer Workload:** Inventarisieren Sie Ihre Workload-Software und -Architektur und identifizieren Sie Komponenten, die aktualisiert werden müssen.
- **Identifizierung von Quellen für Updates:** Identifizieren Sie Quellen für Neuigkeiten und Updates im Zusammenhang mit Ihren Workload-Komponenten. Sie können beispielsweise den [AWS Blog What's New at](#) abonnieren, um die Produkte zu finden, die zu Ihrer Workload-Komponente passen. Sie können den RSS Feed abonnieren oder Ihre [E-Mail-Abonnements](#) verwalten.
- **Definition eines Aktualisierungszeitplans:** Definieren Sie einen Zeitplan zur Evaluierung neuer Services und Features für Ihre Workload.
 - Sie können [AWS Systems Manager Inventory](#) verwenden, um Betriebssystem- (OS), Anwendungs- und Instance-Metadaten von Ihren EC2 Amazon-Instances zu sammeln und so schnell zu verstehen, auf welchen Instances die Software und die Konfigurationen ausgeführt werden, die gemäß Ihrer Softwarerichtlinie erforderlich sind, und welche Instances aktualisiert werden müssen.
- **Bewertung der neuen Aktualisierung:** Erfahren Sie, wie die Komponenten Ihrer Workload aktualisiert werden. Nutzen Sie die Agilität in der Cloud, um schnell zu testen, wie neue Features Ihre Workload verbessern und so die Leistungseffizienz steigern können.
- **Verwendung von Automatisierung:** Verwenden Sie Automatisierung für den Aktualisierungsvorgang, um den Aufwand für die Bereitstellung neuer Features zu reduzieren und Fehler zu begrenzen, die durch manuelle Prozesse verursacht werden.
 - Sie können [CI/CD](#) verwenden, um Container-Images und andere Artefakte im Zusammenhang mit Ihrer Cloud-Anwendung automatisch zu aktualisieren.
 - Sie können Tools wie den [AWS Systems Manager Patch Manager](#) verwenden, um den Systemaktualisierungsprozess zu automatisieren und die Aktivitäten mit [AWS Systems Manager Maintenance Windows](#) zu planen.
- **Dokumentation des Prozesses:** Dokumentieren Sie Ihren Prozess zur Evakuierung von Aktualisierungen und neuen Services. Geben Sie Ihren Eigentümern ausreichend Zeit und Raum zum Forschen, Testen, Experimentieren und zur Validierung von Aktualisierungen und neuen

Services. Schauen Sie sich die dokumentierten Geschäftsanforderungen KPIs an und finden Sie heraus, welche Aktualisierung sich positiv auf Ihr Unternehmen auswirken wird.

Ressourcen

Zugehörige Dokumente:

- [AWS -Blog](#)
- [Was ist neu bei AWS](#)
- [Implementieren von up-to-date Images mit automatisierten EC2 Image Builder Builder-Pipelines](#)

Zugehörige Videos:

- [AWS re:INFORCE 2022 — Automatisierung von Patch-Management und Compliance mithilfe AWS](#)
- [All Things Patch: | Veranstaltungen AWS Systems ManagerAWS](#)

Zugehörige Beispiele:

- [Bestands- und Patch-Verwaltung](#)
- [Workshop zur Beobachtbarkeit](#)

PERF05-BP07 Regelmäßiges Überprüfen von Metriken

Überprüfen Sie im Rahmen der routinemäßigen Wartungsmaßnahme oder als Reaktion auf Ereignisse oder Vorfälle, welche Metriken erfasst werden. Ermitteln Sie anhand dieser Überprüfung, welche Metriken für die Behebung von Problemen wesentlich waren und welche zusätzlichen Metriken, sofern nachverfolgt, helfen könnten, Probleme zu identifizieren, zu beheben oder zu verhindern.

Typische Anti-Muster:

- Sie lassen zu, dass Metriken für einen längeren Zeitraum im Alarmstatus bleiben.
- Sie erstellen Alarme, die von einem Automatisierungssystem nicht umsetzbar sind.

Vorteile der Nutzung dieser bewährten Methode: Überprüfen Sie kontinuierlich Metriken, die erfasst werden, um zu bestätigen, dass sie Probleme ordnungsgemäß identifizieren, beheben oder

verhindern. Metriken können auch veralten, wenn sie für einen längeren Zeitraum im Alarmstatus bleiben.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Verbessern Sie kontinuierlich die Erfassung und Überwachung von Metriken. Bewerten Sie beim Reagieren auf Vorfälle oder Ereignisse diejenigen Metriken, die hilfreich für die Behebung des Problems waren, und überlegen Sie, welche derzeit noch nicht verfolgten Metriken förderlich sein könnten. Verbessern Sie auf diese Weise die Qualität der erfassten Metriken, damit Sie zukünftige Probleme verhindern oder schneller beheben können.

Bewerten Sie beim Reagieren auf Vorfälle oder Ereignisse diejenigen Metriken, die hilfreich für die Behebung des Problems waren, und überlegen Sie, welche derzeit noch nicht verfolgten Metriken förderlich sein könnten. Verbessern Sie auf diese Weise die Qualität der erfassten Metriken, damit Sie zukünftige Probleme verhindern oder schneller beheben können.

Implementierungsschritte

- **Metriken definieren:** Definieren Sie wichtige Leistungsmetriken zur Überwachung, die auf Ihr Workload-Ziel abgestimmt sind, einschließlich Metriken wie Reaktionszeit und Ressourcenauslastung.
- **Ausgangswert festlegen:** Legen Sie für jede Metrik einen Ausgangswert und einen Zielwert fest. Der Ausgangswert sollte Referenzpunkte zur Identifizierung von Abweichungen oder Anomalien enthalten.
- **Takt festlegen:** Legen Sie einen Takt zur Überprüfung wichtiger Kennzahlen fest (z. B. wöchentlich oder monatlich).
- **Leistungsprobleme identifizieren:** Beurteilen Sie bei jeder Überprüfung Trends und Abweichungen von den Ausgangswerten. Suchen Sie nach Leistungsengpässen oder Anomalien. Führen Sie bei identifizierten Problemen eine eingehende Ursachenanalyse durch, um den Hauptgrund für das Problem zu ermitteln.
- **Korrekturmaßnahmen identifizieren:** Identifizieren Sie Korrekturmaßnahmen mithilfe Ihrer Analysen. Dies kann die Parameteroptimierung, das Beheben von Fehlern und das Skalieren von Ressourcen beinhalten.
- **Ergebnisse dokumentieren:** Dokumentieren Sie Ihre Erkenntnisse, einschließlich identifizierter Probleme, Ursachen und Korrekturmaßnahmen.

- Iterieren und verbessern: Beurteilen und verbessern Sie kontinuierlich den Prozess zur Überprüfung der Metriken. Nutzen Sie die Erkenntnisse aus der vorherigen Überprüfung, um den Prozess im Laufe der Zeit zu verbessern.

Ressourcen

Zugehörige Dokumente:

- [CloudWatch-Dokumentation](#)
- [Sammeln von Metriken und Protokollen von Amazon-EC2-Instances und On-Premises-Servern mit dem CloudWatch-Agenten](#)
- [Query your metrics with CloudWatch Metrics Insights](#)
- [Überwachung, Protokollierung und Leistung von AWS Partner Network-Partnern](#)
- [X-Ray-Dokumentation](#)

Zugehörige Videos:

- [AWS re:Invent 2022 - Setting up controls at scale in your AWS environment](#)
- [AWS re:Invent 2022 - How Amazon uses better metrics for improved website performance](#)
- [AWS re:Invent 2023 - Building an effective observability strategy](#)
- [AWS Summit SF 2022 - Full-stack observability and application monitoring with AWS](#)
- [AWS re:Invent 2023 - Take a load off: Diagnose & resolve performance issues with Amazon RDS](#)

Zugehörige Beispiele:

- [Erstellen eines Dashboards mit Quick](#)
- [CloudWatch-Dashboards](#)

Kostenoptimierung

Die Säule „Kostenoptimierung“ umfasst die Fähigkeit, Systeme so auszuführen, dass sie geschäftlichen Wert bei geringstmöglichen Kosten liefern. Verbindliche Anleitungen zur Implementierung finden Sie im [Whitepaper zur Säule „Kostenoptimierung“](#).

Bereiche für bewährte Methoden

- [Praxis für Cloud-Finanzmanagement](#)
- [Ausgabenerkennung und Nutzungsbewusstsein](#)
- [Kostengünstige Ressourcen](#)
- [Verwaltung von Nachfrage und Bereitstellung von Ressourcen](#)
- [Optimierung im Laufe der Zeit](#)

Praxis für Cloud-Finanzmanagement

Frage

- [KOSTEN 1. Wie implementieren Sie das Cloud Financial Management?](#)

KOSTEN 1. Wie implementieren Sie das Cloud Financial Management?

Die Implementierung eines Cloud Financial Managements hilft Organisationen, geschäftliche Mehrwerte und einen wirtschaftlichen Erfolg zu erzielen, während sie die Kosten und die Nutzung optimieren und auf AWS skalieren.

Best Practices

- [COST01-BP01 Übernehmen Sie die Verantwortung für die Kostenoptimierung](#)
- [COST01-BP02 Einrichten einer Partnerschaft zwischen Finanzen und Technologie](#)
- [COST01-BP03 Erstellen von Cloud-Budgets und -Prognosen](#)
- [COST01-BP04 Implementieren von Kostenbewusstsein in Ihre Organisationsprozesse](#)
- [COST01-BP05 Berichte und Benachrichtigungen zur Kostenoptimierung](#)
- [COST01-BP06 Proaktive Überwachung der Kosten](#)
- [COST01-BP07 Bleiben Sie auf up-to-date dem Laufenden mit neuen Service Releases](#)
- [COST01-BP08 Schaffen einer kostenbewussten Kultur](#)
- [COST01-BP09 Quantifizieren Sie den Geschäftswert der Kostenoptimierung](#)

COST01-BP01 Übernehmen Sie die Verantwortung für die Kostenoptimierung

Bilden Sie ein Team (Cloud Business Office, Cloud Center of Excellence oder FinOps Team), das für die Schaffung und Aufrechterhaltung des Kostenbewusstseins in Ihrer gesamten Organisation verantwortlich ist. Für die Kostenoptimierung kann eine Einzelperson oder ein Team zuständig sein

(mit Mitarbeitern aus dem Finanz-, Technologie- und Geschäftsbereich), Voraussetzung ist eine Übersicht über die gesamte Organisation und die Finanzierung der Cloud.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Dies ist die Einführung einer Funktion oder eines Teams im Bereich Cloud Business Office (CBO) oder Cloud Center of Excellence (CCOE), das für den Aufbau und die Aufrechterhaltung einer Kultur des Kostenbewusstseins im Cloud-Computing verantwortlich ist. Bei dieser Funktion kann es sich um eine bereits in der Organisation hierfür zuständige Person, ein Team innerhalb Ihrer Organisation oder um ein neues Team handeln, das sich aus den wichtigsten Finanz-, Technologie- und Organisationsbeteiligten aus der gesamten Organisation zusammensetzt.

Die Funktion (Einzelperson oder Team) priorisiert und verbraucht den erforderlichen Prozentsatz ihrer Zeit für Kostenmanagement- und Kostenoptimierungsaktivitäten. Bei kleinen Unternehmen kann die Funktion einen geringeren Prozentsatz der Zeit im Vergleich zu einer Vollzeitfunktion für ein größeres Unternehmen aufwenden.

Die Funktion erfordert einen multidisziplinären Ansatz, der Kompetenzen in den Bereichen Projektmanagement, Datenwissenschaft, Finanzanalyse und Software- oder Infrastrukturentwicklung voraussetzt. Die Mitarbeiter können die Effizienz von Workloads durch Kostenoptimierungen auf drei unterschiedlichen Verantwortlichkeitsebenen verbessern:

- Zentralisiert: Durch spezielle Teams wie FinOps Team, Cloud Financial Management (CFM) - Team, Cloud Business Office (CBO) oder Cloud Center of Excellence (CCoE) können Kunden Governance-Mechanismen entwerfen und implementieren und unternehmensweit bewährte Verfahren vorantreiben.
- Dezentralisiert: Hierbei werden Technologieteams mit der Durchführung von Kostenoptimierungen beauftragt.
- Hybrid: Zentralisierte und dezentralisierte Teams arbeiten gemeinsam an der Umsetzung von Kostenoptimierungen.

Die Funktion kann anhand ihrer Fähigkeit zur Durchführung und Implementierung im Hinblick auf Kostenoptimierungsziele gemessen werden (z. B. durch Workload-Effizienzmetriken).

Sie müssen sicherstellen, dass Führungskräfte diese Funktion als Sponsoren/Förderer unterstützen. Dies ist ein entscheidender Erfolgsfaktor. Der entsprechende Sponsor befürwortet

eine kosteneffiziente Cloud-Nutzung und bietet Eskalationsunterstützung für das Team, um sicherzustellen, dass die Aktivitäten zur Kostenoptimierung mit der von der Organisation definierten Priorität behandelt werden. Andernfalls können Anweisungen nicht beachtet und Möglichkeiten für Kosteneinsparungen nicht priorisiert werden. Gemeinsam helfen der Sponsor und das Team Ihrer Organisation dabei, die Cloud effizient zu nutzen und Unternehmenswert zu schaffen.

Wenn Sie den Business Enterprise-On-Ramp - oder [Enterprise-Supportplan](#) haben und Hilfe beim Aufbau dieses Teams oder dieser Funktion benötigen, wenden Sie sich über Ihr Account-Team an Ihre Cloud Financial Management (CFM) -Experten.

Implementierungsschritte

- Definieren wichtiger Mitglieder: Alle relevanten Bereiche Ihrer Organisation müssen ihren Beitrag leisten und ein Interesse an der Kostenverwaltung haben. Zu den üblichen Teams innerhalb von Organisationen gehören in der Regel: Finanz-, Anwendungs- oder Produktverantwortliche, Management- und Technikteams (DevOps). Einige Teams setzen ihre ganze Arbeitszeit hierfür ein (Finanz- und Technikbereich), während andere nach Bedarf eingebunden werden. Einzelne Personen oder Teams CFM benötigen die folgenden Fähigkeiten:
 - Softwareentwicklung: um Skripts und Automatisierungen entwickeln zu können.
 - Infrastrukturentwicklung: um Skripts bereitzustellen, Prozesse zu automatisieren und zu verstehen, wie Services oder Ressourcen bereitgestellt werden.
 - Geschäftssinn: Hier CFM geht es darum, effizient in der Cloud zu arbeiten, indem die effiziente Nutzung der Cloud gemessen, überwacht, geändert, geplant und skaliert wird.
- Definieren von Zielen und Metriken: Die Funktion muss der Organisation auf verschiedene Weise Mehrwert bieten. Diese Ziele werden definiert und mit der Entwicklung der Organisation kontinuierlich weiterentwickelt. Häufige Aktivitäten sind das Erstellen und Durchführen von Schulungsprogrammen zur Kostenoptimierung in der gesamten Organisation, Entwickeln von organisationsweiten Standards wie Überwachung und Berichterstellung zur Kostenoptimierung sowie Festlegen der Workload-Ziele bei der Optimierung. Außerdem muss diese Funktion der Organisation regelmäßig über ihre Möglichkeiten zur Kostenoptimierung Bericht erstatten.

Sie können wert- oder kostenbasierte Leistungsindikatoren definieren (KPIs). Wenn Sie die definieren KPIs, können Sie die erwarteten Kosten in Bezug auf Effizienz und erwartete Geschäftsergebnisse berechnen. Wertbasiert KPIs verknüpfen Kosten- und Nutzungskennzahlen mit den Wertfaktoren des Unternehmens und helfen so, Ausgabenänderungen zu rationalisieren. AWS Der erste Schritt zur wertbasierten Ableitung KPIs besteht in der organisationsübergreifenden Zusammenarbeit, um ein Standardpaket auszuwählen und zu vereinbaren. KPIs

- Regelmäßigen Rhythmus etablieren: Die Gruppe (Teams aus den Bereichen Finanzen, Technologie und Geschäft) sollte sich regelmäßig treffen, um Ziele und Metriken zu überprüfen. Dazu gehört in der Regel die Überprüfung des Status der Organisation, der aktuell ausgeführten Programme und der gesamten Finanz- und Optimierungsmetriken. Anschließend werden detaillierte Berichte zu wichtigen Workloads erstellt.

Bei diesen regelmäßigen Überprüfungen können Sie die Workload-Effizienz (Kosten) und die geschäftlichen Ergebnisse bewerten. Eine Kostensteigerung von 20 % für eine Workload könnte beispielsweise mit einer erhöhten Nutzung durch Kunden zusammenhängen. In einem solchen Fall kann die Kostensteigerung von 20 % als Investition betrachtet werden. Diese regelmäßigen Telefongespräche können Teams dabei helfen, Werte zu identifizieren, die für KPIs das gesamte Unternehmen von Bedeutung sind.

Ressourcen

Zugehörige Dokumente:

- [AWS CCOE-Blog](#)
- [Einrichtung von Cloud Business Office](#)
- [CCOE- Cloud-Exzellenzzentrum](#)

Zugehörige Videos:

- [CCOEErfolgsgeschichte von Vanguard](#)

Zugehörige Beispiele:

- [Nutzung eines Cloud Center of Excellence \(CCOE\) zur Transformation des gesamten Unternehmens](#)
- [Aufbau eines CCOE, um das gesamte Unternehmen zu transformieren](#)
- [7 Fallstricke, die Sie beim Bauen vermeiden sollten CCOE](#)

COST01-BP02 Einrichten einer Partnerschaft zwischen Finanzen und Technologie

Beziehen Sie Finanz- und Technologieteams in Kosten- und Nutzungsgespräche in allen Phasen Ihres Wegs in die Cloud mit ein. Teams treffen sich regelmäßig, um Themen wie Unternehmensziele, aktuellen Kosten- und Nutzungsstatus sowie Finanz- und Buchhaltungsmethoden zu besprechen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Technologieteams können in der Cloud dank verkürzter Genehmigungs-, Beschaffungs- und Infrastrukturbereitstellungszyklen schneller Innovationen vorantreiben. Dies kann eine Anpassung für Finanzorganisationen sein, die zuvor an die Ausführung zeitaufwändiger und ressourcenintensiver Prozesse zur Beschaffung und Bereitstellung von Kapital in Rechenzentrums- und On-Premises-Umgebungen und die Kostenzuordnung nur nach Projektgenehmigung gewöhnt waren.

Was die Finanz- und Beschaffungsabteilungen betrifft, wurden die Prozesse in den Bereichen Budgetierung, Kapitalbedarf, Genehmigung, Beschaffung und Installation der physischen Infrastruktur über Jahrzehnte hinweg weiterentwickelt und standardisiert.

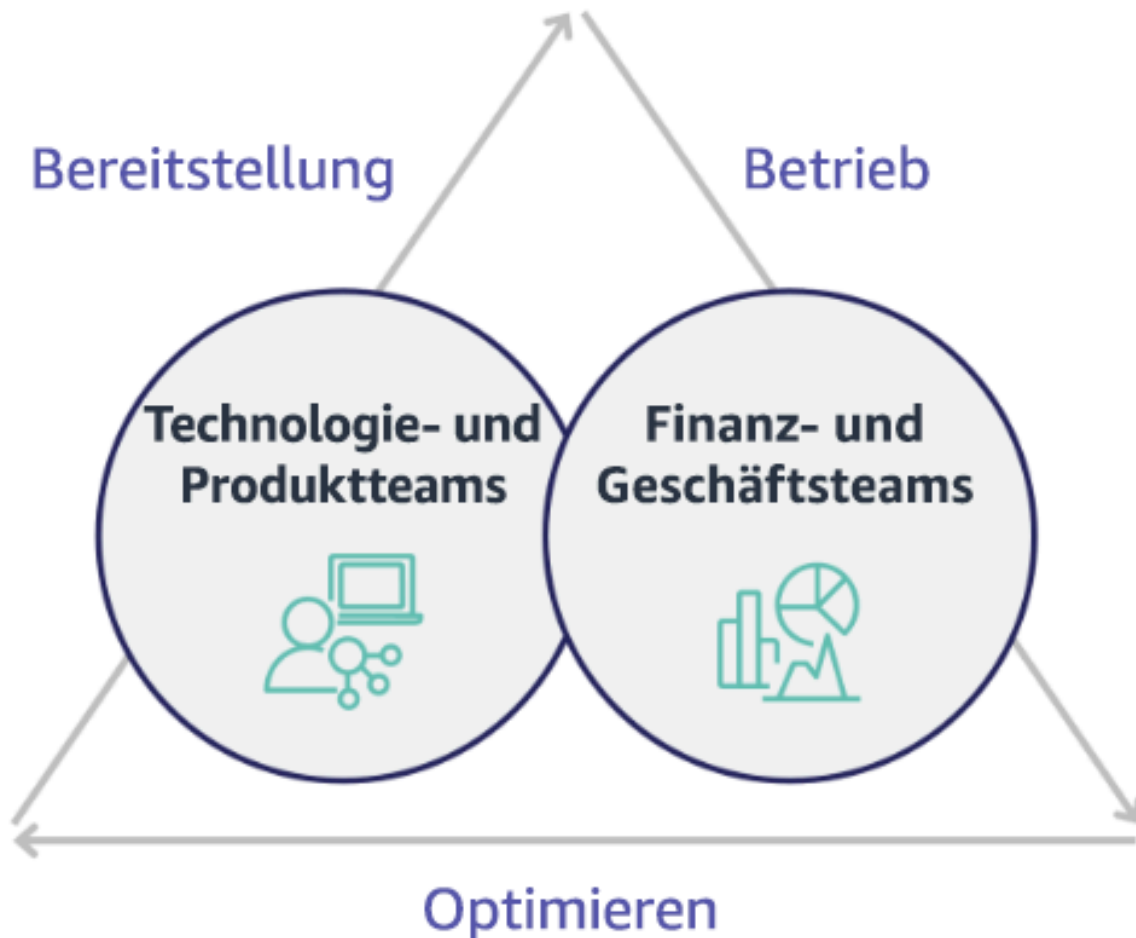
- In der Regel fordern die Entwicklungs- oder IT-Teams die Geldmittel an.
- Die Finanzteams genehmigen und beschaffen die Geldmittel.
- Die operativen Teams stellen die Infrastruktur zusammen, sodass sie direkt eingesetzt werden kann.



Mit der Einführung der Cloud werden Beschaffung und Nutzung der Infrastruktur nicht mehr als Kette von Abhängigkeiten betrachtet. Im Cloud-Modell entwickeln Technologie- und Produktteams ihre Produkte nicht nur, sondern führen sie auch selbst aus und sind für sie verantwortlich. Dabei führen sie die meisten Aktivitäten aus, die bisher als Domäne der Finanz- und operativen Teams betrachtet wurden, einschließlich Beschaffung und Bereitstellung.

Zur Bereitstellung von Cloud-Ressourcen werden lediglich ein Konto und der richtige Satz von Berechtigungen benötigt. Dies reduziert auch die Risiken in den Bereichen IT und Finanzen, da die Teams stets nur einige Klicks oder API-Aufrufe von der Einstellung nicht genutzter oder nicht notwendiger Cloud-Ressourcen entfernt sind. Technologieteams können so auch schneller Innovationen einführen und erhalten die nötige Agilität und Flexibilität, um Experimente zu starten

und zu beenden. Auch wenn sich die variable Natur der Cloud-Nutzung auf die Planbarkeit der Budgetierung und die Genauigkeit von Prognosen auswirken kann, bietet sie Organisationen jedoch auch die Möglichkeit, sowohl die Kosten für Überbereitstellungen als auch die Opportunitätskosten für konservative Unterbereitstellungen zu reduzieren.



Bauen Sie eine Partnerschaft zwischen wichtigen Beteiligten aus dem Finanzwesen und der Technologie auf, um ein gemeinsames Verständnis der organisatorischen Ziele zu schaffen und Mechanismen zu entwickeln, um im variablen Ausgabenmodell von Cloud Computing einen finanziellen Erfolg zu erzielen. Relevante Teams innerhalb Ihres Unternehmens müssen an Kosten- und Nutzungsdiskussionen in allen Phasen Ihrer Cloud-Reise beteiligt sein, einschließlich:

- Verantwortliche im Finanzbereich: CFOs, Finanzkontrolleure, Finanzplaner, Geschäftsanalysten, Beschaffung und Kreditorenbuchhaltung müssen das Cloud-Modell des Verbrauchs, Kaufoptionen und den monatlichen Rechnungsprozess verstehen. Die Teams in den Bereichen Finanzen und Technologie müssen zusammenarbeiten, um die IT-Wertschöpfung zu entwickeln und

darzustellen, damit die geschäftlichen Teams die Verbindung zwischen Technologieausgaben und Geschäftsergebnissen verstehen können. Auf diese Weise werden Technologieaufwendungen nicht als Kosten angesehen, sondern als Investitionen. Aufgrund der grundlegenden Unterschiede zwischen der Cloud (z. B. Änderungsrate der Nutzung, Pay-as-you-go-Preisgestaltung, gestaffelte Preise, Preismodelle und detaillierte Abrechnungs- und Nutzungsinformationen) im Vergleich zum Betrieb vor Ort ist es für die Finanzorganisation von entscheidender Bedeutung, dass sie versteht, wie sich die Nutzung der Cloud auf geschäftliche Aspekte wie Beschaffungsprozesse, Anreizverfolgung, Kostenzuordnung und Finanzberichte auswirken kann.

- Verantwortliche im Technologiebereich: Technologieverantwortliche (einschließlich Produkt- und Anwendungsbesitzer) müssen die finanziellen Anforderungen (z. B. Budgeteinschränkungen) sowie die geschäftlichen Anforderungen (z. B. Service Level Agreements) kennen. Damit kann die Workload implementiert werden, um die gewünschten Ziele des Unternehmens zu erreichen.

Die Partnerschaft zwischen Finanzen und Technologie bietet folgende Vorteile:

- Finanz- und Technologieteams haben nahezu in Echtzeit Einblicke in Kosten und Nutzung.
- Finanz- und Technologieteams legen ein standardmäßiges Betriebsverfahren für die Bewältigung von Ausgabeunterschieden in der Cloud fest.
- Stakeholder im Bereich Finanzen handeln als strategische Berater bei der Nutzung von Kapital für den Kauf rabattierter Programme (z. B. Reserved Instances oder AWS Savings Plans) und der Nutzung der Cloud zur Förderung des Wachstums der Organisation.
- Vorhandene Kreditorenbuchhaltungs- und Beschaffungsprozesse werden mit der Cloud verwendet.
- Die Finanz- und Technologieteams prognostizieren gemeinsam die Kosten und die Nutzung von AWS in der Zukunft, um die Budgets der Organisation entsprechend auszurichten und zu entwickeln.
- Bessere unternehmensübergreifende Kommunikation durch eine gemeinsame Sprache und ein gemeinsames Verständnis von Finanzkonzepten.

Weitere Beteiligte innerhalb Ihres Unternehmens, die an Kosten- und Nutzungsdiskussionen beteiligt sein sollten, sind:

- Besitzer von Geschäftseinheiten: Besitzer von Geschäftseinheiten müssen sich mit dem Cloud-Geschäftsmodell vertraut machen, sodass sie den Geschäftseinheiten und dem gesamten Unternehmen die Richtung vorweisen können. Dieses Cloud-Wissen ist wichtig, wenn es

erforderlich ist, das Wachstum und die Workloadnutzung zu prognostizieren oder verschiedene Kaufoptionen zu bewerten, z. B. Reserved Instances oder Savings Plans.

- **Technikteam:** Eine Partnerschaft zwischen Finanz- und Technologieteams hat kritische Bedeutung für die Entwicklung einer kostenbewussten Kultur, die Entwickler motiviert, im Bereich Cloud Financial Management (CFM) aktiv zu werden. Ein häufiges Problem von CFM- und Finanzteams besteht darin, Entwicklern ein Verständnis des Geschäfts in der Cloud zu vermitteln und sie zur Umsetzung von Best Practices und empfohlenen Aktionen zu motivieren.
- **Dritte:** Wenn Ihr Unternehmen Dritte verwendet (z. B. Berater oder Tools), stellen Sie sicher, dass sie an Ihren finanziellen Zielen ausgerichtet sind, und können sowohl die Ausrichtung durch ihre Engagement-Modelle als auch einen ROI (Return on Investment) nachweisen. In der Regel beteiligen sich Dritte an der Berichterstellung und Analyse der von ihnen verwalteten Systeme, und sie stellen Kostenanalysen für die von ihnen konzipierten Workloads bereit.

Eine erfolgreiche CFM-Implementierung erfordert die Zusammenarbeit von Teams in den Bereichen Finanzen, Technologie und Geschäft sowie eine veränderte Kommunikation und Evaluierung in Bezug auf die Cloud-Ausgaben der Organisation. Beziehen Sie die Entwicklungsteams in alle Phasen der Diskussion über Kosten- und Nutzung ein und motivieren Sie sie zur Befolgung von Best Practices und zur Umsetzung vereinbarter Aktionen.

Implementierungsschritte

- **Definieren wichtiger Mitglieder:** Stellen Sie sicher, dass sich alle relevanten Mitglieder Ihrer Finanz- und Technologieteams aktiv an der Partnerschaft beteiligen. Relevante Mitglieder im Bereich Finanzen sind Personen, die mit Cloud-Ausgaben interagieren. Dies sind in der Regel CFOs, Finanzcontroller, Finanzplaner, Geschäftsanalysten und Mitarbeiter in Beschaffung und Einkauf. Technologiemitglieder sind in der Regel Produkt- und Anwendungsbesitzer, technische Manager und Vertreter aller Teams, die in der Cloud aktiv sind. Weitere Mitglieder können Geschäftsbereiche mit Einfluss auf die Nutzung von Produkten sein, zum Beispiel das Marketing, und Dritte wie Berater, die Sie bei der Ausrichtung an Ihren Zielen und Mechanismen und bei Berichten unterstützen.
- **Definieren von Besprechungsthemen:** Definieren Sie die Themen, die in den Teams häufig auftreten, oder ein gemeinsames Verständnis erfordern. Verfolgen Sie die Kosten ab dem Zeitpunkt, an dem sie generiert werden, bis zur Bezahlung der Rechnung. Beachten Sie alle beteiligten Mitglieder und organisatorischen Prozesse, die angewendet werden müssen. Informieren Sie sich über jeden einzelnen Schritt oder Prozess, den sie durchlaufen, sowie die

zugehörigen Informationen, wie z. B. verfügbare Preismodelle, gestaffelte Preise, Rabattmodelle, Budgetplanung und finanzielle Anforderungen.

- Regelmäßigen Rhythmus etablieren: Um eine Finanz- und Technologiepartnerschaft zu schaffen, sollte ein regelmäßiger Kommunikationsrhythmus festgelegt werden, um eine Abstimmung zu erreichen und aufrechtzuerhalten. Die Gruppe muss regelmäßig im Hinblick auf ihre Ziele und Metriken zusammenkommen. Dazu gehört in der Regel die Überprüfung des Status der Organisation, der aktuell ausgeführten Programme und der gesamten Finanz- und Optimierungsmetriken. Anschließend werden detaillierte Berichte zu wichtigen Workloads erstellt.

Ressourcen

Zugehörige Dokumente:

- [AWS News Blog](#)

COST01-BP03 Erstellen von Cloud-Budgets und -Prognosen

Passen Sie vorhandene Budgetierungs- und Prognoseprozesse so an, dass sie mit der stark variablen Natur der Cloud-Kosten und -Nutzung kompatibel sind. Prozesse müssen dynamisch sein und Algorithmen anwenden, die auf Trends oder Geschäftsfaktoren oder einer Kombination aus beiden basieren.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Bei herkömmlichen On-Premises-IT-Setups stehen Kunden oft vor der Herausforderung, Fixkosten zu planen, die sich nur gelegentlich ändern, typischerweise beim Kauf neuer IT-Geräte und -Services, um die Spitzennachfrage zu decken. Im Gegensatz dazu verfolgt AWS Cloud einen anderen Ansatz, bei dem Kunden nur für die Ressourcen bezahlen, die sie nutzen, und zwar entsprechend ihren tatsächlichen IT- und Geschäftsanforderungen. In der Cloud-Umgebung kann die Nachfrage monatlich, täglich oder sogar stündlich schwanken.

Die Nutzung der Cloud bringt Effizienz, Geschwindigkeit und Agilität, damit allerdings auch ein stark variables Kosten- und Nutzungsmuster. Die Kosten können als Reaktion auf eine höhere Workload-Effizienz oder die Bereitstellung neuer Workloads und Features sinken oder manchmal eben auch steigen. Wenn Workloads skaliert werden, um einen wachsenden Kundenstamm zu bedienen, steigen parallel dazu die Cloud-Nutzung und -Kosten aufgrund der besseren Verfügbarkeit von

Ressourcen. Diese Flexibilität bei Cloud-Services erstreckt sich auch auf die Kosten und Prognosen, was zu einer gewissen Elastizität führt.

Es ist wichtig, sich eng an diesen sich ändernden Geschäftsanforderungen und Nachfragetreibern auszurichten und eine möglichst genaue Planung anzustreben. Traditionelle Budgetprozesse in Organisationen müssen angepasst werden, um dieser Variabilität Rechnung zu tragen.

Ziehen Sie bei der Prognose der Kosten für neue Workloads eine Kostenmodellierung in Betracht. Durch die Kostenmodellierung erhalten Sie eine grundlegende Vorstellung von den erwarteten Cloud-Kosten. Dies hilft Ihnen, die Gesamtbetriebskosten (TCO) und die Kapitalrendite (ROI) zu ermitteln und andere Finanzanalysen durchzuführen, Ziele und Erwartungen mit Stakeholdern festzulegen und Möglichkeiten zur Kostenoptimierung zu identifizieren.

Ihre Organisation muss die Kostendefinitionen und akzeptierten Gruppierungen kennen. Der Detaillierungsgrad, mit dem Sie Prognosen erstellen, kann je nach Struktur und internen Workflows Ihrer Organisation variieren. Wählen Sie eine Granularität, die Ihren spezifischen Anforderungen und Ihrer Organisationsstruktur entspricht. Es ist wichtig zu verstehen, auf welcher Ebene die Prognose durchgeführt wird:

- **Verwaltungskonto oder AWS Organizations-Ebene:** Das Verwaltungskonto ist das Konto, das Sie zum Erstellen von AWS Organizations verwenden. Organisationen haben standardmäßig ein Verwaltungskonto.
- **Verknüpftes Konto oder Mitgliedskonto:** Ein Konto in Organisationen ist ein Standard-AWS-Konto, das Ihre AWS-Ressourcen und die Identitäten enthält, die auf diese Ressourcen zugreifen können.
- **Umgebung:** Eine Umgebung ist eine Sammlung von AWS-Ressourcen, die eine Anwendungsversion ausführen. Eine Umgebung kann mit mehreren verknüpften oder Mitgliedskonten erstellt werden.
- **Projekt:** Ein Projekt ist eine Kombination aus festgelegten Zielen oder Aufgaben, die innerhalb eines bestimmten Zeitraums zu erfüllen sind. Es ist wichtig, den Projektlebenszyklus bei Ihrer Prognose zu berücksichtigen.
- **AWS-Services:** Gruppen oder Kategorien wie Datenverarbeitungs- oder Speicherservices, in denen Sie AWS-Services für Ihre Prognose gruppieren können.
- **Benutzerdefinierte Gruppierung:** Sie können benutzerdefinierte Gruppen erstellen, die auf den Anforderungen Ihrer Organisation basieren, z. B. Geschäftseinheiten, Kostenstellen, Teams, Kostenzuordnungs-Tags, Kostenkategorien, verknüpfte Konten oder eine Kombination davon.

Identifizieren Sie die Geschäftsfaktoren, die sich auf Ihre Nutzungskosten auswirken können, und erstellen Sie für jeden dieser Faktoren separate Prognosen, um die erwartete Nutzung im Voraus zu berechnen. Einige der Faktoren fallen in den Verantwortungsbereich von IT- und Produktteams innerhalb der Organisation. Andere Geschäftsfaktoren, wie Marketingveranstaltungen, Werbeaktionen, geografische Expansionen, Fusionen und Übernahmen, sind den Führungskräften in Vertrieb und Marketing und der Geschäftsleitung bekannt. Es ist wichtig, zusammenzuarbeiten und auch all diese Nachfragetreiber zu berücksichtigen.

Mit [AWS Cost Explorer](#) können Sie Kosten für einen definierten zukünftigen Zeitraum basierend auf Trends und Ihren bisherigen Ausgaben prognostizieren. Die Prognose-Engine von AWS Cost Explorer segmentiert Ihre historischen Daten auf Grundlage von Gebührentypen (z. B. Reserved Instances) und verwendet eine Kombination aus Machine Learning und regelbasierten Modellen, um die Ausgaben für alle Gebührentypen individuell zu prognostizieren.

Sobald Sie Ihren Prognoseprozess eingerichtet und Modelle erstellt haben, können Sie mit [AWS Budgets](#) angepasste, detaillierte Budgets festlegen, indem Sie den Zeitraum, die Wiederholungen oder den Betrag (fest oder variabel) angeben und Filter wie Service, AWS-Region und Tags hinzufügen. Das Budget wird in der Regel für ein Jahr geplant und bleibt unverändert, sodass alle Stakeholder sich strikt daran halten müssen. Im Gegensatz dazu sind Prognosen flexibler, da sie erneute Anpassungen im Laufe des Jahres ermöglichen und dynamische Prognosen über einen Zeitraum von einem, zwei oder drei Jahren liefern. Sowohl die Budgetierung als auch Prognosen spielen eine entscheidende Rolle bei der Definition der Finanzerwartungen verschiedener Stakeholder aus dem technischen und geschäftlichen Bereich. Genaue Prognosen und deren Umsetzung sorgen zudem dafür, dass die Stakeholder, die direkt für die Bereitstellungskosten verantwortlich sind, zur Rechenschaft gezogen werden. Außerdem wird auf diese Weise das allgemeine Kostenbewusstsein gestärkt.

Um über die Leistung Ihrer bestehenden Budgets auf dem Laufenden zu bleiben, können Sie AWS Budgets-Berichte erstellen und planen, die Sie und Ihre Stakeholder in regelmäßigen Abständen per E-Mail erhalten. Sie können auch AWS Budgets-Warnmeldungen basierend auf tatsächlichen Kosten erstellen, also einen reaktiven Prozess. Budgetwarnungen zu prognostizierten Kosten geben Ihnen Zeit, Abhilfemaßnahmen gegen potenzielle Kostenüberschreitungen zu implementieren. Sie können sich benachrichtigen lassen, wenn Ihre Kosten oder Ihre Nutzung ein bestimmtes Niveau übersteigen oder in der Zukunft den budgetierten Betrag möglicherweise überschreiten werden.

Gestalten Sie vorhandene Budget- und Prognoseprozesse dynamischer. Hierzu können Sie trendbasierte Algorithmen (mit historischen Kosten als Eingabe) und auf Geschäftsfaktoren basierende Algorithmen verwenden (z. B. auf der Einführung neuer Produkte, auf einer regionalen

Expansion oder neuen Umgebungen für Workloads), die besonders für Umgebungen mit dynamischen und variablen Ausgaben geeignet sind. Sobald Sie Ihre trendbasierte Prognose mithilfe von Cost Explorer oder anderen Tools ermittelt haben, können Sie mit dem [AWS Pricing Calculator](#) Ihren AWS-Anwendungsfall und die zukünftigen Kosten auf Grundlage der erwarteten Nutzung abschätzen (Datenverkehr, Anfragen pro Sekunde oder erforderliche Amazon-EC2-Instances).

Überprüfen Sie die Genauigkeit dieser Prognose, da Budgets auf Grundlage dieser Prognoseberechnungen und -schätzungen festgelegt werden sollten. Überwachen Sie die Genauigkeit und Effektivität der integrierten Cloud-Kostenprognosen. Überprüfen Sie regelmäßig die tatsächlichen Ausgaben im Vergleich zur Prognose und passen Sie sie bei Bedarf an, um die Prognosepräzision zu verbessern. Verfolgen Sie die Prognoseabweichung und führen Sie eine Ursachenanalyse der berichteten Abweichungen durch, um zu reagieren und die Prognosen anzupassen.

Wie in [COST01-BP02 Einrichten einer Partnerschaft zwischen Finanzen und Technologie](#) erwähnt, ist es wichtig, eine Partnerschaft mit regelmäßigen Konsultationen zwischen IT, Finanzabteilung und anderen Stakeholdern zu schaffen, um zu bestätigen, dass alle in konsistenter Weise die gleichen Tools oder Prozesse anwenden. Wenn Budgets geändert werden müssen, führen Sie häufigere Besprechungen durch, um schneller darauf zu reagieren.

Implementierungsschritte

- Definieren Sie die Kostensprache innerhalb der Organisation: Schaffen Sie eine gemeinsame AWS-Kostensprache innerhalb der Organisation mit mehreren Dimensionen und Gruppierungen. Stellen Sie sicher, dass die Stakeholder die Granularität der Prognosen, die Preismodelle und das Niveau Ihrer Kostenprognosen verstehen.
- Analysieren Sie trendbasierte Prognosen: Verwenden Sie trendbasierte Prognosetools wie AWS Cost Explorer und Amazon Forecast. Analysieren Sie Ihre Nutzungskosten anhand verschiedener Dimensionen wie Service, Konto, Tags und Kostenkategorien.
- Analysieren Sie faktorbasierte Prognosen: Identifizieren Sie die Auswirkungen geschäftlicher Faktoren auf Ihre Cloud-Nutzung und erstellen Sie für jeden Faktor eine separate Prognose, um die erwarteten Nutzungskosten im Voraus zu berechnen. Arbeiten Sie eng mit Verantwortlichen von Geschäftseinheiten und Stakeholdern zusammen, um die Auswirkungen auf neue Faktoren zu verstehen und die erwarteten Kostenänderungen zu berechnen. So können Sie genaue Budgets definieren.
- Aktualisieren Sie die bestehenden Prognose- und Budgetprozesse: Definieren Sie Ihre Prozesse für die Prognose und Budgetierung auf Grundlage von bewährten Prognosemethoden, z. B.

trendbasiert, geschäftsfaktorenbasiert oder einer Kombination aus beiden Ansätzen. Budgets sollten kalkuliert werden, realistisch sein und auf Ihren Prognosen basieren.

- Konfigurieren Sie Warnmeldungen und Benachrichtigungen: Verwenden Sie AWS Budgets-Warnmeldungen und die Erkennung von Kostenanomalien, um Warnmeldungen und Benachrichtigungen zu erhalten.
- Führen Sie regelmäßige Prüfungen zusammen mit wichtigen Stakeholdern durch: Einigen Sie sich mit Stakeholdern in den Bereichen IT, Finanzen, Plattform usw. auf Änderungen der Unternehmensausrichtung und der Nutzung.

Ressourcen

Zugehörige Dokumente:

- [AWS Cost Explorer](#)
- [AWS Cost and Usage Report](#)
- [Prognosen mit Cost Explorer](#)
- [Quick-Prognose](#)
- [AWS Budgets](#)

Zugehörige Videos:

- [Wie kann ich AWS Budgets verwenden, um meine Ausgaben und Nutzung zu verfolgen?](#)
- [AWS Serie zur Kostenoptimierung: AWS Budgets](#)

Zugehörige Beispiele:

- [Faktorbasierte Prognosen verstehen und erstellen](#)
- [Eine Prognosekultur schaffen und fördern](#)
- [Prognosen für Cloud-Kosten optimieren](#)
- [Die richtigen Tools für Cloud-Kostenprognosen verwenden](#)

COST01-BP04 Implementieren von Kostenbewusstsein in Ihre Organisationsprozesse

Implementieren Sie Kostenbewusstsein und sorgen Sie für Transparenz und Verantwortlichkeit bei neuen oder bestehenden Prozessen, die sich auf die Nutzung auswirken, und greifen Sie

auf vorhandene Prozesse zur Steigerung des Kostenbewusstseins zurück. Implementieren Sie Kostenbewusstsein in die Mitarbeiterschulung.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Das Kostenbewusstsein muss in neuen und vorhandenen Organisationsprozessen implementiert werden. Dies ist eine der absoluten Grundlagen für weitere bewährte Methoden. Es wird empfohlen, vorhandene Prozesse nach Möglichkeit wiederzuverwenden und zu ändern. Dadurch werden die Auswirkungen auf Agilität und Geschwindigkeit minimiert. Informieren Sie die Technologieteams und die Entscheidungsträger in den Geschäfts- und Finanzteams über die Cloud-Kosten, um das Kostenbewusstsein zu verbessern, und richten Sie KPIs zur Effizienz für Beteiligte aus dem Finanz- und Geschäftsbereich ein. Die folgenden Empfehlungen helfen Ihnen bei der Implementierung der Kostenerkennung in Ihrer Workload:

- Stellen Sie sicher, dass das Änderungsmanagement eine Kostenmessung umfasst, um die finanziellen Auswirkungen Ihrer Änderungen zu quantifizieren. Auf diese Weise können Sie kostenbezogene Probleme proaktiv lösen und Kosteneinsparungen hervorheben.
- Stellen Sie sicher, dass die Kostenoptimierung eine zentrale Komponente Ihrer Betriebsfunktionen ist. Sie können beispielsweise vorhandene Vorfallmanagementprozesse nutzen, um die Ursache für Kosten- und Nutzungsanomalien (Kostenüberschreitungen) zu ermitteln und zu identifizieren.
- Beschleunigen Sie die Kosteneinsparungen und die Wertschöpfung des Unternehmens durch Automatisierung oder Tools. Wenn Sie über die Kosten der Implementierung nachdenken, sollten Sie das Gespräch so gestalten, dass es eine ROI-Komponente enthält, um die Investition von Zeit oder Geld zu rechtfertigen.
- Weisen Sie Cloud-Kosten zu, indem Sie Showbacks oder Chargebacks für Cloud-Aufwendungen implementieren, einschließlich Aufwendungen für verpflichtungsbasierte Kaufoptionen, gemeinsam genutzte Services und Markteinkäufe, um die Cloudnutzung in möglichst kostenbewusster Weise zu gestalten.
- Erweitern Sie vorhandene Schulungs- und Entwicklungsprogramme, um Schulungen zum Kostenbewusstsein in Ihrer gesamten Organisation einzubeziehen. Es wird empfohlen, dass dies fortlaufende Schulungen und Zertifizierungen umfasst. Dadurch entsteht eine Organisation, die Kosten und Nutzung selbst verwalten kann.
- Nutzen Sie kostenlose, native AWS-Tools, wie etwa [AWS Cost Anomaly Detection](#), [AWS Budgets](#) und [AWS Budgets-Berichte](#).

Wenn Organisationen [Cloud Financial Management](#) (CFM)-Methoden in konsistenter Weise einsetzen, werden die entsprechenden Verhaltensweisen bald echte Bestandteile der Arbeitsweise und der Entscheidungsfindung. Das führt zu einer kostenbewussteren Kultur, in der Entwickler eine neue, in der Cloud entwickelte Anwendung bauen und Finanzmanager den ROI dieser neuen Cloud-Investitionen analysieren.

Implementierungsschritte

- Bestimmen relevanter organisatorischer Prozesse: Jede Organisationseinheit überprüft ihre Prozesse und identifiziert Prozesse, die sich auf Kosten und Nutzung auswirken. Alle Prozesse, die zur Erstellung oder Beendigung einer Ressource führen, müssen zur Überprüfung einbezogen werden. Suchen Sie auch nach Prozessen, die das Kostenbewusstsein in Ihrem Unternehmen unterstützen können, wie z. B. Vorfallmanagement und Schulungen.
- Schaffen einer sich selbst erhaltenden Kostenbewusstseinskultur: Sorgen Sie dafür, dass alle relevanten Beteiligten die Ursache für die Änderung und die Auswirkungen als Kosten betrachten, damit sie die Cloud-Kosten verstehen. So kann Ihre Organisation eine sich selbst erhaltende, kostenbewusste Innovationskultur entwickeln.
- Aktualisieren von Prozessen mit Kostenbewusstsein: Jeder Prozess wird so geändert, dass er kostenbewusst wird. Der Prozess erfordert möglicherweise zusätzliche Vorabprüfungen, z. B. das Bewerten der Auswirkungen von Kosten oder nachträgliche Prüfungen, die bestätigen, dass die erwarteten Kosten- und Nutzungsänderungen stattgefunden haben. Unterstützungsprozesse wie Schulungs- und Vorfallmanagement können auf Kosten- und Nutzungselemente erweitert werden.

Wenden Sie sich für Unterstützung über Ihr Kontoteam an CFM-Sachverständige oder erkunden Sie die nachfolgend aufgeführten Ressourcen und Dokumente.

Ressourcen

Zugehörige Dokumente:

- [AWS-Cloud-Finanzmanagement](#)

Zugehörige Beispiele:

- [Strategie für effizientes Cloud-Kostenmanagement](#)
- [Blog-Serie zum Thema Kostenkontrolle Nr. 3: Umgang mit Kostenschocks](#)
- [AWS Cost Management für Anfänger](#)

COST01-BP05 Berichte und Benachrichtigungen zur Kostenoptimierung

Richten Sie Cloud-Budgets ein und konfigurieren Sie Mechanismen zur Erkennung von Anomalien bei der Nutzung. Konfigurieren Sie zugehörige Tools für Kosten- und Nutzungswarnungen für vordefinierte Ziele und lassen Sie sich benachrichtigen, wenn eine Nutzung diese Ziele überschreitet. Halten Sie regelmäßig Meetings ab, um die Kosteneffektivität Ihrer Workloads zu analysieren und das Kostenbewusstsein zu stärken.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Niedrig

Implementierungsleitfaden

Sie müssen regelmäßig Kosten- und Nutzungsoptimierungen in Ihrer Organisation melden. Sie können dedizierte Sitzungen implementieren, um die Kosteneffizienz zu besprechen, oder die Kostenoptimierung in Ihre regulären operativen Berichtszyklen für Ihre Workloads einschließen. Nutzen Sie Services und Tools, um die Kosteneffizienz regelmäßig zu überwachen und Möglichkeiten zur Kosteneinsparung zu nutzen.

Zeigen Sie Ihre Kosten und Nutzung mit mehreren Filtern und Granularität an, indem Sie [AWS Cost Explorer](#) verwenden, das Dashboards und Berichte wie Kosten pro Service oder Konto, Tageskosten oder Marktplatzkosten bereitstellt. Sie können Ihren Fortschritt bei Kosten und Nutzung mit [AWS Budgets-Berichten](#) anhand konfigurierter Budgets verfolgen.

Mit [AWS Budgets](#) können Sie angepasste Budgets einrichten, um Kosten und Nutzung nachzuverfolgen und schnell auf Warnungen zu reagieren, die Sie per E-Mail oder als Benachrichtigungen von Amazon Simple Notification Service (Amazon SNS) erhalten, wenn Sie Ihren Schwellenwert überschreiten. [Legen Sie den bevorzugten Budgetzeitraum](#) auf täglich, monatlich, vierteljährlich oder jährlich fest und richten Sie spezifische Budgetlimits ein, um zu sehen, wie sich die tatsächlichen oder prognostizierten Kosten in Bezug auf Ihren Budgetschwellenwert entwickeln. Sie können auch eine automatische Ausführung von [Warnungen](#) und [Aktionen](#) oder einen Genehmigungsprozess für den Fall einrichten, dass ein Budgetziel überschritten wird.

Darüber hinaus können Sie mit Benachrichtigungen zu Kosten und Nutzung schnell auf unerwartete Änderungen bei Kosten und Nutzung reagieren. [AWS Cost Anomaly Detection](#) ermöglicht Ihnen die Reduzierung von Überraschungen bei den Kosten und die Verbesserung der Kontrolle, ohne die Innovationsfähigkeit zu beeinträchtigen. AWS Cost Anomaly Detection identifiziert anomale Ausgaben und ihre Ursachen, was Ihnen hilft, das Risiko für Überraschungen bei Abrechnungen zu reduzieren. In drei einfachen Schritten können Sie Ihre eigene kontextorientierte Überwachung einrichten und Benachrichtigungen erhalten, wenn anomale Ausgaben entdeckt werden.

Sie können [QuickSight](#) mit AWS Cost and Usage Report (CUR)-Daten verwenden, um hoch angepasste Berichte mit detaillierteren Daten zu erstellen. Quick Suite ermöglicht Ihnen die Planung von Berichten und den Erhalt regelmäßiger E-Mails mit Berichten zu historischen Kosten und Nutzung oder zu Möglichkeiten für Kosteneinsparungen. Sehen Sie sich unsere [Cost Intelligence Dashboard](#) (CID)-Lösung an, die auf QuickSight basiert und erweiterte Transparenz bietet.

Mit [AWS Trusted Advisor](#) erhalten Sie Anleitungen, um zu überprüfen, ob bereitgestellte Ressourcen die bewährten Methoden von AWS zur Kostenoptimierung befolgen.

Vergleichen Sie Ihre Savings Plans-Empfehlungen anhand detaillierter grafischer Darstellungen zu Ihren Kosten und der Nutzung. Nach Stunden unterteilte Grafiken zeigen die On-Demand-Ausgaben zusammen mit den empfohlenen Savings Plans-Verpflichtungen und geben Aufschluss über die geschätzten Einsparungen, die Savings Plans-Abdeckung und Savings Plans-Nutzung. Auf diese Weise können Organisationen nachvollziehen, wie ihre Savings Plans auf jede aufgewendete Stunde angewendet werden, ohne Zeit und Ressourcen in die Erstellung von Modellen zur Analyse ihrer Ausgaben investieren zu müssen.

Sie können regelmäßige Berichte erstellen, die Savings Plans, Reserved Instances und Amazon-EC2-Empfehlungen aus AWS Cost Explorer für Anpassungen enthalten, um die Kosten für Steady-State-Workloads sowie nicht genutzte und nicht vollständig genutzte Ressourcen zu reduzieren. Identifizieren Sie unnötige Cloud-Ausgaben, die mit bereitgestellten Ressourcen verbunden sind, und gewinnen Sie diese zurück. Unnötige Cloud-Ausgaben entstehen, wenn Ressourcen mit der falschen Größe erstellt werden oder wenn andere als die erwarteten Nutzungsmuster beobachtet werden. Befolgen Sie die bewährten Methoden von AWS, um Ihre Verschwendung zu reduzieren, oder bitten Sie Ihr Kontoteam und Ihren Partner, Ihnen bei der [Optimierung und Einsparung](#) Ihrer Cloud-Kosten zu helfen.

Generieren Sie regelmäßig Berichte zu besseren Kaufoptionen für Ihre Ressourcen, um die Kosten pro Einheit für Ihre Workloads zu senken. Kaufoptionen wie Savings Plans, Reserved Instances oder Amazon EC2 Spot Instances bieten die umfassendsten Kosteneinsparungen für fehlertolerante Workloads. Stakeholder (Geschäftsleitung, Finanz- und Technologieteams) können sich an den Diskussionen zu den damit verbundenen Verpflichtungen beteiligen.

Teilen Sie die Berichte, die Einsparmöglichkeiten beschreiben, oder Ankündigungen neuer Versionen, um die Gesamtbetriebskosten (TCO) der Cloud zu reduzieren. Führen Sie neue Services, Regionen, Funktionen, Lösungen oder neue Möglichkeiten für weitere Kostenreduzierungen ein.

Implementierungsschritte

- Konfigurieren Sie AWS Budgets: Konfigurieren Sie AWS Budgets für alle Konten Ihrer Workload. Legen Sie ein Budget für die Gesamtkontoausgaben und ein Budget für die Workload mithilfe von Tags fest.
 - [Well-Architected Labs: Kosten und Steuerung der Nutzung](#)
- Bericht zur Kostenoptimierung: Richten Sie einen regelmäßigen Zyklus ein, um die Effizienz der Workload zu erörtern und zu analysieren. Melden Sie anhand der eingerichteten Metriken die erreichten Metriken und die Kosten für deren Erreichung. Identifizieren und beheben Sie negative Trends und suchen Sie nach positiven Trends, die Sie in der gesamten Organisation fördern können. Bei der Berichterstellung sollten Vertreter der Anwendungsteams und -Verantwortlichen, Finanzverantwortliche und wichtige Entscheidungsträger in Bezug auf Cloud-Ausgaben einbezogen werden.

Ressourcen

Zugehörige Dokumente:

- [AWS Cost Explorer](#)
- [AWS Trusted Advisor](#)
- [AWS Budgets](#)
- [AWS Cost and Usage Report](#)
- [AWS Budgets Bewährte Methoden](#)
- [Amazon-S3-Analytik](#)

Zugehörige Beispiele:

- [Zentrale Methoden für die Optimierung Ihrer AWS-Cloud-Kosten](#)

COST01-BP06 Proaktive Überwachung der Kosten

Implementieren Sie Tools und Dashboards, um die Kosten für die Workload proaktiv zu überwachen. Überprüfen Sie regelmäßig die Kosten mithilfe konfigurierter oder vorab erstellter Tools. Untersuchen Sie Kosten und Kategorien nicht erst, wenn Sie Benachrichtigungen erhalten. Die proaktive Überwachung und Analyse der Kosten hilft Ihnen, positive Trends zu identifizieren und diese in der gesamten Organisation zu unterstützen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Es wird empfohlen, die Kosten und die Nutzung innerhalb Ihrer Organisation proaktiv zu überwachen, nicht nur, wenn Ausnahmen oder Anomalien vorliegen. Hoch sichtbare Dashboards in Ihrem Büro oder Ihrer Arbeitsumgebung stellen sicher, dass relevante Mitarbeiter Zugriff auf benötigte Informationen haben, und signalisieren den Fokus der Organisation auf Kostenoptimierungen. Mit gut sichtbaren Dashboards können Sie den Erfolg aktiv unterstützen und positive Ergebnisse in der gesamten Organisation implementieren.

Entwickeln Sie eine täglich oder häufig ausgeführte Routine für die Verwendung von [AWS Cost Explorer](#) oder anderen Dashboards wie [Amazon Quick Suite](#), um die Kosten darzustellen und proaktiv zu analysieren. Analysieren Sie mithilfe von Gruppierung und Filterung Kosten und Nutzung von AWS-Services auf der Ebene von AWS-Konten, Workloads oder spezifischen AWS-Services und überprüfen Sie, ob es sich um erwartete oder unerwartete Ergebnisse handelt. Nutzen Sie die Granularität und die Tags auf Stunden- und Ressourcenbasis, um für die wichtigsten Ressourcen wiederkehrende Kosten herauszufiltern und zu identifizieren. Sie können auch über das [Cost Intelligence Dashboard](#) eigene Berichte erstellen. Dabei handelt es sich um eine [Amazon-QuickSight](#)-Lösung, die von AWS Solution Architects entwickelt wurde. Sie ermöglicht Ihnen den Vergleich Ihrer Budgets mit den tatsächlichen Kosten und der tatsächlichen Nutzung.

Implementierungsschritte

- Bericht zur Kostenoptimierung: Richten Sie einen regelmäßigen Zyklus ein, um die Effizienz der Workload zu erörtern und zu analysieren. Melden Sie anhand der eingerichteten Metriken die erreichten Metriken und die Kosten für deren Erreichung. Identifizieren und beheben Sie negative Trends und suchen Sie nach positiven Trends, um diese in der gesamten Organisation zu unterstützen. Bei der Berichterstellung sollten Vertreter der Anwendungsteams und Besitzer, Finanz- und Geschäftsleitung einbezogen werden.
- Erstellen und aktivieren Sie tägliche, detaillierte [AWS Budgets](#) für Kosten und Nutzung, um rechtzeitig Maßnahmen gegen potenzielle Kostenüberschreitungen ergreifen zu können. Mit AWS Budgets können Sie Warnungen konfigurieren, um stets zu wissen, ob ein Budgettyp außerhalb der vorab konfigurierten Schwellenwerte liegt. Die beste Art, AWS Budgets zu nutzen, besteht in der Einrichtung der erwarteten Kosten und der erwarteten Nutzung als Grenzwerte. So können alle Budgetüberschreitungen identifiziert werden.
- Erstellen Sie AWS Cost Anomaly Detection zur Kostenüberwachung: [AWS Cost Anomaly Detection](#) verwendet eine erweiterte Machine-Learning-Technologie, um anomale Ausgaben

und ihre Ursachen schnell zu identifizieren, damit Sie schnell Maßnahmen ergreifen können. Sie können auf diese Weise Tools für die Überwachung der Kosten von Ausgabensegmenten konfigurieren, die Sie überwachen möchten (z. B. einzelne AWS-Services, Mitgliederkonten, Kostenzuweisungs-Tags und Kostenkategorien). Sie können auch festlegen, wann, wo und wie Sie Warnungen erhalten. Jedem Überwachungstool können Sie mehrere Warnungsabonnements für Geschäftsbereichsleiter und Technologieteams anfügen, einschließlich Name, Kostenschwellenwert und Häufigkeit (einzelne Warnungen, tägliche Zusammenfassung, wöchentliche Zusammenfassung) für die einzelnen Abonnements.

- Verwenden Sie AWS Cost Explorer oder integrieren Sie Ihre AWS Cost and Usage Report (CUR)-Daten in Amazon-QuickSight-Dashboards, um die Kosten Ihrer Organisation zu visualisieren: AWS Cost Explorer verfügt über eine benutzerfreundliche Oberfläche, mit der Sie Ihre AWS-Kosten und Nutzung im Laufe der Zeit visualisieren, verstehen und verwalten können. Das [Cost Intelligence Dashboard](#) ist ein anpassbares und zugängliches Dashboard, mit dem Sie die Grundlagen für Ihr eigenes Tool für Kostenmanagement und Optimierung legen können.

Ressourcen

Zugehörige Dokumente:

- [AWS Budgets](#)
- [AWS Cost Explorer](#)
- [Tägliche Kosten und Nutzungsbudgets](#)
- [AWS Cost Anomaly Detection](#)

Zugehörige Beispiele:

- [AWS Cost Anomaly Detection-Warnung mit Slack](#)

COST01-BP07 Bleiben Sie auf up-to-date dem Laufenden mit neuen Service Releases

Konsultieren Sie regelmäßig Experten oder AWS Partner, um herauszufinden, welche Dienste und Funktionen kostengünstiger sind. Lesen Sie AWS Blogs und andere Informationsquellen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

AWS fügt ständig neue Funktionen hinzu, sodass Sie die neuesten Technologien nutzen können, um schneller zu experimentieren und Innovationen zu entwickeln. Möglicherweise können Sie neue AWS Dienste und Funktionen implementieren, um die Kosteneffizienz Ihrer Workloads zu erhöhen. Lesen Sie regelmäßig [AWS Cost Management](#), den [AWS News Blog](#), den [AWS Cost Management-Blog](#) und [Neuerungen bei AWS](#), um Informationen zur Veröffentlichung neuer Services und Features zu erhalten. Neue Beiträge bieten einen kurzen Überblick über alle Ankündigungen zu AWS Diensten, Funktionen und Regionserweiterungen, sobald sie veröffentlicht werden.

Implementierungsschritte

- Blogs abonnieren: Gehen Sie zu den AWS Blogseiten und abonnieren Sie den What's New Blog und andere relevante Blogs. Sie können sich auf der Seite für die [Kommunikationseinstellungen](#) mit Ihrer E-Mail-Adresse registrieren.
- AWS Neuigkeiten abonnieren: Informieren Sie sich regelmäßig im [AWS News-Blog](#) und unter [Neuigkeiten über AWS neue](#) Service- und Feature-Releases. Abonnieren Sie den RSS Feed oder mit Ihrer E-Mail-Adresse, um Ankündigungen und Veröffentlichungen zu verfolgen.
- AWS Preissenkungen folgen: Regelmäßige Preissenkungen bei all unseren Dienstleistungen waren eine Standardmethode, AWS um die wirtschaftliche Effizienz, die sich aus unserer Skalierung ergeben, an unsere Kunden weiterzugeben. Stand 20. September 2023, AWS hat die Preise seit 2006 134 Mal gesenkt. Wenn geschäftliche Entscheidungen aufgrund von Preisbedenken ausstehen, können Sie die Preise nach der Reduzierung und der Integration neuer Services erneut prüfen. In der [Kategorie Preissenkungen im AWS News-Blog](#) können Sie sich über die bisherigen Preissenkungsmaßnahmen, einschließlich Amazon Elastic Compute Cloud (AmazonEC2) -Instances, informieren.
- AWS Veranstaltungen und Treffen: Nehmen Sie an Ihrem lokalen AWS Gipfel und allen lokalen Treffen mit anderen Organisationen aus Ihrer Region teil. Wenn Sie nicht persönlich teilnehmen können, versuchen Sie, virtuelle Veranstaltungen zu besuchen, um mehr von AWS Experten und Geschäftsszenarien anderer Kunden zu erfahren.
- Treffen Sie sich mit Ihrem Account-Team: Planen Sie regelmäßige Treffen mit Ihrem Account-Team, um über Branchentrends und AWS -Services zu sprechen. Sprechen Sie mit Ihrem Account Manager, Solutions Architekt und Support-Team.

Ressourcen

Zugehörige Dokumente:

- [AWS Kostenmanagement](#)
- [Was ist neu bei AWS](#)
- [AWS Nachrichten-Blog](#)

Zugehörige Beispiele:

- [Amazon EC2 — 15 Jahre Optimierung und Einsparung Ihrer IT-Kosten](#)
- [AWS News-Blog — Preissenkung](#)

COST01-BP08 Schaffen einer kostenbewussten Kultur

Implementieren Sie Änderungen oder Programme in Ihrer gesamten Organisation, um eine kostenbewusste Kultur zu schaffen. Es wird empfohlen, klein zu beginnen. Wenn Ihre Kompetenz und die Nutzung der Cloud in Ihrer Organisation zunehmen, implementieren Sie große und umfangreiche Programme.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Niedrig

Implementierungsleitfaden

Eine kostenbewusste Kultur ermöglicht Ihnen die Skalierung von Kostenoptimierung und Cloud-Finanzmanagement (betriebliche Abläufe, Cloud-Kompetenzzentrum, Cloud-Betriebsteams usw.) mithilfe von bewährten Methoden, die in der gesamten Organisation auf organische und dezentralisierte Weise angewendet werden. Wenn Sie ein Kostenbewusstsein entwickeln, können Sie im Vergleich zu einem zentralisierten Top-Down-Approach in der gesamten Organisation mit minimalem Aufwand einen hohen Grad an Kompetenz erzielen.

Die Entwicklung eines Kostenbewusstseins im Bereich Cloud Computing, insbesondere bei primären Kostenfaktoren, ermöglicht Teams, die voraussichtlichen Ergebnisse von Änderungen aus Kostensicht zu verstehen. Teams, die auf Cloud-Umgebungen zugreifen, sollten die Preismodelle kennen und den Unterschied zwischen herkömmlichen On-Premises-Rechenzentren und Cloud Computing verstehen.

Der Hauptvorteil einer Kultur des Kostenbewusstseins besteht darin, dass Technologieteams die Kosten proaktiv und kontinuierlich optimieren, statt bedarfsbasiert reaktive Kostenoptimierungen durchzuführen. (Die Kosten werden beispielsweise als eine nicht funktionale Anforderung betrachtet, wenn neue Workloads entwickelt oder vorhandene Workloads geändert werden.)

Kleine Veränderungen in der Kultur können große Auswirkungen auf die Effizienz Ihrer aktuellen und zukünftigen Workloads haben. Beispiele hierfür sind:

- Transparenz und Schaffung eines Bewusstseins bei Entwicklungsteams, damit diese verstehen, was sie tun und wie sich dies auf die Kosten auswirkt.
- Gamifizierung von Kosten und Nutzung in Ihrer gesamten Organisation. Dies kann über ein öffentliches Dashboard oder einen Bericht erfolgen, der Kosten und Nutzung normalisiert und teamübergreifend vergleicht (z. B. Kosten pro Workload und Kosten pro Transaktion).
- Kosteneffizienz erkennen. Belohnen Sie freiwillige oder unaufgeforderte Kostenoptimierungsleistungen öffentlich oder privat und lernen Sie aus Fehlern, um eine Wiederholung in Zukunft zu vermeiden.
- Erstellen Sie Top-Down-Organisationsanforderungen für die Ausführung von Workloads mit vordefinierten Budgets.
- Hinterfragen Sie die geschäftlichen Anforderungen in Bezug auf Änderungen und die Kostenauswirkungen von Änderungsanforderungen für die Architekturinfrastruktur oder die Workload-Konfiguration, um sicherzustellen, dass Sie nur für das bezahlen, was Sie benötigen.
- Stellen Sie sicher, dass sich Änderungsplaner voraussichtlicher Änderungen mit Auswirkungen auf die Kosten bewusst sind und dass diese Änderungen von den Stakeholdern genehmigt werden, um geschäftliche Ergebnisse auf kosteneffektive Weise zu erzielen.

Implementierungsschritte

- Informieren Sie die Technologieteams über die Cloud-Kosten: So erhöhen Sie das Kostenbewusstsein und können Effizienz-KPIs für Stakeholder in den Bereichen Finanzen und Geschäft einrichten.
- Informieren Sie Stakeholder oder Teammitglieder über geplante Änderungen: Erstellen Sie einen Tagesordnungspunkt zur Erörterung geplanter Änderungen und der Kosten-Nutzen-Auswirkungen auf die Arbeitsbelastung während der wöchentlichen Änderungsbesprechungen.
- Treffen Sie sich mit Ihrem Account-Team: Richten Sie regelmäßige Treffen mit Ihrem Account-Team ein, um über Branchentrends und AWS-Services zu sprechen. Sprechen Sie mit Ihrem Account Manager, Solutions Architect und Support-Team.
- Teilen Sie Erfolgsgeschichten: Teilen Sie Erfolgsgeschichten zu Kostensenkungen für eine Workload, ein AWS-Konto oder eine Abteilung, um eine positive Einstellung zu generieren und zu Kostensenkungen zu motivieren.

- Schulungen: Stellen Sie sicher, dass Technologieteams oder Teammitglieder in Bezug auf die Ressourcenkosten in AWS Cloud geschult sind.
- AWS-Veranstaltungen und -Treffen: Nehmen Sie an lokalen AWS-Summits und weiteren lokalen Treffen mit anderen Organisationen aus Ihrer Region teil.
- Abonnieren Sie Blogs: Rufen Sie die AWS-Blogs-Seiten auf und abonnieren Sie den [Blog „Neuerungen“](#) und weitere relevante Blogs, um bei neuen Veröffentlichungen, Implementierungen, Beispielen und Änderungen auf dem Laufenden zu bleiben, die von AWS geteilt werden.

Ressourcen

Zugehörige Dokumente:

- [AWS-Blog](#)
- [AWS-Kostenmanagement](#)
- [AWS News Blog](#)

Zugehörige Beispiele:

- [AWS-Cloud-Finanzmanagement](#)

COST01-BP09 Quantifizieren Sie den Geschäftswert der Kostenoptimierung

Durch die Quantifizierung des Geschäftswerts von Kostenoptimierungen können Sie die gesamten Vorteile für Ihre Organisation verstehen. Da die Kostenoptimierung eine notwendige Investition ist, können Sie durch die Quantifizierung des Geschäftswerts den Beteiligten den ROI erklären. Die Quantifizierung des Geschäftswerts kann Ihnen helfen, mehr Unterstützung von Beteiligten für zukünftige Investitionen zur Kostenoptimierung zu gewinnen, und bietet einen Rahmen, um die Ergebnisse für die Kostenoptimierung Ihrer Organisation zu messen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Die Quantifizierung des Geschäftswerts bedeutet, dass der Nutzen gemessen wird, den Unternehmen aus ihren Maßnahmen und Entscheidungen ziehen. Bei dem Geschäftswert kann es sich um einen materiellen Wert (z. B. geringere Ausgaben oder höhere Gewinne) oder einen immateriellen Wert (z. B. ein besseres Markenimage oder eine höhere Kundenzufriedenheit) handeln.

Die Quantifizierung des geschäftlichen Nutzens der Kostenoptimierung bedeutet, dass Sie feststellen müssen, wie viel Wert oder Nutzen Sie aus Ihren Bemühungen um effizientere Ausgaben ziehen. Wenn ein Unternehmen beispielsweise 100.000\$ ausgibt, um einen Workload bereitzustellen AWS und ihn später zu optimieren, belaufen sich die neuen Kosten auf nur 80.000\$, ohne dass die Qualität oder der Output beeinträchtigt wird. In diesem Szenario würde der quantifizierte Geschäftswert aus der Kostenoptimierung eine Einsparung von 20.000 USD bedeuten. Aber über die reinen Einsparungen hinaus kann das Unternehmen den Wert auch in Form von kürzeren Lieferzeiten, verbesserter Kundenzufriedenheit oder anderen Metriken, die sich aus den Kostenoptimierungsbemühungen ergeben, quantifizieren. Die Beteiligten müssen Entscheidungen über den potenziellen Wert der Kostenoptimierung, die Kosten für die Optimierung der Workload und den Ertragswert treffen.

Zusätzlich zu den Einsparungen durch Kostenoptimierung wird empfohlen, den zusätzlichen Wert zu quantifizieren. Die Vorteile der Kostenoptimierung werden in der Regel in Bezug auf niedrigere Kosten pro Geschäftsergebnis quantifiziert. Sie können beispielsweise die Kosteneinsparungen Amazon Elastic Compute Cloud(AmazonEC2) quantifizieren, wenn Sie Savings Plans erwerben, die die Kosten senken und das Workload-Output-Niveau aufrechterhalten. Sie können die Kostensenkungen bei den AWS Ausgaben quantifizieren, wenn inaktive EC2 Amazon-Instances entfernt oder nicht verknüpfte Amazon Elastic Block Store (AmazonEBS) -Volumes gelöscht werden.

Die Vorteile der Kostenoptimierung gehen jedoch über die Kostensenkung oder -vermeidung hinaus. Ziehen Sie in Betracht, zusätzliche Daten zu erfassen, um Effizienzsteigerungen und Geschäftswert zu messen.

Implementierungsschritte

- **Bewertung der geschäftlichen Vorteile:** Bei diesem Prozess werden die AWS Cloud Kosten analysiert und so angepasst, dass der Nutzen aus jedem ausgegebenen Dollar maximiert wird. Anstatt sich auf Kostensenkungen ohne geschäftlichen Nutzen zu konzentrieren, sollten Sie die Geschäftsvorteile und die Kapitalrendite für die Kostenoptimierung in Betracht ziehen, da diese einen größeren Nutzen aus den von Ihnen ausgegebenen Mitteln ziehen können. Dabei geht es darum, Ausgaben umsichtig vorzunehmen und Investitionen und Ausgaben in Bereichen zu tätigen, die den besten Ertrag bringen.
- **Analysieren Sie AWS Kostenprognosen:** Mithilfe von Prognosen können Stakeholder aus der Finanzabteilung bei anderen internen und externen Stakeholdern der Organisation ihre Erwartungen festlegen. Außerdem können sie die finanzielle Planbarkeit Ihres Unternehmens verbessern. [AWS Cost Explorer](#) kann verwendet werden, um Prognosen für Ihre Kosten und Nutzung durchzuführen.

Ressourcen

Zugehörige Dokumente:

- [AWS Cloud Wirtschaftswissenschaften](#)
- [AWS Blog](#)
- [AWS Kostenmanagement](#)
- [AWS Nachrichten-Blog](#)
- [Well-Architected-Whitepaper zur Säule „Zuverlässigkeit“](#)
- [AWS Cost Explorer](#)

Zugehörige Videos:

- [Erschließen Sie geschäftlichen Nutzen mit eingeschaltetem Windows AWS](#)

Zugehörige Beispiele:

- [Measuring and Maximizing the Business Value of Customer 360](#)
- [The Business Value of Adopting Amazon Web Services Managed Databases](#)
- [The Business Value of Amazon Web Services for Independent Software Vendors](#)
- [Business Value of Cloud Modernization](#)
- [The Business Value of Migration to Amazon Web Services](#)

Ausgabenerkennung und Nutzungsbewusstsein

Fragen

- [KOSTEN 2. Wie können Sie die Nutzung steuern?](#)
- [KOSTEN 3. Wie überwachen Sie Ihre Kosten und die Nutzung?](#)
- [KOSTEN 4. Wie können Sie Ressourcen außer Betrieb nehmen?](#)

KOSTEN 2. Wie können Sie die Nutzung steuern?

Legen Sie Richtlinien und Mechanismen fest, um zu überprüfen, ob angemessene Kosten anfallen und die Ziele erreicht werden. Durch den Einsatz eines Kontrollsystems können Sie Innovationen vorantreiben, ohne das Budget zu überschreiten.

Best Practices

- [COST02-BP01 Entwickeln von Richtlinien auf Basis Ihrer Organisationsanforderungen](#)
- [COST02-BP02 Implementieren von Zielen und Ergebnissen](#)
- [COST02-BP03 Implementieren einer Kontenstruktur](#)
- [COST02-BP04 Implementieren von Gruppen und Rollen](#)
- [COST02-BP05 Implementieren von Kostenkontrollen](#)
- [COST02-BP06 Projektlebenszyklus verfolgen](#)

COST02-BP01 Entwickeln von Richtlinien auf Basis Ihrer Organisationsanforderungen

Entwickeln Sie Richtlinien, die definieren, wie Ressourcen von Ihrem Unternehmen verwaltet werden, und überprüfen Sie sie regelmäßig. Die Richtlinien sollten sich auch mit den Kostenaspekten der Ressourcen und Workloads befassen, einschließlich Erstellung, Änderung und Außerbetriebnahme während der gesamten Lebensdauer der Ressourcen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Die Kenntnis der Kostentreiber in Ihrem Unternehmen ist für die effektive Verwaltung Ihrer Ausgaben und Nutzung und die Identifizierung von Kostenreduzierungsmöglichkeiten von entscheidender Bedeutung. Unternehmen betreiben in der Regel mehrere Workloads, die von mehreren Teams ausgeführt werden. Diese Teams können sich in verschiedenen Organisationseinheiten befinden, die jeweils über eigene Einnahmequellen verfügen. Die Möglichkeit, die Ressourcenkosten den Workloads, der jeweiligen Organisation oder den Produkteigentümern zuzuordnen, fördert ein effizientes Nutzungsverhalten und hilft, die Verschwendung von Ressourcen einzudämmen. Eine genaue Kosten- und Nutzungsüberwachung hilft Ihnen zu verstehen, wie optimiert ein Workload ist und wie profitabel Geschäftsbereiche und Produkte sind. Mit diesem Wissen können Sie fundiertere Entscheidungen dazu treffen, wo Ressourcen in Ihrem Unternehmen eingesetzt werden sollen. Das Bewusstsein der Nutzung auf allen Unternehmensebenen ist entscheidend für Veränderungen, da eine Änderung der Nutzung zu Kostenänderungen führt. Überlegen Sie sich, beim Ermitteln von Nutzungsmustern und Ausgaben einen mehrschichtigen Ansatz zu nutzen.

Der erste Schritt bei der Implementierung von Governance besteht darin, Richtlinien für die Cloud-Nutzung anhand der Anforderungen Ihres Unternehmens zu entwickeln. Diese Richtlinien definieren, wie Ihr Unternehmen die Cloud verwendet und wie Ressourcen verwaltet werden. Richtlinien sollten

alle Aspekte von Ressourcen und Workloads abdecken, die sich auf Kosten oder Nutzung beziehen, einschließlich Erstellung, Änderung und Außerbetriebnahme über die Lebensdauer der Ressource. Überprüfen Sie, ob die Richtlinien und Verfahren bei jeder Änderung in einer Cloud-Umgebung eingehalten und umgesetzt werden. Stellen Sie bei Ihren IT-Änderungsmanagement-Meetings Fragen zu den Kostenauswirkungen geplanter Änderungen, ob die Kosten steigen oder sinken, zur geschäftlichen Rechtfertigung und zum erwarteten Ergebnis.

Richtlinien sollten einfach sein, damit sie leicht verständlich sind und im gesamten Unternehmen effektiv implementiert werden können. Richtlinien müssen außerdem leicht zu befolgen und zu interpretieren sein (damit sie angewendet werden können) sowie spezifisch sein (keine Fehlinterpretationen zwischen den Teams). Darüber hinaus müssen sie (wie unsere Mechanismen) regelmäßig überprüft und aktualisiert werden, wenn sich die Geschäftsbedingungen oder Prioritäten der Kunden ändern, wodurch die Richtlinie veraltet wäre.

Beginnen Sie mit umfangreichen allgemeinen Richtlinien, z. B. welche geografische Region verwendet werden soll oder zu welchen Tageszeiten Ressourcen ausgeführt werden sollen. Verfeinern Sie schrittweise die Richtlinien für die verschiedenen Organisationseinheiten und Workloads. Zu den allgemeinen Richtlinien gehört, welche Services und Funktionen verwendet werden können (z. B. Speicher mit niedrigerer Leistung in Test- und Entwicklungsumgebungen), welche Ressourcentypen von verschiedenen Gruppen verwendet werden können (z. B. ist die größte Ressource in einem Entwicklungskonto mittelgroß) und wie lange diese Ressourcen verwendet werden (ob vorübergehend, kurzfristig oder für einen bestimmten Zeitraum).

Beispiele für Richtlinien

Im Folgenden finden Sie eine Beispielrichtlinie, die Sie überprüfen können, um Ihre eigenen Cloud-Governance-Richtlinien zur Kostenoptimierung zu erstellen. Stellen Sie sicher, dass Sie die Richtlinien an die Anforderungen Ihres Unternehmens und die Anforderungen Ihrer Interessenvertreter anpassen.

- **Richtliniename:** Definieren Sie einen eindeutigen Namen für die Richtlinie, z. B. Richtlinie zur Ressourcenoptimierung und Kostenreduzierung.
- **Zweck:** Erläutern Sie, warum diese Richtlinie angewendet werden sollte und was das erwartete Ergebnis ist. Mit dieser Richtlinie soll überprüft werden, ob für die Bereitstellung und Ausführung des gewünschten Workloads Mindestkosten anfallen, um die Geschäftsanforderungen zu erfüllen.
- **Umfang:** Definieren Sie klar, wer diese Richtlinie verwenden soll und wann sie verwendet werden soll, z. B. DevOps X-Team für Kunden im Osten der USA für Umgebung X (Produktion oder Nicht-Produktion).

Richtlinienanweisung

1. Wählen Sie basierend auf der Umgebung Ihres Workloads und den Geschäftsanforderungen (Entwicklung, Benutzerakzeptanztests, Vorproduktion oder Produktion) entweder us-east-1 oder mehrere us-east-Regionen aus.
2. Planen Sie die Ausführung von Amazon-EC2- und Amazon-RDS-Instances zwischen sechs Uhr morgens und acht Uhr abends (Eastern Standard Time (EST)).
3. Stoppen Sie alle ungenutzten Amazon-EC2-Instances nach acht Stunden und nicht genutzte Amazon-RDS-Instances nach 24 Stunden Inaktivität.
4. Beenden Sie alle ungenutzten Amazon-EC2-Instances nach 24 Stunden Inaktivität in Nicht-Produktionsumgebungen. Erinnern Sie den Besitzer der Amazon-EC2-Instance (anhand von Tags) daran, seine gestoppten Amazon-EC2-Instances in der Produktion zu überprüfen, und teilen Sie ihm mit, dass seine Amazon-EC2-Instances innerhalb von 72 Stunden beendet werden, wenn sie nicht verwendet werden.
5. Verwenden Sie eine generische Instance-Familie und -größe wie m5.large und passen Sie dann die Größe der Instance anhand der CPU- und Speicherauslastung mithilfe von AWS Compute Optimizer an.
6. Priorisieren Sie mithilfe von Auto Scaling, um die Anzahl der ausgeführten Instances je nach Datenverkehr dynamisch anzupassen.
7. Verwenden Sie Spot-Instances für unkritische Workloads.
8. Prüfen Sie die Kapazitätsanforderungen, um Speicherpläne oder Reserved-Instances für vorhersehbare Workloads festzulegen, und informieren Sie das Cloud-Financial-Management-Team.
9. Verwenden Sie Amazon-S3-Lebenszyklusrichtlinien, um Daten, auf die selten zugegriffen wird, auf günstigere Speicherebenen zu verschieben. Wenn keine Aufbewahrungsrichtlinie definiert ist, verwenden Sie Amazon S3 Intelligent-Tiering, um Objekte automatisch auf die Archivebene zu verschieben.
10. Überwachen Sie die Ressourcenauslastung und richten Sie mithilfe von Amazon CloudWatch Alarmer ein, um Skalierungsereignisse auszulösen.
11. Verwenden Sie für jedes AWS-Konto AWS Budgets, um die Kosten- und Nutzungsbudgets für Ihr Konto basierend auf Kostenstelle und Geschäftsbereichen festzulegen.
12. Indem Sie für Ihr Konto mithilfe von AWS Budgets Kosten- und Nutzungsbudgets festlegen, behalten Sie die Ausgaben im Blick und vermeiden unerwartete Rechnungen, was Ihnen eine bessere Kostenkontrolle ermöglicht.

Verfahren: Richten Sie detaillierte Verfahren für die Umsetzung dieser Richtlinie ein oder verweisen Sie auf andere Dokumente, in denen beschrieben wird, wie die einzelnen Grundsatzserklärungen umgesetzt werden. Dieser Abschnitt sollte schrittweise Anweisungen zur Erfüllung der Richtlinienanforderungen enthalten.

Zur Umsetzung dieser Richtlinie können Sie verschiedene Tools von Drittanbietern oder AWS Config-Regeln verwenden, um die Einhaltung der Richtlinienerklärung zu überprüfen und mithilfe von AWS Lambda-Funktionen automatische Abhilfemaßnahmen auszulösen. Sie können auch AWS Organizations verwenden, um die Richtlinie durchzusetzen. Darüber hinaus sollten Sie Ihre Ressourcennutzung regelmäßig überprüfen und die Richtlinie bei Bedarf anpassen, um sicherzustellen, dass sie weiterhin Ihren Geschäftsanforderungen entspricht.

Implementierungsschritte

- **Treffen mit Interessenvertretern:** Um Richtlinien zu entwickeln, bitten Sie die Interessenvertreter (Cloud-Geschäftsstellen, Techniker oder funktionale Entscheidungsträger für die Durchsetzung von Richtlinien) innerhalb Ihrer Organisation, ihre Anforderungen festzulegen und zu dokumentieren. Führen Sie einen iterativen Ansatz aus, indem Sie bei jedem Schritt umfassend beginnen und kontinuierlich auf die kleinsten Einheiten verfeinern. Zu den Teammitgliedern gehören Personen mit direktem Interesse am Workload, z. B. Organisationseinheiten oder Anwendungsbesitzer sowie unterstützende Gruppen wie Sicherheits- und Finanzteams.
- **Bestätigung einholen:** Vergewissern Sie sich, dass sich diejenigen Teams auf Richtlinien einigen, die auf die AWS Cloud Zugriff haben und darin Bereitstellungen vornehmen können. Sorgen Sie dafür, dass sie die Richtlinien Ihres Unternehmens befolgen und stellen Sie sicher, dass ihre Ressourcenerstellung mit den vereinbarten Richtlinien und Verfahren übereinstimmt.
- **Onboarding-Trainings veranstalten:** Fordern Sie neue Organisationmitglieder auf, Onboarding-Trainings zu absolvieren, um ein Kostenbewusstsein und ein Verständnis für die Organisationsanforderungen zu schaffen. Möglicherweise gehen neue Unternehmensmitarbeiter aufgrund ihrer bisherigen Erfahrungen von anderen Richtlinien aus oder denken überhaupt nicht daran.
- **Festlegen der Speicherorte für Ihre Workload:** Definieren Sie, wo Ihre Workload ausgeführt wird, einschließlich des Landes und der Region innerhalb des Landes. Diese Informationen werden für die Zuweisung zu AWS-Regionen und Availability Zones verwendet.
- **Definieren und Gruppieren von Services und Ressourcen:** Definieren Sie die Services, die für die Workloads erforderlich sind. Geben Sie für jeden Service die Typen, den Umfang und die Anzahl der erforderlichen Ressourcen an. Definieren Sie Gruppen für die Ressourcen nach Funktion, z. B. Anwendungsserver oder Datenbankspeicher. Ressourcen können mehreren Gruppen angehören.

- Definieren und Gruppieren der Benutzer nach Funktion: Definieren Sie die Benutzer, die mit der Workload interagieren, und konzentrieren Sie sich darauf, was sie tun und wie sie die Workload verwenden, nicht auf die Benutzer oder ihre Position in der Organisation. Fassen Sie ähnliche Benutzer oder Funktionen in einer Gruppe zusammen. Sie können die von AWS verwalteten Richtlinien als Leitfaden verwenden.
- Definieren der Aktionen: Definieren Sie mithilfe der zuvor identifizierten Standorte, Ressourcen und Benutzer die Aktionen, die von jedem benötigt werden, um die Workload-Ergebnisse über die Lebensdauer (Entwicklung, Betrieb und Außerbetriebnahme) zu erzielen. Identifizieren Sie die Aktionen an jedem Standort basierend auf den Gruppen, nicht auf den einzelnen Elementen in den Gruppen. Beginnen Sie umfassend mit Lese- oder Schreibvorgängen und verfeinern Sie dann auf bestimmte Aktionen für jeden Service.
- Definieren des Überprüfungszeitraums: Workloads und Organisationsanforderungen können sich im Laufe der Zeit ändern. Definieren Sie den Zeitplan für die Überprüfung des Workloads, um sicherzustellen, dass er mit den Prioritäten der Organisation übereinstimmt.
- Dokumentieren der Richtlinien: Stellen Sie sicher, dass auf die definierten Richtlinien zugegriffen werden kann, wie von Ihrer Organisation gefordert. Diese Richtlinien werden verwendet, um den Zugriff auf Ihre Umgebungen zu implementieren, zu verwalten und zu prüfen.

Ressourcen

Zugehörige Dokumente:

- [Änderungsmanagement in der Cloud](#)
- [Von AWS verwaltete Richtlinien für Auftragsfunktionen](#)
- [AWS-Fakturierungsstrategie mit mehreren Konten](#)
- [Aktionen, Ressourcen und Bedingungsschlüssel für AWS-Services](#)
- [Verwaltung und Governance in AWS](#)
- [Steuern des Zugriffs auf AWS-Regionen mit IAM-Richtlinien](#)
- [Globale Infrastrukturregionen und -AZs](#)

Zugehörige Videos:

- [AWS Management and Governance at Scale](#)

COST02-BP02 Implementieren von Zielen und Ergebnissen

Implementieren Sie Kosten- und Nutzungsziele sowie Vorgaben für Ihre Workload. Ziele geben Ihrer Organisation die Richtung für die erwarteten Ergebnisse vor, und Vorgaben geben spezifische, messbare Ergebnisse vor, die für Ihre Workloads erreicht werden sollen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Entwickeln Sie Kosten- und Nutzungsziele sowie Vorgaben für Ihre Organisation. Als wachsende Organisation mit AWS ist es für Sie wichtig, Ziele zur Kostenoptimierung zu setzen und diese zu verfolgen. Diese Ziele oder wichtigen Leistungsindikatoren ([Key Performance Indicators, KPIs](#)) können Dinge wie den Prozentsatz der Bedarfsausgaben oder die Einführung bestimmter optimierter Services wie AWS-Graviton-Instances oder gp3-EBS-Volumetypen umfassen. Legen Sie messbare und erreichbare Ziele fest, anhand derer Sie Effizienzverbesserungen bewerten können, was für den Geschäftsbetrieb wichtig ist. Ziele bieten Ihrer Organisation richtungsweisende Anleitungen hinsichtlich der erwarteten Ergebnisse.

Vorgaben bieten spezifische messbare Ergebnisse, die erreicht werden müssen. Kurz gesagt: Ein Ziel ist die Richtung, in die Sie gehen wollen, und die Vorgabe ist, wie weit Sie in diese Richtung gehen und wann dieses Ziel erreicht werden soll (verwenden Sie die SMART-Orientierungshilfe (Specific, Measurable, Assignable, Realistic, and Timely), d. h. spezifische, messbare, zuweisbare, realistische und zeitgerechte Ziele). Ein Beispiel für ein Ziel ist, dass die Nutzung der Plattform deutlich steigen soll, wobei die Kosten nur geringfügig (nicht linear) steigen sollen. Ein Beispiel für eine Vorgabe ist eine Steigerung der Plattformnutzung um 20 % bei einem Kostenanstieg von weniger als fünf Prozent. Ein weiteres häufiges Ziel ist, dass Workloads alle sechs Monate effizienter werden müssen. Die damit verbundene Vorgabe wäre, dass die Kosten pro Geschäftsmetrik alle 6 Monate um 5 Prozent sinken müssen. Verwenden Sie die richtigen Metriken und legen Sie berechnete KPIs für Ihre Organisation fest. Sie können mit grundlegenden KPIs beginnen und diese später je nach Geschäftsanforderungen weiterentwickeln.

Ein Ziel der Kostenoptimierung besteht darin, die Workload-Effizienz zu erhöhen, also die Kosten pro Geschäftsergebnis der Workload im Laufe der Zeit zu senken. Implementieren Sie dieses Ziel für alle Workloads und legen Sie als Vorgabe beispielsweise eine 5-prozentige Steigerung der Effizienz alle 6 Monate bis zu 1 Jahr fest. In der Cloud können Sie dies durch den Aufbau von Funktionen zur Kostenoptimierung sowie durch neue Service- und Feature-Releases erreichen.

Vorgaben sind die quantifizierbaren Benchmarks, die Sie anstreben, um Ihre Ziele zu erreichen, und mithilfe von Benchmarks werden die tatsächlichen Ergebnisse mit einer Vorgabe verglichen. Erstellen

Sie Benchmarks mit KPIs für die Kosten pro Einheit von Computing-Services (wie Spot-Einführung, Graviton-Einführung, neueste Instance-Typen und On-Demand-Abdeckung), Speicherdiensten (wie EBS GP3-Einführung, überholte EBS-Snapshots und Speicher von Amazon S3 Standard) oder die Nutzung von Datenbank-Services (wie RDS-Open-Source-Engines, Graviton-Einführung und On-Demand-Abdeckung). Mithilfe dieser Benchmarks und KPIs können Sie überprüfen, ob Sie AWS-Services auf die kostengünstigste Weise nutzen.

Die folgende Tabelle enthält eine Liste von AWS-Standardmetriken als Referenz. Jede Organisation kann unterschiedliche Zielwerte für diese KPIs haben.

Kategorie	KPI (%)	Beschreibung
Datenverarbeitung	EC2-Nutzungsabdeckung	EC2-Instances (in Kosten oder Stunden), die SP+RI+Spot verwenden, im Vergleich zur Gesamtzahl (in Kosten oder Stunden) der EC2-Instances
Datenverarbeitung	Berechnung der SP/RI-Auslastung	Die genutzten SP- oder RI-Stunden im Vergleich zur Gesamtzahl der verfügbaren SP- oder RI-Stunden
Datenverarbeitung	EC2/Kosten pro Stunde	EC2-Kosten geteilt durch die Anzahl der EC2-Instances, die in dieser Stunde ausgeführt wurden
Datenverarbeitung	vCPU-Kosten	Kosten pro vCPU für alle Instances
Datenverarbeitung	Aktuelle Instance-Generation	Prozentsatz der Instances in Graviton (oder anderen Instance-Typen der modernen Generation)
Datenbank	RDS-Abdeckung	RDS-Instances (in Kosten oder Stunden), die RI verwenden, im Vergleich zur

Kategorie	KPI (%)	Beschreibung
		Gesamtzahl (in Kosten oder Stunden) der RDS-Instances
Datenbank	RDS-Nutzung	Die RI-Stunden im Vergleich zur Gesamtzahl der verfügbaren RI-Stunden
Datenbank	RDS-Betriebszeit	RDS-Kosten geteilt durch die Anzahl der RDS-Instances, die in dieser Stunde ausgeführt wurden
Datenbank	Aktuelle Instance-Generation	Prozentsatz der Instances in Graviton (oder anderen modernen Instance-Typen)
Speicher	Speichernutzung	Optimierte Speicherkosten (z. B. Glacier, Deep Archive oder Infrequent Access) geteilt durch die Gesamtspeicherkosten
Tagging	Ressourcen ohne Tags	Cost Explorer: 1. Filtern Sie Gutschriften, Rabatte, Steuern, Rückerstattungen und Marketplace heraus und kopieren Sie die aktuellen Monatskosten. 2. Wählen Sie in Cost Explorer die Option Nur Ressourcen ohne Tag anzeigen aus. 3. Teilen Sie den Betrag in Ressourcen ohne Tags durch Ihre monatlichen Kosten.

Geben Sie anhand dieser Tabelle die Ziel- oder Benchmarkwerte an, die auf der Grundlage Ihrer Organisationsziele berechnet werden sollten. Sie müssen bestimmte Metriken für Ihr Unternehmen messen und die Geschäftsergebnisse für diese Workload verstehen, um genaue und realistische KPIs definieren zu können. Unterscheiden Sie bei der Bewertung von Leistungsmetriken innerhalb einer Organisation zwischen verschiedenen Arten von Metriken, die unterschiedlichen Zwecken dienen. Mit diesen Metriken werden in erster Linie die Leistung und Effizienz der technischen Infrastruktur und nicht direkt die allgemeinen Auswirkungen auf das Geschäft gemessen. Es können beispielsweise die Reaktionszeiten des Servers, die Netzwerklatenz oder die Systemverfügbarkeit verfolgt werden. Diese Metriken sind entscheidend, um zu bewerten, wie gut die Infrastruktur den technischen Betrieb der Organisation unterstützt. Sie bieten jedoch keinen direkten Einblick in umfassendere Geschäftsziele wie Kundenzufriedenheit, Umsatzwachstum oder Marktanteil. Um ein umfassendes Verständnis der Unternehmensleistung zu erhalten, ergänzen Sie diese Effizienzmetriken durch strategische Geschäftsmetriken, die direkt mit den Geschäftsergebnissen korrelieren.

Verschaffen Sie sich einen Überblick nahezu in Echtzeit über Ihre KPIs und die damit verbundenen Einsparmöglichkeiten und verfolgen Sie Ihre Fortschritte im Laufe der Zeit. Um mit der Definition und Verfolgung von KPI-Zielen zu beginnen, empfehlen wir das KPI-Dashboard von [Cloud Intelligence Dashboards](#) (CID). Basierend auf den Daten aus dem Kosten- und Nutzungsbericht (CUR) bietet das KPI-Dashboard eine Reihe von empfohlenen KPIs zur Kostenoptimierung an. Außerdem können Sie benutzerdefinierte Ziele festlegen und den Fortschritt im Laufe der Zeit verfolgen.

Wenn Sie die KPI-Ziele mit anderen Lösungen festlegen und verfolgen, achten Sie darauf, dass diese Methoden von allen Stakeholdern im Cloud-Finanzmanagement in Ihrer Organisation übernommen werden.

Implementierungsschritte

- **Definieren der erwarteten Nutzungsgrade:** Konzentrieren Sie sich zu Beginn auf die Nutzungsgrade. Sprechen Sie mit den Anwendungsbesitzern, der Marketingabteilung und größeren Geschäftsteams, um zu verstehen, wie die erwartete Nutzung für die Workload aussieht. Wie könnte sich die Kundennachfrage im Laufe der Zeit ändern und was kann sich aufgrund saisonaler Anstiege oder Marketingkampagnen ändern?
- **Definieren von Ressourcen und Kosten für Workloads:** Mit den definierten Nutzungsgraden quantifizieren Sie die Änderungen der Workload-Ressourcen, die erforderlich sind, um diese Nutzungsgrade zu erfüllen. Möglicherweise müssen Sie den Umfang oder die Anzahl der Ressourcen für eine Workload-Komponente und die Datenübertragung erhöhen oder Workload-Komponenten in einen anderen Service auf einer bestimmten Ebene ändern. Geben Sie die Kosten

an jedem dieser Hauptpunkte an und prognostizieren Sie die Kostenänderung, wenn sich die Nutzung ändert.

- **Definieren von Geschäftszielen:** Nehmen Sie die Ergebnisse zu den erwarteten Änderungen bei Nutzung und Kosten, kombinieren Sie sie mit den erwarteten Änderungen bei der Technologie oder sonstigen Programmen, die Sie ausführen, und entwickeln Sie Ziele für die Workload. Die Ziele müssen Nutzung und Kosten sowie das Verhältnis zwischen beiden berücksichtigen. Die Ziele müssen einfach und allgemein gehalten sein und den Mitarbeitern helfen zu verstehen, was das Unternehmen an Ergebnissen erwartet (z. B. sicherzustellen, dass ungenutzte Ressourcen unter einem bestimmten Kostenniveau gehalten werden). Sie müssen nicht für jeden ungenutzten Ressourcentyp Ziele definieren oder Kosten festlegen, die Verluste für Ziele und Vorgaben verursachen können. Überprüfen Sie, ob es organisatorische Programme gibt (z. B. Kompetenzaufbau wie Schulungen und Fortbildungen), wenn Kostenänderungen ohne veränderte Nutzung zu erwarten sind.
- **Definieren der Ergebnisse:** Geben Sie für jedes der definierten Ziele ein messbares Ergebnis an. Wenn das Ziel darin besteht, die Effizienz der Workload zu erhöhen, sollte mit der Vorgabe der Umfang der Verbesserung (in der Regel in Form von Geschäftsergebnissen für jeden ausgegebenen Dollar) und der Zeitpunkt der Erreichung dieses Ziels angegeben werden. Sie könnten sich zum Beispiel das Ziel setzen, Verschwendung aufgrund einer Überversorgung zu minimieren. Bei diesem Ziel kann Ihre Vorgabe darin bestehen, dass die Verschwendung aufgrund einer zu hohen Datenverarbeitungsleistung in der ersten Stufe der Produktionsworkloads 10 Prozent der Datenverarbeitungskosten auf dieser Stufe nicht überschreiten sollte. Darüber hinaus könnte eine zweite Vorgabe darin bestehen, dass die Verschwendung aufgrund einer zu hohen Datenverarbeitungsleistung in der zweiten Stufe der Produktionsworkloads 5 Prozent der Datenverarbeitungskosten auf dieser Stufe nicht überschreiten sollte.

Ressourcen

Zugehörige Dokumente:

- [Von AWS verwaltete Richtlinien für Auftragsfunktionen](#)
- [AWS-Fakturierungsstrategie mit mehreren Konten](#)
- [Steuern des Zugriffs auf AWS-Regionen mit IAM-Richtlinien](#)
- [S.M.A.R.T.-Ziele](#)
- [How to track your cost optimization KPIs with the CID KPI Dashboard](#)

Zugehörige Videos:

- [Well-Architected Labs: Goals and Targets \(Level 100\)](#)

Zugehörige Beispiele:

- [What is a unit metric?](#)
- [Selecting a unit metric to support your business](#)
- [Unit metrics in practice – lessons learned](#)
- [How unit metrics help create alignment between business functions](#)

COST02-BP03 Implementieren einer Kontenstruktur

Implementieren Sie eine Kontenstruktur, die für Ihre Organisation geeignet ist. Dadurch werden die Zuweisung und Verwaltung der Kosten in der gesamten Organisation erleichtert.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Mit AWS Organizations können Sie mehrere AWS-Konten erstellen und so Ihre Umgebung zentral verwalten, wenn Sie Workloads in AWS skalieren. Sie können Ihre Organisationshierarchie modellieren, indem Sie AWS-Konten in einer Struktur von Organisationseinheiten (OEs) gruppieren und mehrere AWS-Konten in jeder OE erstellen. Um eine Kontostruktur zu erstellen, müssen Sie zuerst entscheiden, welches Ihrer AWS-Konten das Verwaltungskonto sein soll. Danach können Sie auf Grundlage der geplanten Kontostruktur neue AWS-Konten erstellen oder vorhandene Konten als Mitgliedskonten auswählen. Beachten Sie dabei [bewährte Methoden für Verwaltungskonten](#) und für [Mitgliedskonten](#).

Sie sollten immer mindestens ein Verwaltungskonto mit einem verknüpften Mitgliedskonto haben, unabhängig von der Organisationsgröße oder Nutzung. Alle Workload-Ressourcen sollten sich nur in Mitgliedskonten befinden. In Verwaltungskonten sollten keine Ressourcen erstellt werden. Es gibt keine einheitliche Antwort darauf, über wie viele AWS-Konten Sie verfügen sollten. Zunächst sollten Sie Ihre aktuellen und künftigen Betriebs- und Kostenmodelle bewerten, um sicherzustellen, dass die Struktur Ihrer AWS-Konten die Ziele Ihrer Organisation widerspiegelt. Einige Organisationen erstellen aus geschäftlichen Gründen mehrere AWS-Konten, z. B.:

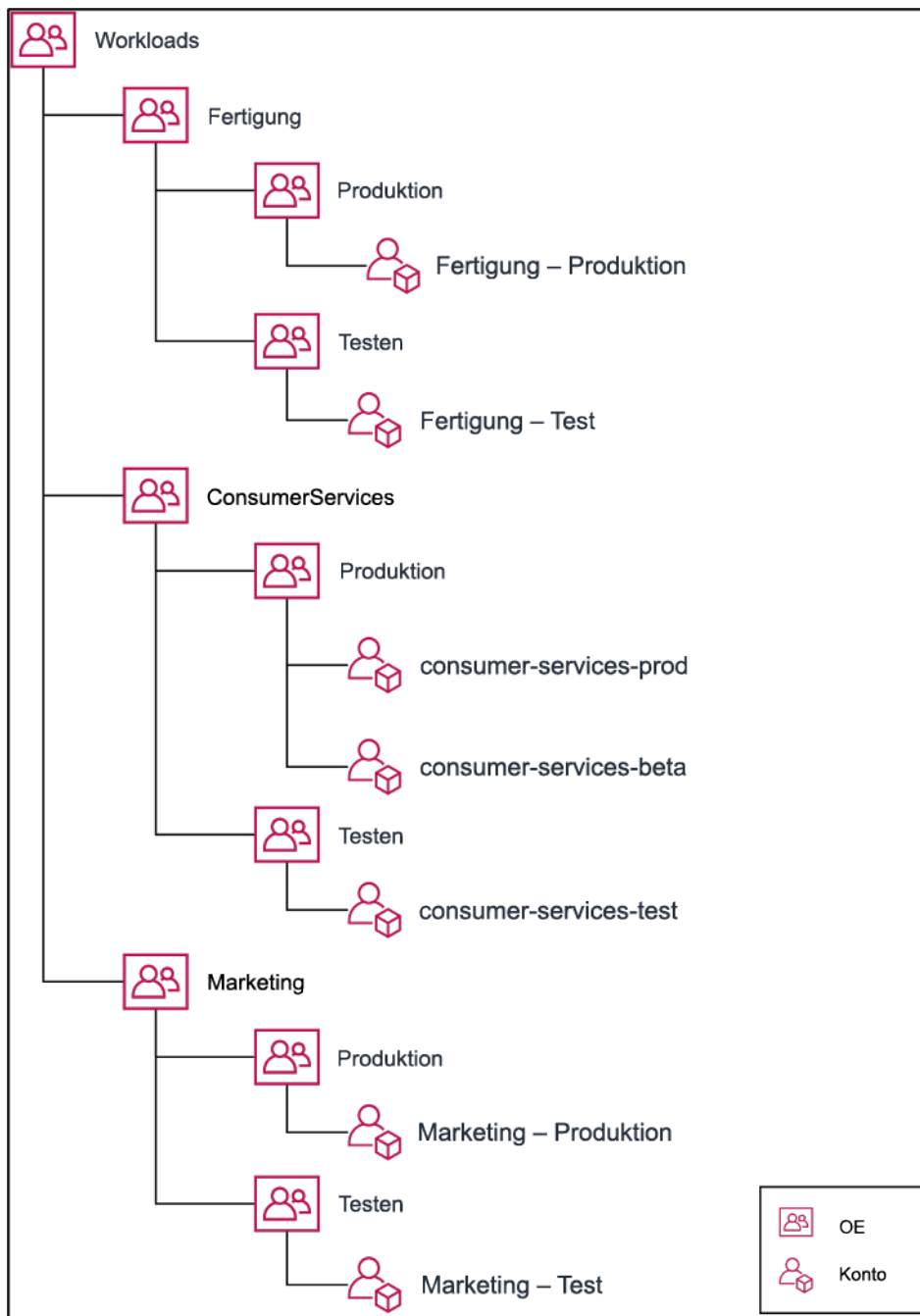
- Es ist eine administrative oder fiskale und fakturierungsbezogene Abgrenzung zwischen Organisationseinheiten, Kostenstellen oder spezifischen Workloads erforderlich.

- AWS-Service-Limits wurden für bestimmte Workloads definiert.
- Es besteht eine Anforderung für Isolierung und Trennung zwischen Workloads und Ressourcen.

Innerhalb von [AWS Organizations](#) erstellt die [konsolidierte Fakturierung](#) das Konstrukt zwischen einem oder mehreren Mitgliedskonten und dem Verwaltungskonto. Mit Mitgliedskonten können Sie Ihre Kosten und Nutzung nach Gruppen isolieren und unterscheiden. In diesem Kontext hat es sich bewährt, separate Mitgliedskonten für jede Organisationseinheit (z. B. Finanzen, Marketing und Vertrieb) oder für jeden Umgebungslebenszyklus (z. B. Entwicklung, Tests und Produktion) oder für jede einzelne Workload (Workload a, b und c) zu erstellen und diese verknüpften Konten dann über die konsolidierte Fakturierung zu aggregieren.

Mit der konsolidierten Fakturierung können Sie die Zahlung für mehrere AWS-Konten unter einem einzelnen Verwaltungskonto konsolidieren und dabei weiterhin die Sichtbarkeit für die Aktivitäten jedes verknüpften Kontos bereitstellen. Da Kosten und Nutzung im Verwaltungskonto aggregiert werden, können Sie sowohl Ihre Service-Volume-Rabatte als auch die Nutzung Ihrer an feste Kapazität gebundene Rabatte (Savings Plans und Reserved Instances) maximieren und so die höchsten Vergünstigungen erzielen.

Im folgenden Diagramm wird gezeigt, wie Sie AWS Organizations mit Organisationseinheiten (OEs) verwenden können, um mehrere Konten zu gruppieren und mehrere AWS-Konten unter jeder OE zu platzieren. Sie sollten OEs für unterschiedliche Anwendungsfälle und Workloads verwenden, die Muster für die Organisation von Konten vorgeben.



Beispiel zum Gruppieren mehrerer AWS-Konten unter Organisationseinheiten

[AWS Control Tower](#) kann schnell mehrere AWS-Konten einrichten und konfigurieren, um sicherzustellen, dass die Governance den Anforderungen Ihrer Organisation entspricht.

Implementierungsschritte

- Definieren von Trennungsanforderungen: Die Trennungsanforderungen sind eine Kombination aus mehreren Faktoren, darunter fallen Sicherheit, Zuverlässigkeit und finanzielle Konstrukte. Arbeiten

Sie die einzelnen Faktoren in der richtigen Reihenfolge durch und geben Sie an, ob die Workload oder die Workload-Umgebung von anderen Workloads getrennt sein sollte. Bei der Sicherheit steht die Einhaltung der Anforderungen an Zugriff und Daten im Vordergrund. Zuverlässigkeit bezieht sich auf die Verwaltung von Limits, sodass Umgebungen und Workloads keine Auswirkungen auf andere Elemente haben. Gehen Sie die Säulen „Sicherheit“ und „Zuverlässigkeit“ des Well-Architected-Framework regelmäßig durch und halten Sie sich an die angegebenen bewährten Methoden. Finanzielle Konstrukte schaffen eine strikte Trennung im Bereich der Finanzen (verschiedene Kostenstellen, Verantwortlichkeiten für die Workloads und Rechenschaftspflicht). Häufige Beispiele für die Trennung sind Produktions- und Test-Workloads, die in separaten Konten ausgeführt werden, oder die Verwendung eines separaten Kontos, sodass die Rechnungs- und Fakturierungsdaten den verschiedenen Organisationseinheiten oder Abteilungen in der Organisation oder dem Stakeholder bereitgestellt werden können, dem das Konto gehört.

- Definieren von Gruppierungsanforderungen: Die Anforderungen für die Gruppierung überschreiben die Trennungsanforderungen nicht, sondern dienen zur Unterstützung der Verwaltung. Gruppieren Sie ähnliche Umgebungen oder Workloads, die keine Trennung erfordern. Ein Beispiel hierfür ist die Gruppierung mehrerer Test- oder Entwicklungsumgebungen aus einer oder mehreren Workloads.
- Definieren der Kontenstruktur: Geben Sie mit diesen Trennungen und Gruppierungen ein Konto für jede Gruppe an und stellen Sie sicher, dass die Trennungsanforderungen erfüllt werden. Diese Konten sind Ihre Mitgliedskonten oder verknüpfte Konten. Indem Sie diese Mitgliedskonten unter einem einzigen Verwaltungs-/Zahlungskonto gruppieren, kombinieren Sie die Nutzung. Dies ermöglicht höhere Mengenrabatte für alle Konten und Sie erhalten eine gemeinsame Rechnung für alle Konten. Es ist möglich, Fakturierungsdaten zu trennen und jedem Mitgliedskonto eine individuelle Ansicht ihrer Fakturierungsdaten bereitzustellen. Definieren Sie mehrere Verwaltungs-/Zahlungskonten, wenn die Nutzungs- oder Fakturierungsdaten eines Mitgliedskontos für kein anderes Konto sichtbar sein dürfen oder wenn eine separate Rechnung von AWS erforderlich ist. In diesem Fall hat jedes Mitgliedskonto ein eigenes Verwaltungs-/Zahlungskonto. Ressourcen sollten immer in Mitgliedskonten oder verknüpften Konten platziert werden. Die Verwaltungs-/Zahlungskonten sollten nur für die Verwaltung verwendet werden.

Ressourcen

Zugehörige Dokumente:

- [Verwendung von Kostenzuordnungs-Tags](#)
- [Von AWS verwaltete Richtlinien für Auftragsfunktionen](#)

- [AWS-Fakturierungsstrategie mit mehreren Konten](#)
- [Steuern des Zugriffs auf AWS-Regionen mit IAM-Richtlinien](#)
- [AWS Control Tower](#)
- [AWS Organizations](#)
- Bewährte Methoden für [Verwaltungskonten](#) und [Mitgliedskonten](#)
- [Organisieren Sie Ihre AWS-Umgebung mit mehreren Konten](#)
- [Aktivieren von geteilten reservierten Instances und Savings Plans-Rabatten](#)
- [Konsolidierte Fakturierung](#)
- [Konsolidierte Fakturierung](#)

Zugehörige Beispiele:

- [Splitting the CUR and Sharing Access](#)

Zugehörige Videos:

- [Introducing AWS Organizations](#)
- [Set Up a Multi-Account AWS Environment that Uses Best Practices for AWS Organizations](#)

Zugehörige Beispiele:

- [Defining an AWS Multi-Account Strategy for telecommunications companies](#)
- [Best Practices for Optimizing AWS-Konten](#)
- [Best Practices for Organizational Units with AWS Organizations](#)

COST02-BP04 Implementieren von Gruppen und Rollen

Implementieren Sie Gruppen und Rollen, die Ihren Richtlinien entsprechen, und steuern Sie, wer Instances und Ressourcen in jeder Gruppe erstellen, ändern oder außer Betrieb nehmen kann. Implementieren Sie beispielsweise Entwicklungs-, Test- und Produktionsgruppen. Dies gilt sowohl für AWS-Services als auch für Lösungen anderer Anbieter.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Niedrig

Implementierungsleitfaden

Benutzerrollen und -gruppen sind grundlegende Bausteine bei der Entwicklung und Implementierung sicherer und effizienter Systeme. Rollen und Gruppen helfen Organisationen dabei, den Bedarf an Kontrolle mit den Anforderungen an Flexibilität und Produktivität in Einklang zu bringen, um letztlich die Organisationsziele und die Bedürfnisse der Benutzer zu unterstützen. Wie im Abschnitt [Identity and Access Management](#) der Säule „Sicherheit“ des AWS Well-Architected Framework empfohlen, benötigen Sie eine robuste Identitätsverwaltung und Berechtigungen, um den richtigen Personen unter den richtigen Bedingungen Zugriff auf die richtigen Ressourcen zu gewähren. Die Benutzer erhalten nur den Zugriff, den sie zur Erfüllung ihrer Aufgaben benötigen. Auf diese Weise wird das Risiko eines nicht autorisierten Zugriffs oder Missbrauchs minimiert.

Nachdem Sie Richtlinien entwickelt haben, können Sie logische Gruppen und Rollen von Benutzern innerhalb Ihrer Organisation erstellen. Auf diese Weise können Sie Berechtigungen zuweisen, die Nutzung kontrollieren und robuste Zugriffskontrollmechanismen implementieren, die den nicht autorisierten Zugriff auf sensible Informationen verhindern. Beginnen Sie mit allgemeinen Personengruppen. Dies entspricht in der Regel den Organisationseinheiten und beruflichen Rollen (z. B. ein Systemadministrator in der IT-Abteilung, ein Financial Controller oder ein Geschäftsanalyst). Den Gruppen treten Personen bei, die ähnliche Aufgaben ausführen und ähnlichen Zugriff benötigen. Rollen definieren, was eine Gruppe tun muss. Es ist einfacher, Berechtigungen für Gruppen und Rollen zu verwalten als für einzelne Benutzer. Rollen und Gruppen weisen allen Benutzern konsistent und systematisch Berechtigungen zu und verhindern so Fehler und Inkonsistenzen.

Wenn sich die Rolle eines Benutzers ändert, können Administratoren den Zugriff auf Rollen- oder Gruppenebene anpassen, anstatt einzelne Benutzerkonten neu zu konfigurieren. Beispielsweise benötigt ein Systemadministrator in der IT Zugriff, um alle Ressourcen zu erstellen, aber ein Analytikteammitglied muss nur Analytikressourcen erstellen.

Implementierungsschritte

- Implementieren von Gruppen: Implementieren Sie bei Bedarf die entsprechenden Gruppen mithilfe der in Ihren Organisationsrichtlinien definierten Benutzergruppen. Bewährte Methoden für Benutzer, Gruppen und Authentifizierung finden Sie unter der [Säule „Sicherheit“](#) des AWS Well-Architected Framework.
- Implementieren von Rollen und Richtlinien: Erstellen Sie mithilfe der Aktionen, die in Ihren Organisationsrichtlinien definiert sind, die erforderlichen Rollen und Zugriffsrichtlinien. Bewährte Methoden zu Rollen und Richtlinien finden Sie unter der [Säule „Sicherheit“](#) des AWS Well-Architected Framework.

Ressourcen

Zugehörige Dokumente:

- [Von AWS verwaltete Richtlinien für Auftragsfunktionen](#)
- [AWS-Fakturierungsstrategie mit mehreren Konten](#)
- [Säule „Sicherheit“ des AWS Well-Architected Framework](#)
- [AWS Identity and Access Management \(IAM\)](#)
- [AWS Identity and Access Management-Richtlinien](#)

Zugehörige Videos:

- [Why use Identity and Access Management](#)

Zugehörige Beispiele:

- [Steuern des Zugriffs auf AWS-Regionen mit IAM-Richtlinien](#)
- [Starting your Cloud Financial Management journey: Cloud cost operations](#)

COST02-BP05 Implementieren von Kostenkontrollen

Implementieren Sie Kontrollmechanismen, die auf den Organisationsrichtlinien sowie auf definierten Gruppen und Rollen basieren. Damit wird sichergestellt, dass nur Kosten im Rahmen der festgelegten Organisationsanforderungen anfallen, z. B. durch Steuerung des Zugriffs auf Regionen oder Ressourcentypen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Ein häufiger erster Schritt bei der Implementierung von Kostenkontrollen ist die Einrichtung von Benachrichtigungen, wenn im Zusammenhang mit Kosten oder Nutzung Ereignisse auftreten, die den Richtlinien nicht entsprechen. Auf diese Weise können Sie schnell agieren und überprüfen, ob Korrekturmaßnahmen erforderlich sind, ohne dass Workloads oder neue Aktivitäten eingeschränkt oder beeinträchtigt werden. Nachdem Sie die Limits für Workloads und Umgebung kennen, können Sie Governance durchsetzen. Mit [AWS Budgets](#) lassen sich Benachrichtigungen festlegen und monatliche Budgets für AWS-Kosten, Nutzung und an feste Kapazität gebundene Rabatte definieren

(Savings Plans und Reserved Instances). Sie können Budgets auf aggregierter Kostenebene (z. B. alle Kosten) oder auf einer detaillierteren Ebene erstellen, in der Sie nur bestimmte Dimensionen wie verknüpfte Konten, Services, Tags oder Availability Zones einschließen.

Wenn Sie die Budgetlimits mit AWS Budgets eingerichtet haben, können Sie mit [AWS Cost Anomaly Detection](#) unerwartete Kosten reduzieren. AWS Cost Anomaly Detection ist ein Kostenmanagementservice, der mithilfe von Machine Learning Ihre Kosten und Nutzung ständig überwacht, um ungewöhnliche Ausgaben zu erkennen. So können Sie untypische Ausgaben und ihre Ursachen schnell identifizieren und so schnell Maßnahmen ergreifen. Erstellen Sie zuerst eine Kostenüberwachung in AWS Cost Anomaly Detection und wählen Sie dann aus, wann Sie gewarnt werden möchten. Hierzu richten Sie einen Schwellenwert in Dollar ein und können sich z. B. bei Unregelmäßigkeiten benachrichtigen lassen, deren Auswirkungen 1.000 \$ überschreiten. Wenn Sie Warnungen erhalten, können Sie die Ursachen hinter den Unregelmäßigkeiten und deren wirtschaftliche Auswirkungen analysieren. Sie können Unregelmäßigkeiten in AWS Cost Explorer auch selbst überwachen und analysieren.

Setzen Sie Governance-Richtlinien in AWS durch [AWS Identity and Access Management](#) und [Service-Kontrollrichtlinien \(SCP\) von AWS Organizations](#) durch. Mit IAM können Sie den Zugriff auf AWS-Services und -Ressourcen sicher verwalten. Mit AWS können Sie steuern, wer -Ressourcen erstellen und verwalten kann, welche Art von Ressourcen erstellt werden kann und wo sie erstellt werden können. So wird die Möglichkeit eingeschränkt, Ressourcen außerhalb der definierten Richtlinie zu erstellen. Verwenden Sie die zuvor erstellten Rollen und Gruppen und weisen Sie [IAM-Richtlinien](#) zu, um die korrekte Nutzung zu erzwingen. SCP bietet eine zentrale Kontrolle über die maximal verfügbaren Berechtigungen für alle Konten in Ihrer Organisation, damit Ihre Konten die Vorgaben Ihrer Zugriffskontrollrichtlinien erfüllen. SCPs sind nur in einer Organisation verfügbar, für die alle Features aktiviert sind, und Sie können die SCPs so konfigurieren, dass sie Aktionen für Mitgliedskonten standardmäßig verweigern oder zulassen. Weitere Informationen zur Implementierung des Zugriffsmanagements finden Sie im [Well-Architected-Whitepaper zur Säule „Sicherheit“](#).

Governance kann auch durch die Verwaltung von [AWS Service Quotas](#) implementiert werden. Indem Sie sicherstellen, dass Service Quotas mit minimalem Overhead definiert und ordnungsgemäß verwaltet werden, können Sie die Ressourcenerstellung über die Geschäftsanforderungen hinaus minimieren. Dazu müssen Sie nachvollziehen, wie schnell sich Ihre Anforderungen ändern können, Sie müssen die derzeit ausgeführten Projekte kennen – in Bezug auf die Erstellung und die Deaktivierung von Ressourcen – und berücksichtigen, wie schnell Kontingentänderungen implementiert werden können. [Service Quotas](#) können bei Bedarf verwendet werden, um Ihre Kontingente zu erhöhen.

Implementierungsschritte

- Implementieren von Benachrichtigungen zu Ausgaben: Erstellen Sie mithilfe Ihrer definierten Organisationsrichtlinien [AWS Budgets](#), um Benachrichtigungen zu erhalten, wenn Ausgaben außerhalb Ihrer Richtlinien liegen. Konfigurieren Sie mehrere Kostenbudgets, eines für jedes Konto, um über die allgemeinen Kontoausgaben informiert zu werden. Konfigurieren Sie zusätzliche Kostenbudgets innerhalb jedes Kontos für kleinere Einheiten innerhalb des Kontos. Diese Einheiten variieren je nach Kontostruktur. Einige gängige Beispiele sind AWS-Regionen, Workloads (mit Tags) oder AWS-Services. Konfigurieren Sie eine E-Mail-Verteilerliste als Empfänger für Benachrichtigungen, nicht das E-Mail-Konto einer Person. Sie können ein tatsächliches Budget für den Fall konfigurieren, dass ein Betrag überschritten wird, oder ein prognostiziertes Budget zur Benachrichtigung über die prognostizierte Nutzung verwenden. Sie können auch AWS-Budgetaktionen vorkonfigurieren, die bestimmte IAM- oder SCP-Richtlinien erzwingen, oder Amazon-EC2- oder Amazon-RDS-Ziel-Instances beenden. Budgetaktionen werden entweder automatisch gestartet oder erfordern eine Workflow-Genehmigung.
- Implementieren von Benachrichtigungen zu ungewöhnlichen Ausgaben: Mit [AWS Cost Anomaly Detection](#) können Sie unerwartete Kosten in Ihrer Organisation reduzieren und die Ursachen potenzieller ungewöhnlicher Ausgaben analysieren. Wenn Sie eine Kostenüberwachung zum Identifizieren ungewöhnlicher Ausgaben mit der angegebenen Granularität erstellen und Benachrichtigungen in AWS Cost Anomaly Detection konfigurieren, erhalten Sie eine Warnung, wenn eine ungewöhnliche Ausgabe erkannt wird. So können Sie die Ursache der Unregelmäßigkeit analysieren und erhalten Informationen zu den Auswirkungen auf Ihre Kosten. Verwenden Sie AWS-Kostenkategorien beim Konfigurieren von AWS Cost Anomaly Detection, um zu ermitteln, welches Projekt- oder Geschäftseinheitsteam die Ursache der unerwarteten Kosten analysieren und zeitnah die erforderlichen Maßnahmen ergreifen kann.
- Implementieren von Nutzungskontrollen: Implementieren Sie mithilfe Ihrer definierten Organisationsrichtlinien IAM-Richtlinien und -Rollen, um anzugeben, welche Aktionen Benutzer ausführen dürfen und welche nicht. In einer AWS-Richtlinie können mehrere Organisationsrichtlinien enthalten sein. Gehen Sie auf die gleiche Art und Weise vor, wie Sie Richtlinien definiert haben. Beginnen Sie umfassend und wenden dann bei jedem Schritt detailliertere Kontrollen an. Service Limits sind auch eine effektive Kontrolle der Nutzung. Implementieren Sie die richtigen Service Limits für alle Ihre Konten.

Ressourcen

Zugehörige Dokumente:

- [Von AWS verwaltete Richtlinien für Auftragsfunktionen](#)
- [AWS Fakturierungsstrategie mit mehreren Konten](#)
- [Steuern des Zugriffs auf AWS-Regionen mit IAM-Richtlinien](#)
- [AWS Budgets](#)
- [AWS Cost Anomaly Detection](#)
- [Behalten Sie Ihre AWS-Kosten im Griff](#)

Zugehörige Videos:

- [Wie kann ich AWS Budgets verwenden, um meine Ausgaben und Nutzung zu verfolgen?](#)

Zugehörige Beispiele:

- [Example IAM access management policies](#)
- [Example service control policies](#)
- [AWS Budgetaktionen](#)
- [Wie erstelle ich eine IAM-Richtlinie, um den Zugriff auf Amazon-EC2-Ressourcen mithilfe von Tags zu steuern?](#)
- [Kann ich den Zugriff einer IAM-Identität auf bestimmte Amazon-EC2-Ressourcen beschränken?](#)
- [Slack-Integrationen für die Erkennung von Kostenanomalien mit Amazon Q Developer in Chat-Anwendungen](#)

COST02-BP06 Projektlebenszyklus verfolgen

Verfolgen, bewerten und überprüfen Sie den Lebenszyklus von Projekten, Teams und Umgebungen, damit Sie keine unnötigen Ressourcen nutzen, für die Sie zahlen müssen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Niedrig

Implementierungsleitfaden

Durch eine effektive Nachverfolgung des Projektlebenszyklus können Organisationen durch verbesserte Planung, Verwaltung und Ressourcenoptimierung eine bessere Kostenkontrolle erreichen. Die durch die Nachverfolgung gewonnenen Erkenntnisse sind von unschätzbarem Wert, um fundierte Entscheidungen zu treffen, die zur Kosteneffizienz und zum Gesamterfolg des Projekts beitragen.

Die Verfolgung des gesamten Lebenszyklus der Workload hilft Ihnen zu verstehen, wann Workloads oder Workload-Komponenten nicht mehr benötigt werden. Die vorhandenen Workloads und Komponenten scheinen in Gebrauch zu sein, aber wenn neue Dienste oder Funktionen AWS veröffentlicht werden, können sie außer Betrieb genommen oder übernommen werden. Prüfen Sie die vorherigen Phasen der Workloads. Nachdem eine Workload in Betrieb genommen wurde, können frühere Umgebungen außer Betrieb genommen oder in ihrer Kapazität stark reduziert werden, bis sie wieder erforderlich sind.

Sie können Ressourcen mit einem Zeitrahmen oder einer Erinnerung versehen, um zu markieren, wann die Workload überprüft wurde. Wenn die Entwicklungsumgebung beispielsweise zuletzt vor Monaten überprüft wurde, könnte es ein guter Zeitpunkt sein, sie erneut zu überprüfen, um festzustellen, ob neue Services eingeführt werden können oder ob die Umgebung verwendet wird. Sie können Ihre Anwendungen gruppieren und mit einem [myApplications](#) Tag versehen AWS, um Metadaten wie Kritikalität, Umgebung, zuletzt überprüft und Kostenstelle zu verwalten und nachzuverfolgen. Sie können sowohl den Lebenszyklus Ihrer Workload verfolgen als auch die Kosten, den Zustand, den Sicherheitsstatus und die Leistung Ihrer Anwendungen überwachen und verwalten.

AWS bietet verschiedene Verwaltungs- und Governance-Dienste, die Sie für die Nachverfolgung des Lebenszyklus von Entitäten verwenden können. Sie können unseren [AWS Systems Manager](#) verwenden [AWS Config](#), um eine detaillierte Bestandsaufnahme Ihrer AWS Ressourcen und Konfiguration zu erstellen. Es wird empfohlen, dass Sie diese mit Ihren vorhandenen Projekt- bzw. Komponentenverwaltungssystemen integrieren, um aktive Projekte und Produkte in Ihrer Organisation zu verfolgen. Durch die Kombination Ihres aktuellen Systems mit den zahlreichen Ereignissen und Kennzahlen von AWS können Sie sich einen Überblick über wichtige Ereignisse im Lebenszyklus verschaffen und Ressourcen proaktiv verwalten, um unnötige Kosten zu reduzieren.

Ähnlich wie beim [Application Lifecycle Management \(ALM\)](#) sollten bei der Nachverfolgung des Projektlebenszyklus mehrere Prozesse, Tools und Teams zusammenarbeiten, z. B. Design und Entwicklung, Tests, Produktion, Support und Workload-Redundanz.

Durch die sorgfältige Überwachung jeder Phase des Lebenszyklus eines Projekts erhalten Organisationen entscheidende Einblicke und eine bessere Kontrolle, was die erfolgreiche Planung, Implementierung und den Abschluss von Projekten erleichtert. Mit dieser sorgfältigen Überwachung wird sichergestellt, dass die Projekte nicht nur den Qualitätsstandards entsprechen, sondern auch pünktlich und innerhalb des Budgets fertiggestellt werden, was die Kosteneffizienz insgesamt fördert.

Weitere Informationen zur Implementierung der Verfolgung des Lebenszyklus von Entitäten finden Sie im [Whitepaper zur Säule „Operative Exzellenz“ des AWS Well-Architected Framework](#).

Implementierungsschritte

- Einrichtung eines Prozesses zur Überwachung des Projektlebenszyklus: Das [Team des Cloud-Kompetenzzentrums](#) hat die Aufgabe, einen Prozess für die Überwachung des Projektlebenszyklus einzurichten. Entwickeln Sie einen strukturierten und systematischen Ansatz zur Überwachung der Workloads, um die Kontrolle, die Sichtbarkeit und die Leistung der Projekte zu verbessern. Sorgen Sie dafür, dass der Überwachungsprozess transparent und kooperativ ist und sich auf kontinuierliche Verbesserungen konzentriert, um seine Effektivität und seinen Wert zu maximieren.
- Durchführen von Workload-Überprüfungen: Richten Sie, wie in Ihren Organisationsrichtlinien festgelegt, einen regelmäßigen Rhythmus ein, um Audits Ihrer bestehenden Projekte und Überprüfungen von Workloads durchzuführen. Der Aufwand für die Prüfung sollte proportional zum ungefähren Risiko, dem Wert oder den Kosten für die Organisation sein. Wichtige Bereiche, die in die Prüfung aufgenommen werden sollen, sind das Risiko eines Vorfalls oder eines Ausfalls, der Wert oder Beitrag für die Organisation (gemessen am Umsatz oder Ruf der Marke), die Kosten der Workload (gemessen als Gesamtkosten für Ressourcen und Betriebskosten) und die Nutzung der Workload (gemessen an der Anzahl der Ergebnisse der Organisation pro Zeiteinheit). Wenn sich diese Bereiche im Laufe des Lebenszyklus ändern, sind Anpassungen der Workload erforderlich, z. B. die vollständige oder teilweise Außerbetriebnahme.

Ressourcen

Zugehörige Dokumente:

- [Leitfaden für das Markieren von AWS](#)
- [Was ist ALM \(Application Lifecycle Management\)?](#)
- [Von AWS verwaltete Richtlinien für Auftragsfunktionen](#)

Zugehörige Beispiele:

- [Steuern Sie den Zugriff AWS-Regionen mithilfe von IAM Richtlinien](#)

Zugehörige Tools

- [AWS Config](#)
- [AWS Systems Manager](#)
- [AWS Budgets](#)

- [AWS Organizations](#)
- [AWS CloudFormation](#)

KOSTEN 3. Wie überwachen Sie Ihre Kosten und die Nutzung?

Definieren Sie Richtlinien und Verfahren, um Ihre Kosten überwachen und richtig zuordnen zu können. So können Sie die Kosteneffizienz einer Workload messen und verbessern.

Best Practices

- [COST03-BP01 Konfigurieren detaillierter Informationsquellen](#)
- [COST03-BP02 Hinzufügen von Organisationsinformationen zu Kosten und Nutzung](#)
- [COST03-BP03 Identifizieren von Kostenzuordnungskategorien](#)
- [COST03-BP04 Organisationskennzahlen festlegen](#)
- [COST03-BP05 Konfigurieren von Tools für die Fakturierung und Kostenverwaltung](#)
- [COST03-BP06 Zuteilen von Kosten basierend auf Workload-Metriken](#)

COST03-BP01 Konfigurieren detaillierter Informationsquellen

Richten Sie Kostenmanagement- und Berichtstools ein, um die Analyse und Transparenz von Kosten- und Nutzungsdaten zu verbessern. Konfigurieren Sie Ihre Workload so, dass Protokolleinträge erstellt werden, die die Nachverfolgung und Segmentierung von Kosten und Nutzung erleichtern.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Detaillierte Abrechnungsinformationen, z. B. durch Aufschlüsselung nach Stunden in Kostenmanagement-Tools, ermöglichen es Organisationen, ihre Ressourcennutzung anhand weiterer Details zu verfolgen und einige der Gründe für den Kostenanstieg zu identifizieren. Diese Datenquellen bieten die genaueste Ansicht der Kosten und Nutzung in Ihrer gesamten Organisation.

Sie können AWS Data Exports verwenden, um Exporte von AWS Cost and Usage Report (CUR) 2.0 zu erstellen. Dies ist die neue und empfohlene Methode, um Ihre detaillierten Kosten- und Nutzungsdaten von AWS abzurufen. Sie bietet tägliche oder stündliche Nutzungsaufschlüsselung, Tarife, Kosten und Nutzungsattribute für alle kostenpflichtigen AWS-Services (dieselben

Informationen wie CUR), zusammen mit einigen Verbesserungen. Alle möglichen Dimensionen befinden sich im CUR, z. B. Tagging, Speicherort, Ressourcenattribute und Konto-IDs.

Je nach Art des Exports, den Sie erstellen möchten, gibt es drei Exporttypen: einen Standarddatenexport, einen Export in ein Kosten- und Nutzungs-Dashboard mit QuickSight-Integration und einen Legacy-Datenexport.

- Standarddatenexport: Ein benutzerdefinierter Export einer Tabelle, der regelmäßig an Amazon S3 gesendet wird.
- Kosten- und Nutzungs-Dashboard: Ein Export und eine Integration in QuickSight zur Bereitstellung eines vorgefertigten Kosten- und Nutzungs-Dashboards.
- Legacy-Datenexport: Ein Export des älteren AWS Cost and Usage Report (CUR).

Sie können Datenexporte mit den folgenden Anpassungen erstellen:

- Ressourcen-IDs einschließen
- Daten zur Zuordnung geteilter Kosten
- Stundenaufschlüsselung
- Versionsverwaltung
- Komprimierungstyp und Dateiformat

Aktivieren Sie für Ihre Workloads, die Container in Amazon ECS oder Amazon EKS ausführen, Daten zur Aufteilung geteilter Kosten, sodass Sie Ihre Containerkosten einzelnen Geschäftseinheiten und Teams zuordnen können, je nachdem, wie Ihre Container-Workloads gemeinsam genutzte Datenverarbeitungs- und Speicherressourcen verbrauchen. Mit Daten zur Aufteilung geteilter Kosten werden Kosten- und Nutzungsdaten für neue Ressourcen auf Containerebene in AWS Cost and Usage Report aufgenommen. Die Daten zur Aufteilung geteilter Kosten werden berechnet, indem die Kosten der einzelnen ECS-Services und Aufgaben berechnet werden, die auf dem Cluster ausgeführt werden.

Ein Kosten- und Nutzungs-Dashboard exportiert die Tabelle des Kosten- und Nutzungs-Dashboards regelmäßig in einen S3-Bucket und stellt ein vorgefertigtes Kosten- und Nutzungs-Dashboard in QuickSight bereit. Verwenden Sie diese Option, wenn Sie schnell ein Dashboard mit Ihren Kosten- und Nutzungsdaten bereitstellen möchten. Mit dieser Option sind keine Anpassungen möglich.

Falls gewünscht, können Sie CUR weiterhin im Legacy-Modus exportieren, in den Sie andere Verarbeitungsservices integrieren können, z. B. [AWS Glue](#), um die Daten für die Analyse

vorzubereiten und Datenanalysen mit [Amazon Athena](#) durchzuführen, indem Sie SQL für die Datenabfrage verwenden.

Implementierungsschritte

- **Datenexporte erstellen:** Erstellen Sie benutzerdefinierte Exporte mit den gewünschten Daten und steuern Sie Ihr Exportschema. Erstellen Sie Fakturierungs- und Kostenmanagementdatenexporte mit einfachem SQL und zeigen Sie Ihre Abrechnungs- und Kostenmanagementdaten durch die Integration mit QuickSight an. Sie können Ihre Daten auch im Standardmodus exportieren, um Ihre Daten mit anderen Verarbeitungstools wie Amazon Athena zu analysieren.
- **Konfiguration des Kosten- und Nutzungsberichts:** Konfigurieren Sie über die Fakturierungskonsole mindestens einen Kosten- und Nutzungsbericht. Konfigurieren Sie einen Bericht mit stündlicher Granularität, der alle IDs und Ressourcen-IDs enthält. Sie können auch andere Berichte mit unterschiedlichen Granularitäten erstellen, um zusammenfassende Informationen bereitzustellen.
- **Konfiguration der stündlichen Granularität in Cost Explorer:** Aktivieren Sie in der Fakturierungskonsole Daten auf Stundenbasis und auf Ressourcenebene, um auf Kosten- und Nutzungsdaten der letzten 14 Tage mit stündlicher Granularität zuzugreifen.
- **Konfiguration der Anwendungsprotokollierung:** Überprüfen Sie, dass Ihre Anwendung jedes Geschäftsergebnis protokolliert, das sie liefert, sodass es nachverfolgt und gemessen werden kann. Stellen Sie sicher, dass die Granularität dieser Daten mindestens stündlich ist, damit sie mit der Aufschlüsselung der Kosten- und Nutzungsdaten übereinstimmt. Weitere Informationen zur Protokollierung und Überwachung finden Sie unter der [Well-Architected-Säule „Operative Exzellenz“](#).

Ressourcen

Zugehörige Dokumente:

- [AWS Data Exports](#)
- [AWS Glue](#)
- [Schnell](#)
- [Preisberechnung des AWS-Kostenmanagements](#)
- [Taggen von AWS-Ressourcen](#)
- [Analysieren der Kosten mit Cost Explorer](#)
- [Verwalten von AWS Cost and Usage Reports](#)

Zugehörige Beispiele:

- [AWS Account Setup](#)
- [Data Exports for AWS Billing and Cost Management](#)
- [AWS Cost Explorer Common Use Cases](#)

COST03-BP02 Hinzufügen von Organisationsinformationen zu Kosten und Nutzung

Definieren Sie ein auf Ihrer Organisation basierendes Markierungsschema, Workload-Attribute und Kostenzuordnungskategorien, damit Sie nach Ressourcen filtern und suchen oder die Kosten und Nutzung in Kostenverwaltungstools überwachen können. Implementieren Sie ein einheitliches Markieren aller Ressourcen, wenn möglich nach Zweck, Team, Umgebung oder anderen für Ihr Unternehmen relevanten Kriterien.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Implementieren Sie das [Taggen in AWS](#), um Organisationsinformationen zu Ihren Ressourcen hinzuzufügen, die dann zu Ihren Kosten- und Nutzungsinformationen hinzugefügt werden. Ein Tag ist ein Schlüssel-Wert-Paar – der Schlüssel ist definiert und muss innerhalb Ihrer Organisation eindeutig sein und der Wert ist für eine Gruppe von Ressourcen eindeutig. Ein Beispiel für ein Schlüssel-Wert-Paar ist der Schlüssel `Environment` mit dem Wert `Production`. Alle Ressourcen in der Produktionsumgebung verfügen über dieses Schlüssel-Wert-Paar. Mit Tags können Sie Ihre Kosten mit aussagekräftigen, relevanten Organisationsinformationen kategorisieren und nachverfolgen. Sie können Tags anwenden, die Organisationskategorien (z. B. Kostenstellen, Anwendungsnamen, Projekte oder Besitzer) darstellen und Workloads und Merkmale von Workloads (z. B. Test oder Produktion) identifizieren, um Ihre Kosten und Nutzung in Ihrer gesamten Organisation zuzuordnen.

Wenn Sie Tags auf Ihre AWS-Ressourcen anwenden (z. B. Amazon Elastic Compute Cloud-Instances oder Amazon Simple Storage Service-Buckets) und die Tags aktivieren, fügt AWS diese Informationen zu Ihren Kosten- und Nutzungsberichten hinzu. Sie können Berichte ausführen und Analysen für markierte und nicht markierte Ressourcen durchführen, um eine größere Compliance mit internen Kostenverwaltungsrichtlinien zu ermöglichen und eine genaue Zuordnung zu gewährleisten.

Mit der Erstellung und Implementierung eines Standards für AWS-Tagging für alle Konten in Ihrer Organisation können Sie Ihre AWS-Umgebungen auf konsistente und einheitliche Weise

verwalten und steuern. Verwenden Sie [Tag-Richtlinien](#) in AWS Organizations, um Regeln für die Verwendung von Tags für AWS-Ressourcen in Ihren Konten in AWS Organizations zu definieren. Mit Tag-Richtlinien können Sie problemlos einen standardisierten Ansatz für das Taggen von AWS-Ressourcen anwenden.

Mit [AWS Tag Editor](#) können Sie Tags für mehrere Ressourcen hinzufügen, löschen und verwalten. Mit Tag Editor können Sie nach den Ressourcen suchen, die Sie mit Tags markieren möchten, und dann die Tags für die Ressourcen in Ihren Suchergebnissen verwalten.

Mit [AWS Cost Categories](#) können Sie Ihren Kosten eine Organisationsbedeutung zuweisen, ohne dass Tags für Ressourcen erforderlich sind. Sie können Ihre Kosten- und Nutzungsinformationen eindeutigen internen Organisationsstrukturen zuordnen. Sie definieren Kategorieregeln, um Kosten mithilfe von Fakturierungsdimensionen wie Konten und Tags zuzuordnen und zu kategorisieren. Dies bietet zusätzlich zum Tagging eine weitere Ebene der Verwaltungsfunktionen. Sie können auch bestimmte Konten und Tags mehreren Projekten zuordnen.

Implementierungsschritte

- Definieren eines Tagging-Schemas: Versammeln Sie alle Beteiligten aus Ihrem gesamten Unternehmen, um ein Schema zu definieren. Dies umfasst in der Regel Mitarbeiter in technischen, finanziellen und leitenden Funktionen. Definieren Sie eine Liste der Tags, die alle Ressourcen haben müssen, sowie eine Liste der Tags, die Ressourcen haben sollten. Stellen Sie sicher, dass die Tag-Namen und -Werte in Ihrer Organisation konsistent sind.
- Taggen von Ressourcen: Platzieren Sie mithilfe Ihrer definierten Kostenzuordnungskategorien [Tags](#) für alle Ressourcen in Ihren Workloads entsprechend den Kategorien. Verwenden Sie Tools wie CLI, Tag Editor oder AWS Systems Manager, um die Effizienz zu steigern.
- Implementieren von AWS Cost Categories: Sie können [Kostenkategorien](#) erstellen, ohne Tagging zu implementieren. Kostenkategorien verwenden die vorhandenen Kosten- und Nutzungsdimensionen. Erstellen Sie Kategorieregeln aus Ihrem Schema und implementieren Sie diese in Kostenkategorien.
- Automatisiertes Tagging: Automatisieren Sie das Tagging, um sicherzustellen, dass Sie ein hohes Maß an Tagging für alle Ressourcen aufrechterhalten, damit Ressourcen automatisch bei ihrer Erstellung getaggt werden. Nutzen Sie Services wie [AWS CloudFormation](#), um zu überprüfen, ob die Ressourcen bei der Erstellung mit Tags versehen wurden. Sie können auch eine benutzerdefinierte Lösung für das automatische Taggen mithilfe von Lambda-Funktionen erstellen oder einen Microservice verwenden, der die Workload regelmäßig überprüft und alle nicht getaggten Ressourcen entfernt, was ideal für Test- und Entwicklungsumgebungen ist.

- Überwachung von und Berichterstellung zu Tags: Um sicherzustellen, dass Sie in Ihrer Organisation ein hohes Maß an Tagging aufrechterhalten, melden und überwachen Sie die Tags in Ihren Workloads. Sie können [AWS Cost Explorer](#) verwenden, um die Kosten für getaggte und nicht getaggte Ressourcen anzuzeigen. Alternativ können Sie auch Services wie [Tag Editor](#) verwenden. Überprüfen Sie regelmäßig die Anzahl der nicht getaggten Ressourcen und ergreifen Sie Maßnahmen, um Tags hinzuzufügen, bis Sie die gewünschte Tagging-Stufe erreichen.

Ressourcen

Zugehörige Dokumente:

- [Tagging Best Practices](#)
- [AWS CloudFormation Resource Tag](#)
- [AWS Kostenkategorien](#)
- [Taggen von AWS-Ressourcen](#)
- [Analysieren Ihrer Kosten mit AWS-Budgets](#)
- [Analysieren der Kosten mit Cost Explorer](#)
- [Verwalten von AWS-Kosten- und -Nutzungsberichten](#)

Zugehörige Videos:

- [How can I tag my AWS resources to divide up my bill by cost center or project](#)
- [Taggen von AWS-Ressourcen](#)

COST03-BP03 Identifizieren von Kostenzuordnungskategorien

Identifizieren Sie Organisationskategorien wie Geschäftsbereiche, Abteilungen oder Projekte, anhand derer die Kosten innerhalb Ihrer Organisation den internen Verbrauchern zugewiesen werden können. Verwenden Sie diese Kategorien, um ein Gefühl der Verantwortung für Ausgaben zu fördern, Bewusstsein für Kosten zu schaffen und ein effektives Nutzungsverhalten zu unterstützen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Der Prozess der Kostenkategorisierung ist für Budgetierung, Buchhaltung, Finanzberichterstattung, Entscheidungsfindung, Benchmarking und Projektmanagement von entscheidender Bedeutung.

Durch die Klassifizierung und Kategorisierung von Ausgaben können Teams die Arten von Kosten besser nachvollziehen, die auf dem Weg in die Cloud entstehen werden. So können sie fundierte Entscheidungen treffen und Budgets effektiv verwalten.

Die Rechenschaftspflicht bei den Cloud-Ausgaben ist ein starker Anreiz für ein diszipliniertes Nachfrage- und Kostenmanagement. Das Ergebnis sind deutlich höhere Cloud-Kosteneinsparungen für Organisationen, die den größten Teil ihrer Cloud-Ausgaben für verbrauchende Geschäftsbereiche oder Teams aufwenden. Darüber hinaus hilft die Zuweisung von Cloud-Ausgaben Organisationen dabei, mehr bewährte Methoden für die zentralisierte Cloud-Governance einzuführen.

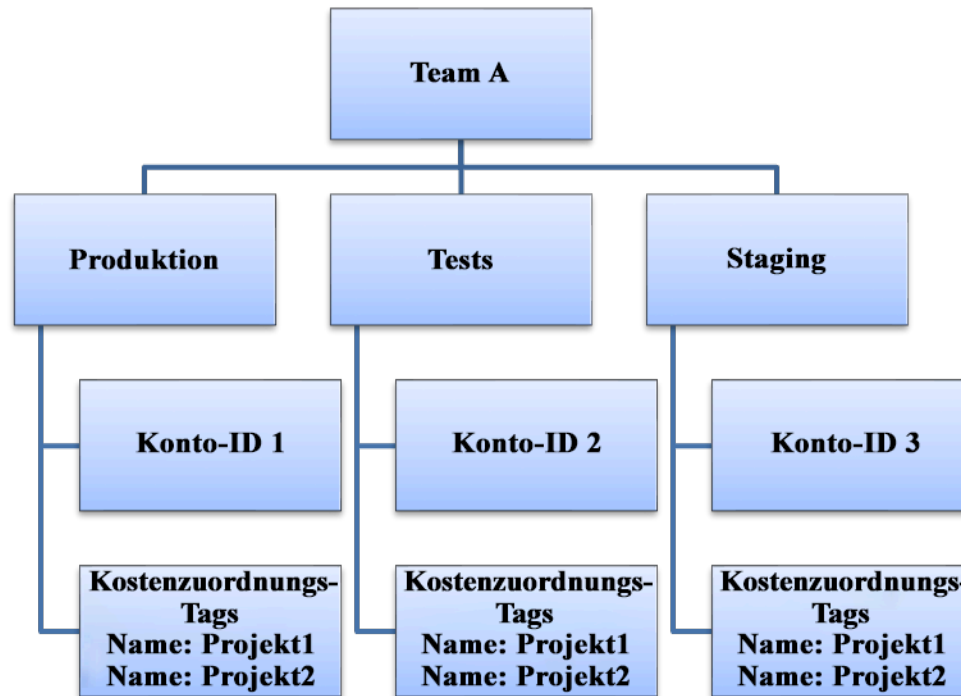
Arbeiten Sie in regelmäßigen Besprechungen mit Ihrem Finanzteam und anderen relevanten Stakeholdern zusammen, um zu verstehen, wie die Kosten innerhalb Ihrer Organisation zugeordnet werden müssen. Workload-Kosten müssen über den gesamten Lebenszyklus hinweg zugeordnet werden, einschließlich Entwicklung, Tests, Produktion und Außerbetriebnahme. Analysieren Sie, welche Kosten durch Schulungen, Personalentwicklung und Ideenentwicklung in der Organisation entstehen. Dies kann hilfreich sein, um Konten, die zu diesem Zweck verwendet werden, korrekt den Schulungs- und Entwicklungsbudgets zuzuordnen, anstatt allgemeinen IT-Kostenbudgets.

Nachdem Sie Ihre Kostenzuordnungskategorien mit Stakeholdern in Ihrer Organisation definiert haben, können Sie mit [AWS Cost Categories](#) Ihre Kosten- und Nutzungsinformationen in aussagekräftige Kategorien in der AWS Cloud gruppieren, z. B. Kosten für ein bestimmtes Projekt oder AWS-Konten für Abteilungen oder Geschäftsbereiche. Sie können benutzerdefinierte Kategorien erstellen und Ihre Kosten- und Nutzungsinformationen diesen Kategorien zuordnen, und zwar basierend auf Regeln, die Sie anhand verschiedener Dimensionen wie Konto, Tag, Service, oder Kostenart definieren. Sobald die Kostenkategorien eingerichtet sind, können Sie Ihre Kosten- und Nutzungsinformationen nach diesen Kategorien aufgeschlüsselt anzeigen, sodass Ihre Organisation bessere Strategie- und Kaufentscheidungen treffen kann. Diese Kategorien sind auch in AWS Cost Explorer, AWS Budgets und AWS Cost and Usage Report sichtbar.

Erstellen Sie beispielsweise Kostenkategorien für Ihre Geschäftseinheiten (DevOps-Team) und erstellen Sie unter jeder Kategorie mehrere Regeln (Regeln für jede Unterkategorie) mit mehreren Dimensionen (AWS-Konten, Kostenzuordnungs-Tags, Services oder Kostenart) basierend auf den von Ihnen definierten Gruppierungen. Sie können mit Cost Categories Ihre Kosten mithilfe einer regelbasierten Engine organisieren. Die von Ihnen konfigurierten Regeln unterteilen Ihre Kosten in Kategorien. Innerhalb dieser Regeln können Sie mithilfe mehrerer Dimensionen für jede Kategorie filtern, z. B. nach bestimmten AWS-Konten, bestimmten AWS-Services oder bestimmten Kostenarten. Sie können diese Kategorien dann für mehrere Produkte in der [Konsole](#) von [AWS](#)

[Fakturierung und Kostenmanagement and Cost Management](#) verwenden. Dazu zählen AWS Cost Explorer, AWS Budgets, AWS Cost and Usage Report und AWS Cost Anomaly Detection.

Das folgende Diagramm zeigt Ihnen beispielsweise, wie Ihre Kosten- und Nutzungsinformationen in Ihrer Organisation gruppiert werden können, z. B. mit mehreren Teams (Kostenkategorie) und mehreren Umgebungen (Regeln), wobei jede Umgebung mehrere Ressourcen oder Assets (Dimensionen) aufweist.



Organigramm für Kosten und Nutzung

Sie können mithilfe von Kostenkategorien auch Kostengruppierungen erstellen. Nachdem Sie die Kostenkategorien erstellt haben (es kann nach dem Erstellen einer Kostenkategorie bis zu 24 Stunden dauern, bis die Werte in Ihren Nutzungsdatensätzen aktualisiert sind), erscheinen sie in [AWS Cost Explorer](#), [AWS Budgets](#), [AWS Cost and Usage Report](#) und [AWS Cost Anomaly Detection](#). In AWS Cost Explorer und AWS Budgets erscheint eine Kostenkategorie als zusätzliche Fakturierungsdimension. Damit können Sie Daten nach dem bestimmten Cost Category-Wert filtern oder nach Cost Category gruppieren.

Implementierungsschritte

- Definieren der Organisationskategorien: Treffen Sie sich mit internen Stakeholdern und Vertretern aus verschiedenen Unternehmensbereichen, um Kategorien zu definieren, die die Struktur und Anforderungen Ihrer Organisation widerspiegeln. Diese werden direkt der Struktur vorhandener

Finanzkategorien zugeordnet, z. B. Geschäftsbereich, Budget, Kostenstelle oder Abteilung. Sehen Sie sich die Ergebnisse an, die die Cloud für Ihr Unternehmen liefert, z. B. Schulungen oder Fortbildungen, da es sich auch um Organisationskategorien handelt.

- Definieren der funktionalen Kategorien: Treffen Sie sich mit internen Stakeholdern und Vertretern aus verschiedenen Unternehmensbereichen, um Kategorien zu definieren, die die Funktionen in Ihrer Organisation widerspiegeln. Dabei kann es sich um den Workload- oder Anwendungsnamen und die Art der Umgebung handeln, z. B. Produktion, Test oder Entwicklung.
- Definieren von AWS-Kostenkategorien: Erstellen Sie Kostenkategorien, um Ihre Kosten- und Nutzungsinformationen zu organisieren. Verwenden Sie hierfür [AWS Cost Categories](#) und ordnen Sie Ihre AWS-Kosten und -Nutzung [aussagekräftigen Kategorien](#) zu. Einer Ressource können mehrere Kategorien zugewiesen werden und eine Ressource kann sich in mehreren verschiedenen Kategorien befinden. Definieren Sie daher so viele Kategorien wie nötig, um Ihre Kosten innerhalb der kategorisierten Struktur mithilfe von AWS Cost Categories zu [verwalten](#).

Ressourcen

Zugehörige Dokumente:

- [Taggen von AWS-Ressourcen](#)
- [Verwenden von Kostenzuordnungs-Tags](#)
- [Analyse Ihrer Kosten mit AWS Budgets](#)
- [Analysieren der Kosten mit Cost Explorer](#)
- [Verwalten von AWS Cost and Usage Reporten](#)
- [AWS Kostenkategorien](#)
- [Verwaltung Ihrer Kosten mit AWS-Kostenkategorien](#)
- [Erstellen von Kostenkategorien](#)
- [Tagging von Kostenkategorien](#)
- [Aufteilen von Gebühren innerhalb von Kostenkategorien](#)
- [AWS-Features für Kostenkategorien](#)

Zugehörige Beispiele:

- [Organize your cost and usage data with AWS Cost Categories](#)
- [Verwalten Ihrer Kosten mit AWS Cost Categories](#)

COST03-BP04 Organisationskennzahlen festlegen

Definieren Sie die Organisationsmetriken, die für diese Workload erforderlich sind. Beispiele für Metriken einer Workload sind erstellte Kundenberichte oder Webseiten, die den Kunden angezeigt werden.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Entwickeln Sie ein Verständnis dafür, wie die Ausgabe Ihrer Workload im Vergleich zum Geschäftserfolg gemessen wird. Jede Workload verfügt in der Regel über einen kleinen Satz von Hauptausgaben, die auf die Leistung hinweisen. Wenn Sie eine komplexe Workload mit vielen Komponenten haben, können Sie die Liste priorisieren oder Metriken für jede Komponente definieren und nachverfolgen. Arbeiten Sie mit Ihren Teams zusammen, um zu verstehen, welche Metriken verwendet werden sollen. Diese Einheit wird verwendet, um die Effizienz der Workload oder die Kosten für die einzelnen Geschäftsausgaben zu verstehen.

Implementierungsschritte

- **Definieren von Workload-Ergebnissen:** Treffen Sie sich mit den Beteiligten im Unternehmen und definieren Sie die Ergebnisse für die Workload. Hierbei handelt es sich um eine primäre Maßnahme für die Kundennutzung. Es müssen Geschäftsmetriken und keine technischen Metriken gemessen werden. Es sollte eine kleine Anzahl von allgemeinen Metriken (weniger als fünf) pro Workload geben. Wenn die Workload mehrere Ergebnisse für verschiedene Anwendungsfälle erzeugt, gruppieren Sie sie in einer einzigen Metrik.
- **Definieren der Ergebnisse von Workload-Komponenten:** Wenn Sie eine große und komplexe Workload haben oder Ihre Workload problemlos in Komponenten (z. B. Microservices) mit gut definierten Ein- und Ausgaben aufteilen können, definieren Sie optional Metriken für jede Komponente. Der Aufwand sollte den Wert und die Kosten der Komponente widerspiegeln. Beginnen Sie mit den größten Komponenten und arbeiten Sie sich zu den kleineren Komponenten vor.

Ressourcen

Zugehörige Dokumente:

- [Ressourcen taggen AWS](#)
- [Analysieren Sie Ihre Kosten mit Budgets AWS](#)

- [Analysieren der Kosten mit Cost Explorer](#)
- [Verwalten von AWS -Kosten- und -Nutzungsberichten](#)

COST03-BP05 Konfigurieren von Tools für die Fakturierung und Kostenverwaltung

Konfigurieren Sie Kostenverwaltungstools in Übereinstimmung mit den Richtlinien Ihrer Organisation, um die Cloud-Ausgaben zu verwalten und zu optimieren. Dazu gehören Services, Tools und Ressourcen zur Organisation und Nachverfolgung von Kosten- und Nutzungsdaten, zur Verbesserung der Kontrolle durch konsolidierte Fakturierung und Zugriffsberechtigungen, zur Verbesserung der Planung durch Budgetierung und Prognosen, zum Erhalt von Benachrichtigungen oder Warnmeldungen und zur Kostensenkung durch Ressourcen- und Preisoptimierungen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Um eine starke Rechenschaftspflicht zu gewährleisten, erörtern Sie im Rahmen Ihrer Kostenzuordnungsstrategie zunächst Ihre Kontostrategie. Wenn Sie das richtig machen, reicht das möglicherweise schon aus. Andernfalls fehlen wichtige Informationen und es kann zu weiteren Problemen kommen.

Um die Rechenschaftspflicht für Cloud-Ausgaben zu fördern, gewähren Sie Benutzern Zugriff auf Tools, die einen Überblick über ihre Kosten und Nutzung bieten. AWS empfiehlt, dass Sie alle Workloads und Teams für die folgenden Zwecke konfigurieren:

- **Organisation:** Legen Sie Ihre Basis für die Kostenzuordnung und Governance mit Ihrer eigenen Tagging-Strategie und -Taxonomie fest. Erstellen Sie mehrere AWS-Konten mit Tools wie AWS Control Tower oder AWS Organization. Taggen Sie die unterstützten AWS-Ressourcen und kategorisieren Sie sie anhand Ihrer Organisationsstruktur (Geschäftseinheiten, Abteilungen oder Projekte) aussagekräftig. Taggen Sie Kontonamen für bestimmte Kostenstellen und ordnen Sie sie AWS Kostenkategorien zu, um Konten für Geschäftseinheiten für ihre Kostenstellen zu gruppieren, sodass der Eigentümer der Geschäftseinheit den Verbrauch mehrerer Konten an einem Ort sehen kann.
- **Zugriff:** Verfolgen Sie organisationsweite Fakturierungsinformationen durch konsolidierte Abrechnungen nach. Stellen Sie sicher, dass die richtigen Stakeholder und Geschäftsinhaber Zugriff darauf haben.
- **Kontrolle:** Entwickeln Sie effektive Governance-Mechanismen mit dem richtigen Integritätsschutz, um unerwartete Szenarien bei der Verwendung von Service-Kontrollrichtlinien (SCP), Tag-

Richtlinien, IAM-Richtlinien und Budgetwarnmeldungen zu verhindern. Beispielsweise können Sie Teams erlauben, bestimmte Ressourcen nur in bevorzugten Regionen zu erstellen, indem Sie effektive Kontrollmechanismen verwenden und verhindern, dass Ressourcen ohne bestimmte Tags (z. B. Kostenstelle) erstellt werden.

- **Aktueller Status:** Konfigurieren Sie ein Dashboard mit aktuellen Kosten- und Nutzungsraten. Das Dashboard sollte an einem gut sichtbaren Ort innerhalb der Arbeitsumgebung verfügbar sein (wie bei einem Betriebs-Dashboard). Sie können Daten exportieren und für eine gute Sichtbarkeit das Kosten- und Nutzungs-Dashboard aus dem AWS Cost Optimization Hub oder einem beliebigen unterstützten Produkt verwenden. Möglicherweise müssen Sie verschiedene Dashboards für verschiedene Personengruppen erstellen. Beispielsweise kann sich das Manager-Dashboard von einem Engineering-Dashboard unterscheiden.
- **Benachrichtigungen:** Stellen Sie mit AWS Budgets oder AWS Cost Anomaly Detection Benachrichtigungen bereit, wenn Kosten oder Nutzungen definierte Grenzwerte überschreiten und Anomalien auftreten.
- **Berichte:** Fassen Sie alle Kosten- und Nutzungsinformationen zusammen. Erhöhen Sie das Bewusstsein und die Verantwortlichkeit für Ihre Cloud-Ausgaben mit detaillierten, zuordnungsfähigen Kostendaten. Erstellen Sie Berichte, die für das Team, das sie bearbeitet, relevant sind und Empfehlungen enthalten.
- **Nachverfolgung:** Zeigen Sie die aktuellen Kosten und Nutzung in Bezug zu konfigurierten Zielen oder Vorgaben an.
- **Analyse:** Ermöglichen Sie Teammitgliedern die Durchführung benutzerdefinierter und detaillierter Analysen mit stündlicher, täglicher oder monatlicher Granularität und verschiedenen Filtern (Ressource, Konto, Tag usw.).
- **Prüfung:** Bleiben Sie hinsichtlich Ihrer Ressourcenbereitstellung und Ihrer Möglichkeiten zur Kostenoptimierung auf dem Laufenden. Erhalten Sie mit Amazon CloudWatch, Amazon SNS oder Amazon SES Benachrichtigungen für Ressourcenbereitstellungen auf Organisationsebene. Überprüfen Sie die Empfehlungen zur Kostenoptimierung mit AWS Trusted Advisor oder AWS Compute Optimizer.
- **Trendberichte:** Zeigen Sie die Variabilität von Kosten und Nutzung über den erforderlichen Zeitraum mit der erforderlichen Aufschlüsselung an.
- **Prognosen:** Zeigen Sie geschätzte zukünftige Kosten und schätzen Sie Ihre Ressourcennutzung und Ihre Ausgaben mit von Ihnen erstellten Prognose-Dashboards.

Sie können [AWS Cost Optimization Hub](#) verwenden, um potenzielle Kostenoptimierungsmöglichkeiten zu ermitteln, die an einem zentralen Standort konsolidiert werden,

und Datenexporte für die Integration mit Amazon Athena zu erstellen. Mit AWS Cost Optimization Hub können Sie außerdem das Kosten- und Nutzungs-Dashboard bereitstellen, das QuickSight für interaktive Kostenanalysen und den sicheren Austausch von Kosteninformationen verwendet.

Wenn Sie in Ihrer Organisation nicht über die notwendigen Kenntnisse oder die erforderliche Bandbreite verfügen, können Sie mit [AWS ProServ](#), [AWS Managed Services \(AMS\)](#) oder [AWS-Partnern](#) arbeiten. Sie können auch Tools von Drittanbietern verwenden. Stellen Sie jedoch sicher, dass Sie das Wertversprechen validieren.

Implementierungsschritte

- Teambasierten Zugriff auf Tools ermöglichen: Konfigurieren Sie Ihre Konten und erstellen Sie Gruppen, die Zugriff auf die erforderlichen Kosten- und Nutzungsberichte für ihre Verbräuche haben, und verwenden Sie [AWS Identity and Access Management](#), um den Zugriff auf die Tools wie AWS Cost Explorer zu [kontrollieren](#). Diese Gruppen müssen Vertreter aller Teams umfassen, die für eine Anwendung zuständig sind oder diese verwalten. Auf diese Weise wird sichergestellt, dass jedes Team Zugriff auf seine Kosten- und Nutzungsinformationen hat, um seinen Verbrauch nachzuverfolgen.
- Organisieren von Kosten-Tags und -Kategorien: Organisieren Sie Ihre Kosten nach Teams, Geschäftseinheiten, Anwendungen, Umgebungen und Projekten. Verwenden Sie Ressourcen-Tags, um Kosten nach Kostenzuordnungs-Tags zu organisieren. Erstellen Sie Kostenkategorien auf der Grundlage der Dimensionen und verwenden Sie dabei Tags, Konten, Services usw., um Ihre Kosten abzubilden.
- Konfigurieren von AWS-Budgets: [Konfigurieren Sie AWS-Budgets](#) für alle Konten für Ihre Workloads. Legen Sie mithilfe von Tags und Kostenkategorien Budgets für die Gesamtkontoausgaben und Budgets für die Workloads fest. Konfigurieren Sie Benachrichtigungen in AWS-Budgets, um Warnmeldungen zu erhalten, wenn Sie Ihre budgetierten Beträge überschreiten, oder wenn Ihre geschätzten Kosten Ihre Budgets übersteigen.
- Konfigurieren von AWS Cost Anomaly Detection: Verwenden Sie [AWS Cost Anomaly Detection](#) für Ihre Konten, Kernservices oder von Ihnen erstellte Kostenkategorien, um Ihre Kosten und Nutzung zu überwachen und ungewöhnliche Ausgaben zu erkennen. Sie können Warnmeldungen einzeln in aggregierten Berichten, in einer E-Mail oder einem Amazon-SNS-Thema erhalten. Dies ermöglicht es Ihnen, die Ursache der Anomalie zu analysieren und zu bestimmen und den Faktor zu identifizieren, der die Kostensteigerung verursacht.
- Verwenden von Kostenanalysetools: Konfigurieren Sie [AWS Cost Explorer](#) für Ihre Workload und Ihre Konten, um Ihre Kostendaten für die weitere Analyse zu visualisieren. Erstellen Sie ein Dashboard für die Workload, das die Gesamtausgaben, die wichtigsten Nutzungskennzahlen für

die Workload und die Prognose künftiger Kosten auf der Grundlage Ihrer historischen Kostendaten nachverfolgt.

- Verwenden Sie Analysetools zur Kosteneinsparung: Verwenden Sie AWS Cost Optimization Hub, um Einsparmöglichkeiten mit maßgeschneiderten Empfehlungen zu identifizieren, darunter das Löschen ungenutzter Ressourcen, die richtige Dimensionierung, Savings Plans, Reservierungen und Empfehlungen von Optimierern der Datenverarbeitung.
- Konfigurieren von erweiterten Tools: Sie können optional Grafiken erstellen, um die interaktive Analyse und den Austausch von Kosteninformationen zu erleichtern. Mit Datenexporten in AWS Cost Optimization Hub können Sie ein Kosten- und Nutzungs-Dashboard erstellen, das von QuickSight für Ihre Organisation bereitgestellt wird und zusätzliche Details und Granularität bietet. Sie können auch erweiterte Analysefunktionen implementieren, indem Sie Datenexporte in [Amazon Athena](#) für erweiterte Abfragen verwenden und Dashboards in [Quick](#) erstellen. Arbeiten Sie mit [AWS-Partnern](#) zusammen, um Cloud-Management-Lösungen für die konsolidierte Überwachung und Optimierung von Cloud-Rechnungen einzuführen.

Ressourcen

Zugehörige Dokumente:

- [What is AWS Fakturierung und Kostenmanagement and Cost Management?](#)
- [Einrichten einer AWS-Umgebung mit bewährten Methoden](#)
- [Best Practices for Tagging AWS Resources](#)
- [Markieren Ihrer AWS-Ressourcen](#)
- [AWS-Kostenkategorien](#)
- [Analysieren Ihrer Kosten mit AWS-Budgets](#)
- [Analyse Ihrer Kosten mit AWS Cost Explorer](#)
- [What is AWS Data Exports?](#)

Zugehörige Videos:

- [Deploying Cloud Intelligence Dashboards](#)
- [Get Alerts on any FinOps or Cost Optimization Metric or KPI](#)

Zugehörige Beispiele:

- [Kosten- und Nutzungs-Dashboard powered by Quick](#)
- [AWS Cost and Usage Governance Workshop](#)

COST03-BP06 Zuteilen von Kosten basierend auf Workload-Metriken

Teilen Sie die Kosten der betreffenden Workload anhand von Nutzungsmetriken oder Geschäftsergebnissen zu, um die Kosteneffizienz der Workload zu bewerten. Implementieren Sie einen Prozess zur Analyse der Kosten- und Nutzungsdaten mithilfe von Analytikservices, um von genaueren Einblicken und Rückbelastungsmöglichkeiten zu profitieren.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Niedrig

Implementierungsleitfaden

Die Kostenoptimierung ermöglicht Geschäftsergebnisse zum niedrigsten Preis, was nur durch Zuweisung von Workload-Kosten nach Workload-Metriken (gemessen nach Workload-Effizienz) erreicht werden kann. Überwachen Sie die definierten Workload-Metriken durch Protokolldateien oder andere Anwendungsüberwachung. Kombinieren Sie diese Daten mit den Workload-Kosten, die Sie erhalten können, indem Sie Kosten mit einem bestimmten Tag-Wert oder einer Konto-ID betrachten. Führen Sie diese Analyse stündlich durch. Ihre Effizienz ändert sich in der Regel, wenn Sie statische Kostenkomponenten haben (z. B. eine Backend-Datenbank, die dauerhaft ausgeführt wird) mit einer variierenden Anfragerate (z. B. Nutzungsspitzen von 9 bis 17 Uhr, mit wenigen Anfragen in der Nacht). Wenn Sie die Beziehung zwischen den statischen und variablen Kosten verstehen, können Sie Ihre Optimierungsaktivitäten besser fokussieren.

Das Erstellen von Workload-Metriken für gemeinsam genutzte Ressourcen kann im Vergleich zu Ressourcen wie containerisierten Anwendungen in Amazon Elastic Container Service (Amazon ECS) und Amazon API Gateway eine Herausforderung sein. Es gibt jedoch bestimmte Möglichkeiten, die Nutzung zu kategorisieren und die Kosten zu verfolgen. Wenn Sie gemeinsam genutzte Ressourcen von Amazon ECS und AWS Batch nachverfolgen müssen, können Sie die Zuordnung geteilter Kosten in AWS Cost Explorer aktivieren. Mithilfe von Daten zur Aufteilung der Kosten können Sie die Kosten und die Nutzung Ihrer containerisierten Anwendungen nachvollziehen und optimieren und die Anwendungskosten auf Grundlage des Verbrauchs der gemeinsam genutzten Rechen- und Speicherressourcen einzelnen Geschäftsbereichen zuteilen.

Implementierungsschritte

- Zuteilen von Kosten an Workload-Metriken: Erstellen Sie mit den definierten Metriken und konfigurierten Tags eine Metrik, die die Workload-Ausgabe und die Workload-Kosten kombiniert.

Verwenden Sie Analytik-Services wie Amazon Athena und Amazon Quick, um ein Effizienz-Dashboard für die gesamte Workload und alle Komponenten zu erstellen.

Ressourcen

Zugehörige Dokumente:

- [Taggen von AWS-Ressourcen](#)
- [Analysieren Ihrer Kosten mit AWS-Budgets](#)
- [Analysieren der Kosten mit Cost Explorer](#)
- [Verwalten von AWS-Kosten- und -Nutzungsberichten](#)

Zugehörige Beispiele:

- [Improve cost visibility of Amazon ECS and AWS Batch with AWS Split Cost Allocation Data](#)

KOSTEN 4. Wie können Sie Ressourcen außer Betrieb nehmen?

Implementieren Sie vom Beginn bis zum Abschluss eines Projekts eine Änderungskontrolle und Ressourcenverwaltung. So stellen Sie sicher, dass Sie ungenutzte Ressourcen abschalten oder beenden, um Verschwendung zu vermeiden.

Best Practices

- [COST04-BP01 Ressourcen im Laufe ihrer Lebensdauer verfolgen](#)
- [COST04-BP02 Implementieren eines Prozesses für die Außerbetriebnahme](#)
- [COST04-BP03 Außerbetriebnahme von Ressourcen](#)
- [COST04-BP04 Automatische Außerbetriebnahme von Ressourcen](#)
- [COST04-BP05 Durchsetzen von Richtlinien zur Datenaufbewahrung](#)

COST04-BP01 Ressourcen im Laufe ihrer Lebensdauer verfolgen

Definieren und implementieren Sie eine Methode zur Verfolgung von Ressourcen und deren Verknüpfungen mit Systemen über ihre gesamte Lebensdauer hinweg. Mit entsprechenden Tags können Sie die Workload oder die Funktion der Ressource identifizieren.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Nicht mehr benötigte Workload-Ressourcen werden außer Betrieb genommen. Ein gängiges Beispiel sind Ressourcen, die zum Testen verwendet werden. Nach Abschluss des Tests können die Ressourcen entfernt werden. Das Nachverfolgen von Ressourcen mit Tags (und Ausführen von Berichten zu diesen Tags) kann Ihnen helfen, Komponenten zu identifizieren, die außer Betrieb genommen werden können, weil sie nicht genutzt werden oder ihre Lizenz abläuft. Die Verwendung von Tags ist eine effektive Möglichkeit, Ressourcen zu verfolgen, indem die Ressource mit ihrer Funktion oder einem bekannten Datum, an dem sie außer Betrieb genommen werden kann, gekennzeichnet wird. Berichte können dann zu diesen Tags ausgeführt werden. Beispielwerte für das Taggen von Features sind `feature-X_testing`, um den Zweck der Ressource in Bezug auf den Workload-Lebenszyklus zu identifizieren. Ein anderes Beispiel ist die Verwendung von `LifeSpan` oder `TTL` für Ressourcen, wie z. B. den Namen und den Wert des `to-be-deleted` Tag-Schlüssels, um den Zeitraum oder eine bestimmte Zeit für die Außerbetriebnahme zu definieren.

Implementierungsschritte

- Implementieren eines Tagging-Schemas: Implementieren Sie ein Tagging-Schema, das die Workload identifiziert, zu der die Ressource gehört, und stellen Sie sicher, dass alle Ressourcen innerhalb der Workload entsprechend getaggt sind. Durch das Markieren mit Tags können Sie Ressourcen nach Zweck, Team, Umgebung oder anderen, für Ihr Unternehmen relevanten Kriterien kategorisieren. Detaillierte Informationen zu Anwendungsfällen, Strategien und Verfahren zum Taggen finden Sie in den [bewährten Methoden beim Tagging in AWS](#).
- Implementieren von Überwachung des Workload-Durchsatzes oder der Workload-Ausgabe: Implementieren Sie die Überwachung des Workload-Durchsatzes oder Alarme, die entweder bei Eingabeanforderungen oder Ausgabeabschlüssen ausgelöst werden. Konfigurieren Sie die Überwachung so, dass Benachrichtigungen erstellt werden, wenn Workload-Anforderungen oder -Ausgaben auf Null fallen. Dies bedeutet, dass die Workload-Ressourcen nicht mehr verwendet werden. Integrieren Sie einen Zeitfaktor, wenn die Workload unter normalen Bedingungen regelmäßig auf Null fällt. Weitere Informationen zu ungenutzten oder nicht ausgelasteten Ressourcen finden Sie im [Artikel zu Checks für die Kostenoptimierung mit AWS Trusted Advisor](#).
- AWS Ressourcen gruppieren: Erstellen Sie Gruppen für AWS Ressourcen. Sie können [AWS - Ressourcengruppen](#) verwenden, um Ihre AWS Ressourcen zu organisieren und zu verwalten, die sich in derselben Gruppe befinden AWS-Region. Den meisten Ressourcen lassen sich Tags hinzufügen, um sie innerhalb der Organisation zu identifizieren und zu sortieren. Mit [Tag Editor](#) können Sie mehreren unterstützten Ressourcen gleichzeitig Tags hinzufügen. Ziehen Sie die Verwendung von [AWS Service Catalog](#) in Erwägung, um Portfolios mit genehmigten Produkten

zu erstellen, zu verwalten und an Endnutzer zu verteilen und um den Produktlebenszyklus zu verwalten.

Ressourcen

Zugehörige Dokumente:

- [AWS Auto Scaling](#)
- [AWS Trusted Advisor](#)
- [AWS Trusted Advisor Prüfungen zur Kostenoptimierung](#)
- [Ressourcen kennzeichnen AWS](#)
- [Veröffentlichen von benutzerdefinierten Metriken](#)

Zugehörige Videos:

- [Wie optimiert man die Kosten mit AWS Trusted Advisor](#)

Zugehörige Beispiele:

- [AWS Ressourcen organisieren](#)
- [Optimieren Sie die Kosten mit AWS Trusted Advisor](#)

COST04-BP02 Implementieren eines Prozesses für die Außerbetriebnahme

Implementieren Sie einen Prozess für die Identifizierung und Außerbetriebnahme nicht genutzter Ressourcen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Implementieren Sie einen standardisierten Prozess in Ihrer gesamten Organisation, um ungenutzte Ressourcen zu identifizieren und zu entfernen. Der Prozess sollte definieren, wie häufig Suchvorgänge durchgeführt werden, und die Prozesse zum Entfernen der Ressource festlegen, um sicherzustellen, dass alle Organisationsanforderungen erfüllt sind.

Implementierungsschritte

- Erstellen und Implementieren eines Prozesses für die Außerbetriebnahme: Erstellen Sie in Zusammenarbeit mit den Workload-Entwicklern und -Besitzern einen Prozess zur Außerbetriebnahme der Workloads und ihrer Ressourcen. Der Prozess sollte die Methode abdecken, um zu überprüfen, ob die Workload verwendet wird, und auch, ob jede der Workload-Ressourcen verwendet wird. Definieren Sie die erforderlichen Schritte, um die Ressource außer Betrieb zu nehmen und gleichzeitig die Einhaltung gesetzlicher Anforderungen sicherzustellen. Alle zugeordneten Ressourcen sollten dabei eingeschlossen werden, z. B. Lizenzen oder zugehöriger Speicher. Informieren Sie die Workload-Besitzer darüber, dass die Außerbetriebnahme gestartet wurde.

Die folgenden Schritte für die Außerbetriebnahme geben vor, was im Rahmen des Prozesses geprüft werden sollte:

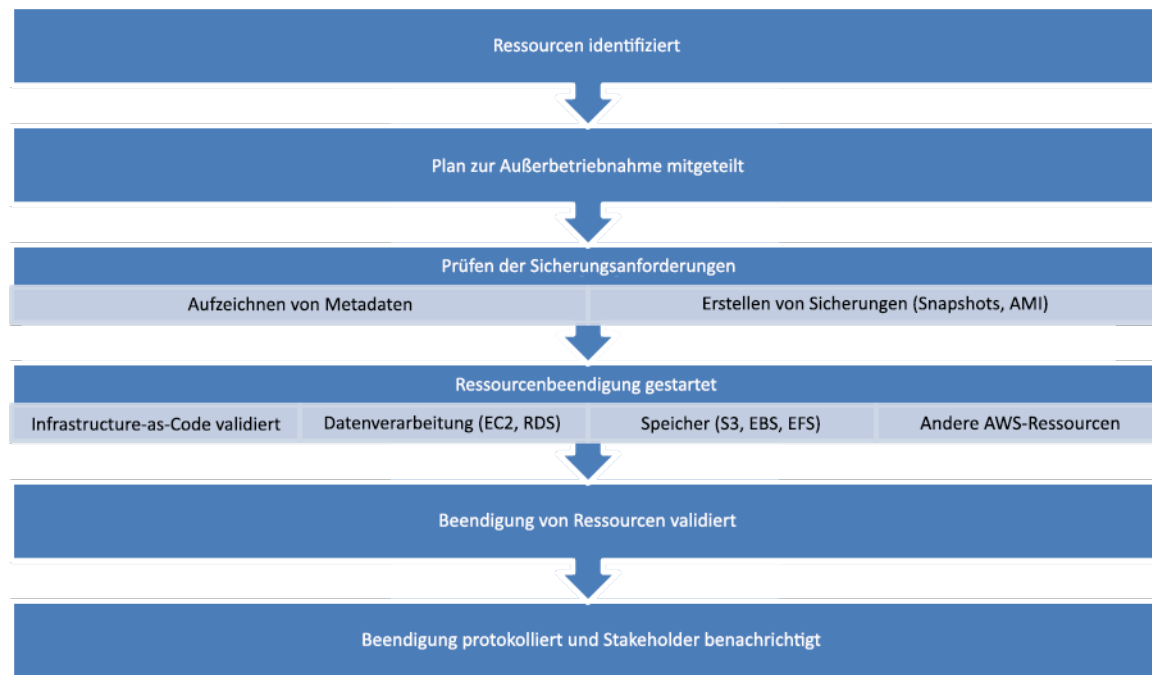
- Identifizieren der Ressourcen, die außer Betrieb genommen werden sollen: Identifizieren Sie die Ressourcen, die in Ihrer AWS Cloud für die Außerbetriebnahme in Frage kommen. Erfassen Sie alle erforderlichen Informationen und planen Sie die Außerbetriebnahme. Achten Sie bei der Zeitplanung darauf, unerwartete Probleme im Prozess zu berücksichtigen.
- Koordination und Kommunikation: Arbeiten Sie mit Workload-Besitzern zusammen, um zu bestätigen, dass die Ressource außer Betrieb genommen werden soll.
- Erfassen von Metadaten und Erstellen von Sicherungen: Erfassen Sie Metadaten (wie öffentliche IPs, Region, AZ, VPC, Subnetz und Sicherheitsgruppen) und erstellen Sie Sicherungen (z. B. Snapshots von Amazon Elastic Block Store oder AMI, Schlüssel- und Zertifikatexporte), wenn dies für die Ressourcen in der Produktionsumgebung erforderlich ist oder es sich um kritische Ressourcen handelt.
- Validieren von Infrastructure-as-Code: Bestimmen Sie, ob Ressourcen mit CloudFormation, Terraform, AWS Cloud Development Kit (AWS CDK) oder einem anderen Infrastructure-as-code-Bereitstellungstool bereitgestellt wurden, damit sie bei Bedarf erneut bereitgestellt werden können.
- Verhindern des Zugriffs: Wenden Sie restriktive Kontrollen für einen bestimmten Zeitraum an, um zu verhindern, dass Ressourcen genutzt werden, während Sie bestimmen, ob diese benötigt werden. Stellen Sie sicher, dass die Ressourcenumgebung bei Bedarf in den ursprünglichen Zustand zurückversetzt werden kann.
- Einhalten des internen Prozesses für die Außerbetriebnahme: Halten Sie sich an die Verwaltungsaufgaben und den Außerbetriebnahmeprozess Ihrer Organisation, z. B. Entfernen der Ressourcen aus der Organisationsdomain, Entfernen des DNS-Datensatzes und Entfernen der Ressourcen aus Ihrem Konfigurationsverwaltungstool, Überwachungstool, Automatisierungstools und Sicherheitstools.

Wenn es sich bei der Ressource um eine Amazon-EC2-Instance handelt, beachten Sie folgende Liste. [Weitere Details finden Sie unter „Wie kann ich meine Amazon-EC2-Ressourcen löschen oder beenden?“](#)

- Stoppen oder beenden Sie alle Ihre Amazon-EC2-Instances und Load Balancer. Amazon-EC2-Instances sind nach dem Beenden für kurze Zeit in der Konsole sichtbar. Instances, die sich nicht im Ausführungsstatus befinden, werden Ihnen nicht in Rechnung gestellt.
- Löschen Sie Ihre Auto-Scaling-Infrastruktur.
- Geben Sie alle Dedicated Hosts frei.
- Löschen Sie alle Amazon-EBS-Volumes und -Snapshots.
- Geben Sie alle elastischen IP-Adressen frei.
- Melden Sie alle Amazon Machine Images (AMIs) ab.
- Beenden Sie alle AWS Elastic Beanstalk-Umgebungen.

Wenn die Ressource ein Objekt in Amazon-Glacier-Speicher ist und Sie ein Archiv löschen, bevor die Mindestspeicherdauer erreicht wurde, wird eine anteilige Gebühr für das frühzeitige Löschen in Rechnung gestellt. Die Mindestspeicherdauer für Amazon Glacier ist abhängig von der verwendeten Speicherkategorie. Eine Übersicht über die Mindestspeicherdauer der einzelnen Speicherkategorien finden Sie in der Übersicht über die [Leistung für die verschiedenen Amazon-S3-Speicherkategorien](#). Informationen zu Gebühren für vor Ablauf der Mindestspeicherdauer gelöschte Objekte finden Sie in der [Amazon-S3-Preisübersicht](#).

Das folgende Flussdiagramm eines einfachen Außerbetriebnahmeprozesses zeigt die einzelnen Schritte. Bestätigen Sie vor der Außerbetriebnahme von Ressourcen, dass die Ressourcen, die Sie für die Außerbetriebnahme identifiziert haben, von der Organisation nicht genutzt werden.



Ablauf der Außerbetriebnahme von Ressourcen

Ressourcen

Zugehörige Dokumente:

- [AWS Auto Scaling](#)
- [AWS Trusted Advisor](#)
- [AWS CloudTrail](#)

Zugehörige Videos:

- [Delete CloudFormation stack but retain some resources](#)
- [Find out which user launched Amazon EC2 instance](#)

Zugehörige Beispiele:

- [Löschen oder beenden von Amazon-EC2-Ressourcen](#)
- [Identifizieren Sie den Benutzer, der eine Amazon-EC2-Instance in einem Konto gestartet hat](#)

COST04-BP03 Außerbetriebnahme von Ressourcen

Außerbetriebnahme von Ressourcen, die durch Ereignisse wie regelmäßige Prüfungen oder Änderungen der Nutzung ausgelöst werden. Die Außerbetriebnahme erfolgt normalerweise regelmäßig und kann manuell oder automatisiert durchgeführt werden.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Die Häufigkeit und der Aufwand für die Suche nach ungenutzten Ressourcen sollten die potenziellen Einsparungen widerspiegeln, sodass ein Konto mit geringen Kosten seltener analysiert werden sollte als ein Konto mit größeren Kosten. Suchanfragen und Außerbetriebnahmeereignisse können durch Statusänderungen in der Workload initiiert werden, z. B. ein Produkt, das sich dem Ende seiner Lebensdauer nähert oder ersetzt wird. Suchen und Außerbetriebnahme können auch durch externe Ereignisse initiiert werden, wie z. B. Änderungen der Marktbedingungen oder Produkteinstellung.

Implementierungsschritte

- **Außerbetriebnahme von Ressourcen:** Dies ist die Phase, in der AWS-Ressourcen, die nicht mehr benötigt werden oder deren Lizenzvereinbarung abläuft, als veraltet deaktiviert werden. Führen Sie alle abschließenden Prüfungen durch und erstellen Sie Snapshots und Sicherungen, bevor Sie zur Entsorgungsphase übergehen, um unerwünschte Unterbrechungen zu vermeiden. Befolgen Sie den Außerbetriebnahmeprozess, um jede der Ressourcen, die als nicht genutzt identifiziert wurde, außer Betrieb zu nehmen.

Ressourcen

Zugehörige Dokumente:

- [AWS Auto Scaling](#)
- [AWS Trusted Advisor](#)

COST04-BP04 Automatische Außerbetriebnahme von Ressourcen

Gestalten Sie Ihre Workload so, dass sie die Beendigung von Ressourcen reibungslos handhabt, wenn Sie unkritische Ressourcen, nicht benötigte Ressourcen oder Ressourcen mit geringer Auslastung identifizieren und außer Betrieb nehmen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Niedrig

Implementierungsleitfaden

Verwenden Sie die Automatisierung, um die damit verbundenen Kosten für die Außerbetriebnahme zu reduzieren oder zu entfernen. Wenn Sie Ihre Workload so konzipieren, dass die eine automatische Außerbetriebnahme durchführt, werden die gesamten Workload-Kosten während der Nutzungsdauer gesenkt. Sie können den Außerbetriebnahmeprozess mit [Amazon EC2 Auto Scaling oder Application Auto Scaling](#) durchführen. Sie können auch benutzerdefinierten Code mithilfe der [API oder des SDK](#) implementieren, um Workload-Ressourcen automatisch außer Betrieb zu nehmen.

[Moderne Anwendungen](#) basieren auf Serverless First, einer Strategie, die der Einführung von Serverless-Services Priorität einräumt. AWS hat [Serverless-Services](#) für alle drei Ebenen Ihres Stacks entwickelt: Datenverarbeitung, Integration und Datenspeicher. Mit einer Serverless-Architektur können Sie in Phasen mit wenig Datenverkehr dank automatischer Skalierung Kosten sparen.

Implementierungsschritte

- Implementieren von Amazon EC2 Auto Scaling oder Application Auto Scaling: Konfigurieren Sie unterstützte Ressourcen mit Amazon EC2 Auto Scaling oder Application Auto Scaling. Mit diesen Services können Sie die Nutzung und Kosteneffizienz bei der Verwendung von AWS-Services optimieren. Wenn die Nachfrage sinkt, entfernen diese Services automatisch überschüssige Ressourcenkapazitäten, damit keine unnötigen Kosten entstehen.
- Konfigurieren von CloudWatch zum Beenden von Instances: Das Beenden von Instances kann mithilfe von [CloudWatch-Alarmen](#) konfiguriert werden. Implementieren Sie mithilfe der Metriken aus dem Außerbetriebnahmeprozess einen Alarm mit einer Aktion von Amazon Elastic Compute Cloud. Überprüfen Sie den Vorgang vor der Einführung in einer Nicht-Produktionsumgebung.
- Implementieren von Code innerhalb der Workload: Sie können Workload-Ressourcen mit dem AWS-SDK oder der AWS CLI außer Betrieb nehmen. Implementieren Sie Code innerhalb der in AWS integrierten Anwendung, die nicht mehr verwendete Ressourcen beendet oder entfernt.
- Verwenden von Serverless-Services: Priorisieren Sie das Erstellen von [Serverless-Architekturen](#) und [ereignisgesteuerten Architekturen](#) in AWS, um Ihre Anwendungen zu erstellen und auszuführen. AWS bietet Services mit verschiedenen Serverless-Technologien an, die von sich aus eine automatisch optimierte Ressourcennutzung und automatisierte Außerbetriebnahme bereitstellen (Abskalieren und Aufskalieren). Bei Serverless-Anwendungen wird die Ressourcennutzung automatisch optimiert und Ihnen entstehen nie Kosten für die Überbereitstellung.

Ressourcen

Zugehörige Dokumente:

- [Amazon EC2 Auto Scaling](#)
- [Erste Schritte mit Amazon EC2 Auto Scaling](#)
- [Application Auto Scaling](#)
- [AWS Trusted Advisor](#)
- [Serverless in AWS](#)
- [Erstellen Sie Alarme, um eine Instance anzuhalten, zu beenden, neu zu starten oder wiederherzustellen.](#)
- [Hinzufügen von Aktionen zum Beenden zu Amazon-CloudWatch-Alarmen](#)

Zugehörige Beispiele:

- [Scheduling automatic deletion of AWS CloudFormation stacks](#)

COST04-BP05 Durchsetzen von Richtlinien zur Datenaufbewahrung

Definieren Sie Richtlinien zur Datenaufbewahrung auf unterstützten Ressourcen, um das Löschen von Objekten gemäß den Anforderungen Ihrer Organisation durchzuführen. Identifizieren und löschen Sie entbehrliche und verwaiste Ressourcen und Objekte, die nicht mehr benötigt werden.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Mit Richtlinien zur Datenaufbewahrung und Lebenszyklusrichtlinien können Sie die mit der Außerbetriebnahme von Prozessen verbundenen Kosten sowie die Speicherkosten für die identifizierten Ressourcen reduzieren. Die Definition von Richtlinien zur Datenaufbewahrung und Lebenszyklusrichtlinien zur Durchführung einer automatischen Speicherklassenmigration und Löschung verringert die Gesamtspeicherkosten während des Lebenszyklus. Sie können Amazon Data Lifecycle Manager verwenden, um die Erstellung und Löschung von Snapshots von Amazon Elastic Block Store und Amazon EBS-gestützten Amazon Machine Images (AMIs) zu automatisieren, und Sie können Amazon S3 Intelligent-Tiering oder eine Amazon-S3-Lebenszykluskonfiguration verwenden, um den Lebenszyklus Ihrer Amazon-S3-Objekte zu verwalten. Mithilfe der [API oder des SDK](#) können Sie auch benutzerdefinierten Code implementieren, um Lebenszyklusrichtlinien und Richtlinienregeln für die automatische Löschung von Objekten zu erstellen.

Implementierungsschritte

- Verwenden von Amazon Data Lifecycle Manager: Verwenden Sie Lebenszyklusrichtlinien in Amazon Data Lifecycle Manager, um die Löschung von Amazon-EBS-Snapshots und Amazon-EBS-gestützten AMIs zu automatisieren.
- Einrichten der Lebenszykluskonfiguration für einen Bucket: Verwenden Sie die Amazon-S3-Lebenszykluskonfiguration für einen Bucket, um Aktionen für Amazon S3 zu definieren, die während des Lebenszyklus des Objekts ergriffen werden sollen, sowie die Löschung am Ende des Lebenszyklus des Objekts basierend auf Ihren geschäftlichen Anforderungen.

Ressourcen

Zugehörige Dokumente:

- [AWS Trusted Advisor](#)
- [Amazon Data Lifecycle Manager](#)
- [How to set lifecycle configuration on Amazon S3 bucket](#)

Zugehörige Videos:

- [Automate Amazon EBS Snapshots with Amazon Data Lifecycle Manager](#)
- [Empty an Amazon S3 bucket using a lifecycle configuration rule](#)

Zugehörige Beispiele:

- [Empty an Amazon S3 bucket using a lifecycle configuration rule](#)

Kostengünstige Ressourcen

Fragen

- [KOSTEN 5. Wie können Sie die Kosten bei der Auswahl von Services einschätzen?](#)
- [KOSTEN 6. Wie können Sie bei der Auswahl des Ressourcentyps, -umfangs und der Anzahl der Ressourcen Kostenziele erfüllen?](#)
- [KOSTEN 7. Wie können Sie Kosten mithilfe von Preismodellen senken?](#)
- [KOSTEN 8. Wie können Sie die Kosten für Datenübertragungen planen?](#)

KOSTEN 5. Wie können Sie die Kosten bei der Auswahl von Services einschätzen?

Bei Amazon EC2, Amazon EBS und Amazon S3 handelt es sich um AWS-Services, die als einzelne Bausteine angeboten werden. Verwaltete Services, etwa Amazon RDS und Amazon DynamoDB, sind AWS-Services auf einer höheren Ebene oder Anwendungsebene. Wenn Sie sich für die richtigen Bausteine und verwalteten Services entscheiden, können Sie die Kosten dieser Workload optimieren. Durch die Nutzung von verwalteten Services können Sie einen Großteil Ihres administrativen und betrieblichen Overheads reduzieren oder beseitigen und damit Kapazitäten für anwendungs- und geschäftsbezogene Aktivitäten gewinnen.

Best Practices

- [COST05-BP01 Identifizieren Sie die organisatorischen Anforderungen an die Kosten](#)
- [COST05-BP02 Analysieren sämtlicher Komponenten dieser Workload](#)
- [COST05-BP03 Durchführen einer gründlichen Analyse der einzelnen Komponenten](#)
- [COST05-BP04 Auswahl von Software mit kosteneffizienter Lizenzierung](#)
- [COST05-BP05 Auswahl von Komponenten dieser Workload zur Optimierung der Kosten im Einklang mit den Prioritäten der Organisation](#)
- [COST05-BP06 Durchführen einer Kostenanalyse für unterschiedliche Nutzungen im Lauf der Zeit](#)

COST05-BP01 Identifizieren Sie die organisatorischen Anforderungen an die Kosten

Definieren Sie gemeinsam mit den Teammitgliedern für diese Workload das Gleichgewicht zwischen Kostenoptimierung und anderen Säulen wie Leistung und Zuverlässigkeit.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

In den meisten Organisationen besteht die Abteilung für Informationstechnologie (IT) aus mehreren kleinen Teams, von denen jedes seine eigene Agenda und seinen eigenen Schwerpunktbereich hat, der die Spezialgebiete und Fähigkeiten seiner Teammitglieder widerspiegelt. Sie müssen die allgemeinen Ziele, Prioritäten und Vorgaben Ihrer Organisation verstehen und wissen, wie jede Abteilung oder jedes Projekt zu diesen Zielen beiträgt. Die Kategorisierung aller wesentlichen Ressourcen, einschließlich Personal, Ausrüstung, Technologie, Material und externer Dienstleistungen, ist für die Erreichung der organisatorischen Ziele und eine umfassende Budgetplanung von entscheidender Bedeutung. Die Anwendung dieses systematischen Ansatzes

zur Kostenermittlung und zum Kostenverständnis ist für die Erstellung eines realistischen und soliden Kostenplans für die Organisation von grundlegender Bedeutung.

Bei der Auswahl von Services für Ihre Workload ist es wichtig, dass Sie die Prioritäten Ihrer Organisation verstehen. Schaffen Sie ein Gleichgewicht zwischen Kostenoptimierung und anderen AWS Well-Architected Framework-Säulen wie Leistung und Zuverlässigkeit. Dieser Prozess sollte systematisch und regelmäßig durchgeführt werden, um Veränderungen in den Zielen der Organisation, den Marktbedingungen und der betrieblichen Dynamik zu berücksichtigen. Eine vollständig kostenoptimierte Workload ist die Lösung, die am meisten an den Anforderungen Ihrer Organisation ausgerichtet ist, nicht notwendigerweise an den niedrigsten Kosten. Treffen Sie sich mit allen Teams innerhalb Ihrer Organisation, um Informationen zu sammeln, z. B. mit den Produkt-, Geschäfts-, Technik- und Finanz-Teams. Bewerten Sie die Auswirkungen von Kompromissen zwischen konkurrierenden Interessen oder alternativen Ansätzen, um fundiert zu entscheiden, auf welche Bereiche die Anstrengungen konzentriert werden sollten, oder eine geeignete Handlungsweise zu wählen.

Beispielsweise kann die Beschleunigung der Markteinführung neuer Features einer Kostenoptimierung vorgezogen werden oder Sie können eine relationale Datenbank für nicht relationale Daten wählen, um die Migration eines Systems zu vereinfachen, anstatt zu einer für Ihren Datentyp optimierten Datenbank zu migrieren und Ihre Anwendung zu aktualisieren.

Implementierungsschritte

- Ermitteln der Organisationsanforderungen zur Kosteneinschätzung: Treffen Sie sich mit Teammitgliedern aus Ihrer Organisation, darunter Produktmanagement, Anwendungsbesitzern, Entwicklungs- und Betriebsteams, Management und Finanzen. Setzen Sie die Prioritäten hinsichtlich der Well-Architected-Säulen für diese Workload und seine Komponenten. Die Ausgabe sollte eine Liste der Säulen in der entsprechenden Reihenfolge sein. Sie können jeder Säule auch eine Gewichtung zuweisen, um anzugeben, wie viel zusätzlicher Fokus sie hat, oder wie ähnlich der Fokus zwischen zwei Säulen ist.
- Erfassen und Dokumentieren der technischen Schulden: Befassen Sie sich bei der Workload-Überprüfung mit den technischen Schulden. Dokumentieren Sie ein Backlog-Element, um die Workload in Zukunft wieder aufzugreifen und erneut zu überarbeiten oder neu zu strukturieren, mit dem Ziel, sie weiter zu optimieren. Es ist wichtig, dass Sie die Kompromisse, die Sie eingegangen sind, den anderen Beteiligten klar mitteilen.

Ressourcen

Zugehörige bewährte Methoden:

- [REL11-BP07 Gestalten Sie Ihr Produkt so, dass es Verfügbarkeitsziele und Service Level Agreements für Verfügbarkeit erfüllt \(\) SLAs](#)
- [OPS01-BP06 Bewerten Sie Kompromisse](#)

Zugehörige Dokumente:

- [AWS Rechner für die Gesamtbetriebskosten \(\) TCO](#)
- [Amazon-S3-Speicherklassen](#)
- [Cloud-Produkte](#)

COST05-BP02 Analysieren sämtlicher Komponenten dieser Workload

Stellen Sie sicher, dass jede Workload-Komponente unabhängig von der derzeitigen Größe oder den aktuellen Kosten analysiert wird. Der Überprüfungsaufwand sollte in einem angemessenen Verhältnis zu dem potenziellen Nutzen stehen, z. B. bei einer Prüfung der derzeitigen und prognostizierten Kosten.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Workload-Komponenten, die der Organisation einen geschäftlichen Nutzen bringen sollen, können verschiedene Services umfassen. Für jede Komponente können Sie bestimmte AWS Cloud-Services auswählen, um den Geschäftsanforderungen gerecht zu werden. Diese Auswahl könnte von Faktoren wie der Vertrautheit mit diesen Services oder früheren Erfahrungen mit ihnen beeinflusst sein.

Nachdem Sie die Anforderungen Ihrer Organisation ermittelt haben (wie in [COST05-BP01 Ermitteln der Organisationsanforderungen zur Kosteneinschätzung](#) erwähnt), führen Sie eine gründliche Analyse aller Komponenten Ihrer Workload durch. Analysieren Sie jede Komponente unter Berücksichtigung der aktuellen und prognostizierten Kosten und Größen. Wägen Sie die Kosten der Analyse gegen die potenziellen Einsparungen bei der Workload während des Lebenszyklus ab. Der Aufwand, der für die Analyse aller Komponenten dieser Workloads betrieben wird, sollte den potenziellen Einsparungen oder Verbesserungen entsprechen, die durch die Optimierung dieser spezifischen Komponente zu erwarten sind. Wenn zum Beispiel die Kosten der vorgeschlagenen Ressource 10 USD/Monat betragen und bei prognostizierter Belastung 15 USD/Monat nicht

überschreiten würden, könnte ein Tag Aufwand, um die Kosten um 50 % zu reduzieren (5 USD pro Monat), den potenziellen Nutzen über die Lebensdauer des Systems übersteigen. Verwenden Sie eine schnellere und effizientere datenbasierte Schätzung, um das beste Gesamtergebnis für diese Komponente zu erzielen.

Workloads können sich im Laufe der Zeit ändern. Die richtigen Services sind möglicherweise nicht optimal, wenn sich die Workload-Architektur oder -Nutzung ändert. Die Analyse für die Auswahl von Services muss aktuelle und zukünftige Workload-Zustände und Nutzungsebenen umfassen. Die Implementierung eines Service für den zukünftigen Workload-Status oder die Nutzung kann die Gesamtkosten senken, indem der Aufwand reduziert oder beseitigt wird, der für zukünftige Änderungen erforderlich ist. Zum Beispiel könnte die Verwendung von EMR Serverless zunächst die richtige Wahl sein. Wenn jedoch die Nutzung dieses Services zunimmt, könnte die Umstellung auf EMR in EC2 die Kosten für diese Komponente der Workload senken.

[AWS Cost Explorer](#) und die AWS Cost and Usage Reports ([CUR](#)) können die Kosten eines Machbarkeitsnachweises (Proof of Concept, PoC) oder einer laufenden Umgebung analysieren. Sie können [AWS Pricing Calculator](#) auch verwenden, um die Workload-Kosten zu schätzen.

Schreiben Sie einen Workflow, an den sich die technischen Teams halten, um ihre Workloads zu überprüfen. Halten Sie diesen Workflow einfach, decken Sie aber auch alle notwendigen Schritte ab, um sicherzustellen, dass die Teams jede Komponente der Workload und ihre Preisgestaltung verstehen. Ihre Organisation kann diesen Workflow dann verfolgen und an die spezifischen Bedürfnisse jedes Teams anpassen.

1. Listen Sie jeden Dienst auf, der für Ihre Workload verwendet wird: Dies ist ein guter Ausgangspunkt. Identifizieren Sie alle Services, die derzeit genutzt werden und woher die Kosten stammen.
2. Verstehen Sie, wie die Preisgestaltung für diese Services funktioniert: Machen Sie sich mit dem [Preismodell](#) der einzelnen Services vertraut. Verschiedene AWS-Services haben unterschiedliche Preismodelle, die auf Faktoren wie Nutzungsvolumen, Datenübertragung und Feature-spezifischen Preisen basieren.
3. Konzentrieren Sie sich auf die Services, für die unerwartete Workload-Kosten anfallen und die nicht mit der erwarteten Nutzung und dem erwarteten Geschäftsergebnis übereinstimmen: Identifizieren Sie mithilfe von AWS Cost Explorer oder AWS Cost and Usage Reports Ausreißer oder Services, bei denen die Kosten nicht proportional zum Wert oder zur Nutzung sind. Es ist wichtig, die Kosten mit den Geschäftsergebnissen zu korrelieren, um Optimierungsmaßnahmen zu priorisieren.

4. Nutzen Sie AWS Cost Explorer, CloudWatch Logs, VPC Flow Logs und Amazon S3 Storage Lens, um die Ursache dieser hohen Kosten zu verstehen: Diese Tools sind für die Diagnose hoher Kosten von entscheidender Bedeutung. Jeder Service bietet einen anderen Blickwinkel, um die Nutzung und Kosten zu betrachten und zu analysieren. Cost Explorer hilft beispielsweise bei der Bestimmung der Gesamtkostentrends, CloudWatch Logs liefert betriebliche Erkenntnisse, VPC Flow Logs zeigt den IP-Verkehr an und Amazon S3 Storage Lens ist nützlich für Speicheranalytik.
5. Verwenden Sie AWS Budgets, um Budgets für bestimmte Beträge für Services oder Konten festzulegen: Die Festlegung von Budgets ist eine proaktive Methode zur Kostenverwaltung. Nutzen Sie AWS Budgets, um benutzerdefinierte Budgetschwellenwerte festzulegen und Warnmeldungen zu erhalten, wenn die Kosten diese Schwellenwerte überschreiten.
6. Konfigurieren Sie Amazon-CloudWatch-Alarme zum Senden von Abrechnungs- und Nutzungsmetriken: Richten Sie Überwachungs- und Warnmeldungen für Kosten- und Nutzungsmetriken ein. CloudWatch-Alarme können Sie benachrichtigen, wenn bestimmte Schwellenwerte überschritten werden, was die Reaktionszeit verbessert.

Erzielen Sie im Laufe der Zeit bemerkenswerte Verbesserungen und finanzielle Einsparungen durch eine strategische Überprüfung aller Workload-Komponenten, unabhängig von ihren gegenwärtigen Merkmalen. Der Aufwand für diesen Überprüfungsprozess sollte bewusst und unter sorgfältiger Abwägung der möglichen Vorteile betrieben werden.

Implementierungsschritte

- Erstellen einer Liste der Workload-Komponenten: Erstellen Sie eine Liste mit den Komponenten Ihrer Workload. Verwenden Sie diese Liste, um zu überprüfen, ob jede Komponente analysiert wurde. Der Aufwand sollte die Kritikalität für die Workload widerspiegeln, die durch die Prioritäten Ihrer Organisation definiert wird. Die Gruppierung von Ressourcen verbessert die Effizienz (z. B. die Speicherung von Produktionsdatenbanken, wenn es mehrere Datenbanken gibt).
- Priorisieren der Komponentenliste: Priorisieren Sie die Komponentenliste entsprechend dem Aufwand. In der Regel erfolgt die Priorisierung nach den Kosten der Komponente – von der teuersten zur günstigsten. Alternativ kann sie auch nach der von den Prioritäten Ihrer Organisation definierten Kritikalität erfolgen.
- Durchführen der Analyse: Überprüfen Sie für jede Komponente auf der Liste die verfügbaren Optionen und Services und wählen Sie die Option aus, die am besten mit Ihren Organisationsprioritäten übereinstimmt.

Ressourcen

Zugehörige Dokumente:

- [AWS Pricing Calculator](#)
- [AWS Cost Explorer](#)
- [Amazon-S3-Speicherklassen](#)
- [AWS Cloud-Produkte](#)

Zugehörige Videos:

- [AWS Cost Optimization Series: CloudWatch](#)

COST05-BP03 Durchführen einer gründlichen Analyse der einzelnen Komponenten

Nehmen Sie die Gesamtkosten, die der Organisation durch die einzelnen Komponenten entstehen, unter die Lupe. Berechnen Sie die Gesamtbetriebskosten unter Berücksichtigung der Betriebs- und Verwaltungskosten, insbesondere bei der Nutzung von verwalteten Services durch den Cloud-Anbieter. Der Überprüfungsaufwand sollte in einem angemessenen Verhältnis zum potenziellen Nutzen stehen, z. B. muss die Zeit, die für die Analyse benötigt wird, den Komponentenkosten entsprechen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Bedenken Sie die Zeitersparnis, die es Ihrem Team ermöglicht, sich auf das Aufholen technischen Rückstands, Innovation, wertschöpfende Features und die Herausarbeitung eines Alleinstellungsmerkmals zu konzentrieren. So könnten Sie beispielsweise Ihre Datenbank von Ihrer On-Premises-Umgebung so schnell wie möglich per Lift and Shift in die Cloud verlagern (auch als Hostwechsel bekannt) und die Optimierung im Nachgang ausführen. Es lohnt sich, die möglichen Einsparungen zu untersuchen, die Sie durch den Einsatz von verwalteten Services auf AWS erzielen könnten, die Lizenzkosten entfernen oder reduzieren. Verwaltete Services auf AWS eliminieren den betrieblichen und administrativen Aufwand für die Wartung eines Service, wie das Patching oder die Aktualisierung des Betriebssystems, sodass Sie sich auf Innovationen und das Geschäft konzentrieren können.

Da verwaltete Services in der großen Cloud-Umgebung ausgeführt werden, profitieren Sie hier von geringeren Kosten pro Transaktion oder Service. Sie können potenzielle Optimierungen vornehmen,

um konkrete Vorteile zu erzielen, ohne die Kernarchitektur der Anwendung zu ändern. Beispielsweise ist es möglich, den Zeitaufwand, den Sie für die Verwaltung von Datenbank-Instances aufbringen, zu verringern, indem Sie zu einer Database-as-a-Service-Plattform wie [Amazon Relational Database Service \(Amazon RDS\)](#) migrieren oder Ihre Anwendung zu einer vollständig verwalteten Plattform wie [AWS Elastic Beanstalk](#) migrieren.

Verwaltete Services weisen in der Regel Attribute auf, die Sie festlegen können, um zu gewährleisten, dass ausreichend Kapazität bereitsteht. Sie müssen diese Attribute festlegen und überwachen, damit Ihre überschüssige Kapazität auf ein Minimum begrenzt und die Leistung maximiert werden. Sie können die Attribute der AWS Managed Services mithilfe der AWS-Managementkonsole oder AWS-APIs und SDKs ändern, um den Ressourcenbedarf an den sich ändernden Bedarf anzupassen. So können Sie beispielsweise die Anzahl der Knoten in einem Amazon-EMR-Cluster (oder einem Amazon-Redshift-Cluster) auf- oder abskalieren.

Außerdem können Sie mehrere Instances in eine AWS-Ressource packen, um eine Nutzung mit höherer Dichte zu aktivieren. So können Sie beispielsweise mehrere kleine Datenbanken auf einer einzelnen Datenbank-Instance von Amazon Relational Database Service (Amazon RDS) bereitstellen. Mit zunehmendem Wachstum können Sie eine der Datenbanken über einen Snapshot- und Wiederherstellungsprozess zu einer dedizierten Datenbank-Instance von Amazon RDS migrieren.

Wenn Sie Workloads auf verwalteten Services bereitstellen, müssen Sie sich mit den Anforderungen für das Anpassen der Service-Kapazität vertraut machen. Diese Anforderungen sind in der Regel Zeit, Aufwand und die Auswirkungen auf den normalen Workload-Betrieb. Die bereitgestellte Ressource muss Zeit für Änderungen einräumen und den erforderlichen Overhead bereitstellen, damit dies möglich ist. Der laufende Aufwand für das Ändern von Services kann praktisch auf null reduziert werden, wenn Sie APIs und SDKs verwenden, die mit den System- und Überwachungs-Tools wie Amazon CloudWatch integriert sind.

[Amazon RDS](#), [Amazon Redshift](#) und [Amazon ElastiCache](#) bieten einen verwalteten Datenbankservice. [Amazon Athena](#), [Amazon EMR](#) und [Amazon OpenSearch Service](#) bieten einen verwalteten Analytikservice.

[AMS](#) ist ein Service, der die AWS-Infrastruktur für Unternehmenskunden und -partner betreibt. Er bietet eine sichere und konforme Umgebung, in der Sie Ihre Workloads bereitstellen können. AMS verwendet Enterprise-Cloud-Betriebsmodelle mit Automatisierung, damit Sie Ihre Organisationsanforderungen erfüllen, schneller in die Cloud wechseln und Ihre laufenden Verwaltungskosten senken können.

Implementierungsschritte

- Durchführen einer gründlichen Analyse: Arbeiten Sie anhand der Komponentenliste jede Komponente von der höchsten Priorität bis zur niedrigsten Priorität ab. Führen Sie für die Komponenten mit höherer Priorität sowie für die teureren Komponenten zusätzliche Analysen durch und bewerten Sie alle verfügbaren Optionen und deren langfristige Auswirkungen. Bewerten Sie bei Komponenten mit niedrigerer Priorität, ob Änderungen in der Nutzung die Priorität der Komponente ändern. Führen Sie anschließend eine Analyse des angemessenen Aufwands durch.
- Vergleichen von verwalteten und nicht verwalteten Ressourcen: Berücksichtigen Sie die Betriebskosten für die von Ihnen verwalteten Ressourcen und vergleichen Sie sie mit von AWS verwalteten Ressourcen. Prüfen Sie beispielsweise Ihre Datenbanken, die auf Amazon-EC2-Instances ausgeführt werden, und vergleichen Sie sie mit Optionen von Amazon RDS (ein von AWS verwalteter Service) oder Amazon EMR verglichen mit der Ausführung von Apache Spark in Amazon EC2. Recherchieren Sie sorgfältig, welche Optionen Sie beim Wechsel von einer selbstverwalteten Workload zu einer vollständig verwalteten AWS-Workload haben. Berücksichtigen Sie dabei die drei wichtigsten Faktoren: die [Art des verwalteten Service](#), den Sie verwenden möchten, den Prozess, den Sie zur [Migration Ihrer Daten](#) verwenden, und ein Verständnis des [AWS-Modells der geteilten Verantwortung](#).

Ressourcen

Zugehörige Dokumente:

- [AWS Rechner für Gesamtbetriebskosten \(TCO\)](#)
- [Amazon-S3-Speicherklassen](#)
- [AWS Cloud-Produkte](#)
- [AWS-Modell der geteilten Verantwortung](#)

Zugehörige Videos:

- [Why move to a managed database?](#)
- [What is Amazon EMR and how can I use it for processing data?](#)

Zugehörige Beispiele:

- [Why move to a managed database](#)

- [Consolidate data from identical SQL Server databases into a single Amazon RDS for SQL Server database using AWS DMS](#)
- [Deliver data at scale to Amazon Managed Streaming for Apache Kafka \(Amazon MSK\)](#)
- [Migrate an ASP.NET web application to AWS Elastic Beanstalk](#)

COST05-BP04 Auswahl von Software mit kosteneffizienter Lizenzierung

Open-Source-Software eliminiert Softwarelizenzkosten, die erhebliche Kosten in Workloads verursachen können. Wenn lizenzierte Software erforderlich ist, vermeiden Sie Lizenzen, die an beliebige Attribute wie CPUs gebunden sind, und suchen Sie nach Lizenzen, die an die Ausgabe oder Ergebnisse gebunden sind. Die Kosten dieser Lizenzen lassen sich besser auf die von ihnen bereitgestellten Vorteile skalieren.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Niedrig

Implementierungsleitfaden

Der Begriff „Open Source“ hat seinen Ursprung in der Softwareentwicklung und bedeutet, dass die Software bestimmte Kriterien für die freie Verteilung erfüllt. Open-Source-Software zeichnet sich durch einen Quellcode aus, der von jedem eingesehen, verändert und verbessert werden kann. Auf Grundlage der geschäftlichen Anforderungen, der Fähigkeiten der Techniker, der prognostizierten Nutzung oder anderer technologischer Abhängigkeiten können Organisationen die Verwendung von Open-Source-Software in AWS in Betracht ziehen, um ihre Lizenzkosten zu minimieren. Mit anderen Worten, die Kosten für Softwarelizenzen können durch den Einsatz von [Open-Source-Software](#) gesenkt werden. Dies kann erhebliche Auswirkungen auf die Workload-Kosten haben, da die Größe der Workload skaliert wird.

Wägen Sie die Vorteile lizenzierter Software gegen die Gesamtkosten ab, um Ihre Workload zu optimieren. Modellieren Sie Änderungen bei der Lizenzierung und wie sich diese auf Ihre Workload-Kosten auswirken würden. Wenn ein Anbieter die Kosten Ihrer Datenbanklizenz ändert, untersuchen Sie, wie sich dies auf die Gesamteffizienz Ihrer Workload auswirkt. Berücksichtigen Sie historische Preisankündigungen von Ihren Anbietern für Trends bei Lizenzänderungen in ihren Produkten. Die Lizenzkosten können auch unabhängig vom Durchsatz oder der Nutzung skaliert werden, z. B. Lizenzen, die nach Hardware skaliert werden (CPU-gebundene Lizenzen). Diese Lizenzen sollten vermieden werden, da sich die Kosten ohne entsprechende Ergebnisse schnell erhöhen können.

Wenn Sie beispielsweise eine Amazon-EC2-Instance in us-east-1 mit einem Linux-Betriebssystem betreiben, können Sie die Kosten um etwa 45 % senken, verglichen mit einer anderen Amazon-EC2-Instance, die unter Windows läuft.

Der [AWS Pricing Calculator](#) bietet eine umfassende Möglichkeit, die Kosten verschiedener Ressourcen mit unterschiedlichen Lizenzoptionen zu vergleichen, z. B. Amazon-RDS-Instances und verschiedene Datenbank-Engines. Darüber hinaus bietet das AWS Cost Explorer eine unschätzbare Perspektive für die Kosten bestehender Workloads, insbesondere derjenigen, die mit verschiedenen Lizenzen einhergehen. Für die Lizenzverwaltung bietet [AWS License Manager](#) eine optimierte Methode zur Überwachung und Verwaltung von Softwarelizenzen. Kunden können ihre bevorzugte Open-Source-Software in der AWS Cloud bereitstellen und einsetzen.

Implementierungsschritte

- Analysieren der Lizenzoptionen: Überprüfen Sie die Lizenzbedingungen der verfügbaren Software. Suchen Sie nach Open-Source-Versionen, die über die erforderliche Funktionalität verfügen, und stellen Sie fest, ob die Vorteile der lizenzierten Software die Kosten überwiegen. Bei günstigen Bedingungen stimmen die Kosten der Software mit ihren Vorteilen überein.
- Analysieren des Softwareanbieters: Überprüfen Sie alle bisherigen Preis- oder Lizenzänderungen des Anbieters. Suchen Sie nach Änderungen, die nicht im Einklang mit den Ergebnissen stehen, wie z. B. Strafen für die Ausführung auf Hardware oder Plattformen bestimmter Anbieter. Achten Sie zudem darauf, wie mögliche Prüfungen und Strafen durchgeführt werden.

Ressourcen

Zugehörige Dokumente:

- [Open Source at AWS](#)
- [AWS-Rechner für Gesamtbetriebskosten \(TCO\)](#)
- [Amazon-S3-Speicherklassen](#)
- [Cloud-Produkte](#)

Zugehörige Beispiele:

- [Open-Source-Blogs](#)
- [Open-Source-Blogs von AWS](#)
- [Optimierung und Lizenzierungsbewertung](#)

COST05-BP05 Auswahl von Komponenten dieser Workload zur Optimierung der Kosten im Einklang mit den Prioritäten der Organisation

Berücksichtigen Sie bei der Auswahl sämtlicher Komponenten für Ihre Workload die Kosten. Dies umfasst die Nutzung von verwalteten Services und Services auf Anwendungsebene oder einer Serverless-, Container- oder ereignisgesteuerten Architektur, um die Gesamtkosten zu verringern. Minimieren Sie Lizenzkosten mithilfe von Open-Source-Software, Software, für die keine Lizenzgebühren anfallen, oder Alternativen zur Verringerung der Ausgaben.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Berücksichtigen Sie die Kosten von Services und Optionen, wenn Sie alle Komponenten auswählen. Dies beinhaltet auch die Verwendung von Services auf Anwendungsebene sowie verwalteter Services wie etwa [Amazon Relational Database Service \(Amazon RDS\)](#), [Amazon DynamoDB](#), [Amazon Simple Notification Service](#) (Amazon SNS) und [Amazon Simple Email Service](#) (Amazon SES) zur Reduzierung der Gesamtkosten der Organisation.

Verwenden Sie Serverless-Lösungen und Container für die Datenverarbeitung, zum Beispiel [AWS Lambda](#) und [Amazon Simple Storage Service](#) (Amazon S3) für statische Websites. Containerisieren Sie Ihre Anwendung wenn möglich und verwenden Sie verwaltete AWS-Container-Services wie [Amazon Elastic Container Service](#) (Amazon ECS) oder [Amazon Elastic Kubernetes Service](#) (Amazon EKS).

Minimieren Sie Lizenzkosten, indem Sie Open-Source-Software oder Software ohne Lizenzgebühren verwenden, wie z. B. Amazon Linux für Datenverarbeitungs-Workloads. Alternativ können Sie Datenbanken auch zu Amazon Aurora migrieren.

Sie können Serverless-Services oder Services auf Anwendungsebene wie [Lambda](#), [Amazon Simple Queue Service \(Amazon SQS\)](#), [Amazon SNS](#) und [Amazon SES](#) verwenden. Mit diesen Services müssen Sie keine Ressourcen mehr verwalten und sie stellen die Funktion der Codeausführung, Warteschlangenservices und Nachrichtenzustellung bereit. Der andere Vorteil besteht darin, dass die Leistung und Kosten entsprechend der Nutzung skaliert werden, was eine effiziente Kostenzuordnung ermöglicht.

Die Verwendung einer [ereignisgesteuerten Architektur](#) ist auch mit Serverless-Services möglich. Ereignisgesteuerte Architekturen sind Push-basiert, es geschieht also alles On-Demand, während das Ereignis im Router auftritt. So bezahlen Sie nicht für eine kontinuierliche Abfragung, um auf

ein Ereignis zu prüfen. Das Ergebnis: weniger Verbrauch der Netzwerkbandbreite, weniger CPU-Nutzung, weniger nicht genutzte Flottenkapazität und weniger SSL-/TLS-Handshakes.

Weitere Informationen zu Serverless finden Sie im Whitepaper [Serverless Applications Lens - Well-Architected Framework](#).

Implementierungsschritte

- Auswahl der einzelnen Services zur Kostenoptimierung: Wählen Sie unter Verwendung Ihrer Prioritätenliste und Analyse jede Option aus, die am besten mit Ihren Organisationsprioritäten übereinstimmt. Statt die Kapazität zu erhöhen, um die Nachfrage zu erfüllen, denken Sie über andere Optionen nach, die eine bessere Leistung mit geringeren Kosten bedeuten können. Wenn Sie beispielsweise den erwarteten Datenverkehr für Ihre Datenbanken in AWS prüfen müssen, können Sie entweder die Instance vergrößern oder Amazon-ElastiCache-Services (Redis oder Memcached) verwenden, um Ihren Datenbanken zwischengespeicherte Mechanismen bereitzustellen.
- Bewerten der ereignisgesteuerten Architektur: Durch die Verwendung einer Serverless-Architektur können Sie auch eine ereignisgesteuerte Architektur für verteilte, auf Microservices basierende Anwendungen erstellen. So erhalten Sie skalierbare, resiliente, agile und kostengünstige Lösungen.

Ressourcen

Zugehörige Dokumente:

- [AWS-Rechner für Gesamtbetriebskosten \(TCO\)](#)
- [AWS Serverless](#)
- [Was ist ereignisgesteuerte Architektur \(EDA\)?](#)
- [Amazon-S3-Speicherklassen](#)
- [Cloud-Produkte](#)
- [Amazon ElastiCache \(Redis OSS\)](#)

Zugehörige Beispiele:

- [Getting started with event-driven architecture](#)
- [Ereignisgesteuerte Architektur](#)
- [How Statsig runs 100x more cost-effectively using Amazon ElastiCache \(Redis OSS\)](#)

- [Best practices for working with AWS Lambda functions](#)

COST05-BP06 Durchführen einer Kostenanalyse für unterschiedliche Nutzungen im Lauf der Zeit

Workloads können sich im Laufe der Zeit ändern. Einige Services oder Features sind auf unterschiedlichen Nutzungsebenen kostengünstiger. Wenn Sie jede Komponente im zeitlichen Verlauf und mit einer prognostizierten Nutzung analysieren, bleibt diese Workload über ihre gesamte Lebensdauer hinweg kostengünstig.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Wenn AWS neue Services und Features veröffentlicht, können sich die optimalen Services für Ihre Workload ändern. Der erforderliche Aufwand sollte potenzielle Vorteile widerspiegeln. Die Häufigkeit der Workload-Überprüfung hängt von den Anforderungen Ihrer Organisation ab. Wenn es sich um eine Workload mit erheblichen Kosten handelt, wird die Implementierung neuer Services früher die Kosteneinsparungen maximieren, sodass eine häufigere Überprüfung von Vorteil sein kann. Ein weiterer Auslöser für die Überprüfung ist die Änderung der Nutzungsmuster. Signifikante Änderungen bei der Nutzung können darauf hinweisen, dass alternative Services optimaler wären.

Wenn Sie Daten in AWS Cloud verschieben müssen, können Sie aus einer Vielzahl von AWS-Services und Partnertools auswählen, die Sie bei der Migration Ihrer Datensätze unterstützen, ganz gleich, ob es sich um Dateien, Datenbanken, Computerabbilder, Block-Volumes oder sogar Bandsicherungen handelt. Wenn Sie zum Beispiel große Datenmengen zu und von AWS verschieben oder Daten am Edge verarbeiten möchten, können Sie eines der speziell entwickelten AWS-Geräte verwenden, um kostengünstig Petabytes an Daten offline zu verschieben. Bei höheren Datenübertragungsraten kann ein Direct-Connect-Service beispielsweise günstiger als ein VPN sein und die erforderliche konsistente Konnektivität für Ihr Unternehmen bereitstellen.

Prüfen Sie Ihre Skalierungsaktivität basierend auf der Kostenanalyse für unterschiedliche Nutzungen im Laufe der Zeit. Analysieren Sie das Ergebnis, um herauszufinden, ob die Skalierungsrichtlinie so angepasst werden kann, dass Instances mit mehreren Instance-Typen und Kaufoptionen hinzugefügt werden können. Überprüfen Sie Ihre Einstellungen, um zu sehen, ob das Minimum zur Verarbeitung von Benutzeranfragen reduziert werden kann (jedoch mit einer kleineren Flottengröße), und fügen Sie mehr Ressourcen hinzu, um die erwartete hohe Nachfrage zu erfüllen.

Führen Sie eine Kostenanalyse für unterschiedliche Nutzungen im Lauf der Zeit durch, indem Sie mit Stakeholdern in Ihrer Organisation sprechen und das Prognosefeature von [AWS Cost](#)

[Explorer](#) verwenden, um die potenziellen Auswirkungen von Serviceänderungen zu prognostizieren. Überwachen Sie Auslöser auf Nutzungsebene mithilfe von AWS Budgets, CloudWatch-Fakturierungsalarmen und AWS Cost Anomaly Detection, um die kosteneffektivsten Services früher zu identifizieren und zu implementieren.

Implementierungsschritte

- Definieren vorhergesagter Nutzungsmuster: Dokumentieren Sie in Zusammenarbeit mit Organisationbereichen, wie z. B. Marketing- und Produktbesitzern, wie die erwarteten und vorausgesagten Nutzungsmuster für die Workload aussehen werden. Sprechen Sie mit Business-Stakeholdern über historische und prognostizierte Kosten und gestiegene Nutzungen und stellen Sie sicher, dass solche Steigerungen mit den Geschäftsanforderungen übereinstimmen. Ermitteln Sie Kalendertage, -wochen oder -monate, in denen Sie mit einer erhöhten Nutzung Ihrer AWS-Ressourcen rechnen. Dies bedeutet, dass Sie die Kapazität der vorhandenen Ressourcen erhöhen oder zusätzliche Services einführen sollten, um die Kosten zu senken und die Leistung zu steigern.
- Durchführen einer Kostenanalyse bei vorhergesagter Nutzung: Führen Sie mithilfe der definierten Nutzungsmuster die Analyse an jedem dieser Punkte durch. Der Analyseaufwand sollte das potenzielle Ergebnis widerspiegeln. Wenn beispielsweise die Änderung der Nutzung groß ist, sollte eine gründliche Analyse durchgeführt werden, um etwaige Kosten und Änderungen zu überprüfen. Mit anderen Worten: Wenn die Kosten steigen, sollte auch die Nutzung für Unternehmen zunehmen.

Ressourcen

Zugehörige Dokumente:

- [AWS Rechner für Gesamtbetriebskosten \(TCO\)](#)
- [Amazon-S3-Speicherklassen](#)
- [Cloud-Produkte](#)
- [Amazon EC2 Auto Scaling](#)
- [Cloud-Datenmigration](#)
- [AWS Snow Family](#)

Zugehörige Videos:

- [AWS OpsHub for Snow Family](#)

KOSTEN 6. Wie können Sie bei der Auswahl des Ressourcentyps, -umfangs und der Anzahl der Ressourcen Kostenziele erfüllen?

Stellen Sie sicher, dass Sie den geeigneten Ressourcenumfang und die Anzahl der Ressourcen für die jeweilige Aufgabe auswählen. Durch die Auswahl des kostengünstigsten Typs, Umfangs und der kostengünstigsten Anzahl minimieren Sie die Verschwendung von Ressourcen.

Best Practices

- [COST06-BP01 Durchführen einer Kostenmodellierung](#)
- [COST06-BP02 Auswahl von Ressourcentyp, -größe und -anzahl basierend auf Daten](#)
- [COST06-BP03 Auswahl von Ressourcentyp, -umfang und -anzahl basierend auf Metriken](#)
- [COST06-BP04 Erwägen Sie die Verwendung gemeinsam genutzter Ressourcen](#)

COST06-BP01 Durchführen einer Kostenmodellierung

Identifizieren Sie die Anforderungen der Organisation (z. B. Geschäftsanforderungen und bestehende Verpflichtungen) und führen Sie eine Kostenmodellierung (Gesamtkosten) der Workload und aller ihrer Komponenten durch. Führen Sie Benchmark-Aktivitäten für die Workload unter verschiedenen prognostizierten Belastungen durch und vergleichen Sie die Kosten. Der Modellierungsaufwand sollte in einem angemessenen Verhältnis zu dem potenziellen Nutzen stehen, z. B. muss der Zeitaufwand den Komponentenkosten entsprechen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Führen Sie eine Kostenmodellierung für Ihren Workload und jede ihrer Komponenten durch, um das Gleichgewicht zwischen Ressourcen zu verstehen und die richtige Größe für jede Ressource in dem Workload zu finden, unter Berücksichtigung eines bestimmten Leistungsgrads. Ein Verständnis der Kostenerwägungen kann den Geschäftsfall und die Entscheidungsfindung Ihrer Organisation bei der Bewertung der Ergebnisse der Wertrealisierung für die geplante Workload-Bereitstellung unterstützen.

Führen Sie Benchmark-Aktivitäten für die Workload unter verschiedenen prognostizierten Belastungen durch und vergleichen Sie die Kosten. Der Modellierungsaufwand sollte in einem angemessenen Verhältnis zu dem potenziellen Nutzen stehen, z. B. muss der Zeitaufwand proportional zu den Komponentenkosten oder prognostizierten Einsparungen sein. Die bewährten

Methoden hierzu finden Sie im [Abschnitt „Prüfverfahren“ im Whitepaper zur Säule „Leistungseffizienz“ des AWS Well-Architected Framework](#).

Beispiel: Zur Erstellung einer Kostenmodellierung für einen Workload, der aus Datenverarbeitungsressourcen besteht, kann [AWS Compute Optimizer](#) Sie bei der Kostenmodellierung für die Ausführung von Workloads unterstützen. Es bietet Empfehlungen zur richtigen Dimensionierung für Datenverarbeitungsressourcen basierend auf der bisherigen Nutzung. Stellen Sie sicher, dass CloudWatch-Agents in den Amazon-EC2-Instances bereitgestellt sind, um Speichermetriken zu sammeln, die Ihnen helfen, genauere Empfehlungen innerhalb von AWS Compute Optimizer abzugeben. Dies ist die ideale Datenquelle für Datenverarbeitungsressourcen, da es sich um einen kostenlosen Service handelt, der Machine Learning nutzt, um je nach Risikograd mehrere Empfehlungen zu geben.

Es gibt [mehrere Services](#), die Sie mit benutzerdefinierten Protokollen als Datenquellen für Dimensionierungen für andere Services und Workload-Komponenten verwenden können, wie z. B. [AWS Trusted Advisor](#), [Amazon CloudWatch](#) und [Amazon CloudWatch Logs](#). AWS Trusted Advisor prüft Ressourcen und kennzeichnet sie bei geringer Auslastung, was Ihnen helfen kann, Ihre Ressourcen richtig zu dimensionieren und ein Kostenmodell zu erstellen.

Im Folgenden finden Sie Empfehlungen für Daten und Metriken zur Kostenmodellierung:

- Die Überwachung muss die Benutzererfahrung genau widerspiegeln. Wählen Sie die richtige Detaillierung für die Dauer aus, und wählen Sie das Maximum oder den 99. Perzentil statt des Durchschnitts aus.
- Wählen Sie die richtige Aufschlüsselung für die Dauer der Analyse aus, die für die Deckung der Workload-Zyklen erforderlich ist. Bei einer zweiwöchigen Analyse könnten Sie beispielsweise einen monatlichen Zyklus mit hoher Nutzung übersehen, der zu einer Unterbereitstellung führen könnte.
- Wählen Sie die richtigen AWS-Services für Ihre geplante Workload danach, wie Ihre bestehenden Verpflichtungen, ausgewählten Preismodelle für andere Workloads und die Fähigkeit, Innovationen schneller umzusetzen und sich auf Ihren Kerngeschäftswert zu konzentrieren, aussehen.

Implementierungsschritte

- Durchführen einer Kostenmodellierung: Stellen Sie die Workload oder einen Machbarkeitsnachweis in einem separaten Konto mit den spezifischen zu testenden Ressourcentypen und -größen bereit. Führen Sie die Workload mit den Testdaten aus und zeichnen die Ergebnisse zusammen mit den Kostendaten zum Zeitpunkt der Testausführung auf. Anschließend stellen Sie die Workload erneut bereit oder ändern die Ressourcentypen und -umfänge und führen den Test noch

einmal aus. Fügen Sie die Lizenzgebühren für alle Produkte, die Sie möglicherweise mit diesen Ressourcen verwenden, sowie die geschätzten Betriebskosten (Arbeits- oder Ingenieurkosten) für die Bereitstellung und Verwaltung dieser Ressourcen bei der Erstellung der Kostenmodelle hinzu. Erwägen Sie eine Kostenmodellierung für einen bestimmten Zeitraum (stündlich, täglich, monatlich, jährlich oder drei Jahre).

Ressourcen

Zugehörige Dokumente:

- [AWS Auto Scaling](#)
- [Identifizieren von Möglichkeiten zur Größenanpassung](#)
- [Amazon-CloudWatch-Features](#)
- [Cost Optimization: Amazon EC2 Right Sizing](#)
- [AWS Compute Optimizer](#)
- [AWS Pricing Calculator](#)

Zugehörige Beispiele:

- [Perform a Data-Driven Cost Modeling](#)
- [Wie schätze ich die Kosten meiner geplanten AWS-Ressourcenkonfigurationen?](#)
- [Wählen der richtigen AWS-Tools](#)

COST06-BP02 Auswahl von Ressourcentyp, -größe und -anzahl basierend auf Daten

Wählen Sie die Ressourcengröße oder den -typ basierend auf Daten zur Workload und der Ressourcenmerkmale aus. Zu berücksichtigen sind hier beispielsweise Datenverarbeitung, Speicher, Durchsatz oder Schreibintensität. Diese Auswahl erfolgt in der Regel unter Verwendung einer früheren (On-Premises)-Version der Workload, der Dokumentation oder anderer Informationsquellen über die Workload.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Amazon EC2 bietet eine große Auswahl an Instance-Typen mit unterschiedlichen CPU-, Arbeitsspeicher-, Speicher- und Netzwerkkapazitäten für verschiedene Anwendungsfälle. Diese

Instance-Typen bieten unterschiedliche Kombinationen von CPU-, Arbeitsspeicher-, Speicher- und Netzwerkkapazitäten, sodass Sie bei der Wahl der richtigen Ressourcenkombination für Ihre Projekte flexibel sind. Jeder Instance-Typ ist in mehreren Größen verfügbar, sodass Sie Ihre Ressourcen an die Anforderungen Ihrer Workload anpassen können. Um herauszufinden, welchen Instance-Typ Sie benötigen, informieren Sie sich über die Systemanforderungen der Anwendung oder Software, die Sie auf Ihrer Instance ausführen möchten. Diese Angaben sollten Folgendes umfassen:

- Betriebssystem
- Anzahl der CPU-Kerne
- GPU-Kerne
- Größe des Systemspeichers (RAM)
- Speichertyp und Umgebung
- Anforderung an die Netzwerkbandbreite

Ermitteln Sie den Zweck der Rechenanforderungen und welche Instance benötigt wird, um anschließend die verschiedenen Familien von Amazon-EC2-Instances zu untersuchen. Amazon bietet die folgenden Instance-Typfamilien an:

- Allgemeine Zwecke
- Für Datenverarbeitung optimiert
- RAM-optimiert
- Speicheroptimiert
- Beschleunigte Datenverarbeitung
- HPC-optimiert

Für ein tiefergehendes Verständnis der jeweiligen Zwecke und Anwendungsfälle, die eine bestimmte Amazon-EC2-Instance-Familie erfüllen kann, siehe [AWS-Instance-Typen](#).

Die Erfassung der Systemanforderungen ist entscheidend, damit Sie die passende Instance-Familie und den geeigneten Instance-Typ für Ihre Anforderungen auswählen können. Die Namen der Instance-Typen setzen sich aus dem Familiennamen und der Größe der Instance zusammen. Die Instance „t2.micro“ zum Beispiel gehört zur T2-Familie und entspricht der Micro-Größe.

Wählen Sie die Ressourcengröße oder den -typ basierend auf der Workload und den Ressourcenmerkmalen aus (beispielsweise Datenverarbeitung, Speicher, Durchsatz oder

Schreibintensität). Diese Auswahl erfolgt in der Regel unter Verwendung der Kostenmodellierung, einer früheren Version des Workloads (z. B. einer On-Premises-Version), mithilfe der Dokumentation oder unter Verwendung anderer Informationsquellen über den Workload (Whitepaper, veröffentlichte Lösungen). Die Verwendung von AWS-Preisrechnern oder Kostenmanagement-Tools kann dabei helfen, fundierte Entscheidungen über Instance-Typen, -Größen und -Konfigurationen zu treffen.

Implementierungsschritte

- Auswahl von Ressourcen anhand von Daten: Verwenden Sie Ihre Kostenmodellierungsdaten, um den erwarteten Workload-Nutzungsgrad auszuwählen, und wählen Sie den angegebenen Ressourcentyp und die -größe aus. Bestimmen Sie auf Grundlage der Kostenmodellierungsdaten die Anzahl der virtuellen CPUs, den Gesamtspeicher (GiB), die lokale Instance-Speichervolumen (GB), die Amazon EBS-Volumes und das Leistungsniveau des Netzwerks unter Berücksichtigung der für die Instance erforderlichen Datenübertragungsrate. Treffen Sie Ihre Auswahl stets auf Grundlage detaillierter Analysen und genauer Daten, um die Leistung zu optimieren und gleichzeitig die Kosten effektiv zu verwalten.

Ressourcen

Zugehörige Dokumente:

- [AWS Instance-Typen](#)
- [AWS Auto Scaling](#)
- [Amazon-CloudWatch-Features](#)
- [Kostenoptimierung: Richtige EC2-Dimensionierung](#)

Zugehörige Videos:

- [Selecting the right Amazon EC2 instance for your workloads](#)
- [Right Size Your Services](#)

Zugehörige Beispiele:

- [It just got easier to discover and compare Amazon EC2 instance types](#)

COST06-BP03 Auswahl von Ressourcentyp, -umfang und -anzahl basierend auf Metriken

Nutzen Sie Metriken aus dem derzeit aktiven Workload für die Auswahl des richtigen Umfangs und Typs, um Kosten zu optimieren. Sorgen Sie für die richtige Bereitstellung von Durchsatz, Umfang und Speicher für Computing-, Speicher-, Daten- und Netzwerkservices. Dies kann mit einer Feedback-Schleife wie Auto Scaling oder durch benutzerdefinierten Code im Workload erfolgen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Niedrig

Implementierungsleitfaden

Erstellen Sie eine Feedback-Schleife innerhalb des Workloads, die aktive Metriken aus dem laufenden Workload verwendet, um Änderungen an diesem Workload vorzunehmen. Sie können einen verwalteten Service wie [AWS Auto Scaling](#) verwenden, den Sie so konfigurieren, dass er die richtigen Dimensionierungsvorgänge für Sie durchführt. AWS stellt außerdem [APIs](#), [SDKs](#) und Features bereit, mit denen Ressourcen mit minimalem Aufwand angepasst werden können. Sie können eine Workload so programmieren, dass eine Amazon-EC2-Instance angehalten und gestartet wird, um eine Änderung der Instance-Größe oder des Instance-Typs zuzulassen. Dies bietet die Vorteile der richtigen Dimensionierung und eliminiert nahezu alle Betriebskosten, die für die Änderung erforderlich sind.

Einige AWS-Services verfügen über eine automatische Auswahl von Typ oder Größe, z. B. [Amazon Simple Storage Service Intelligent-Tiering](#). Amazon S3 Intelligent-Tiering verschiebt Ihre Daten basierend auf Ihren Nutzungsmustern automatisch zwischen den zwei Zugriffsebenen, häufiger Zugriff und seltener Zugriff.

Implementierungsschritte

- Steigern der Beobachtbarkeit durch Konfigurieren von Workload-Metriken: Erfassen Sie wichtige Metriken für die Workload. Diese Metriken geben die Kundenerfahrung an, z. B. die Workload-Ausgabe. Sie passen sich außerdem an die Unterschiede zwischen Ressourcentypen und -umfängen, z. B. CPU- und Speichernutzung, an. Analysieren Sie bei Datenverarbeitungsressourcen Leistungsdaten, um die Größe der Amazon-EC2-Instances richtig zu bemessen. Ermitteln Sie inaktive und nicht ausgelastete Instances. Schlüsselmetriken sind CPU- und Speicherauslastung (z. B. 40 % CPU-Auslastung in 90 % der Zeit, wie im Artikel zum [Ermitteln der richtigen Dimensionierung, wenn AWS Compute Optimizer und die Arbeitsspeicherauslastung aktiviert sind](#), beschrieben). Ermitteln Sie Instances mit einer maximalen CPU- und Speicherauslastung von unter 40 % in einem Zeitraum von vier Wochen. Bei diesen Instances sollte die Größe angepasst werden, um die Kosten zu reduzieren. Für

Speicherressourcen wie Amazon S3 können Sie [Amazon S3 Storage Lens](#) verwenden. Hiermit sehen Sie standardmäßig 28 Metriken aus unterschiedlichen Kategorien auf Bucket-Ebene sowie Verlaufsdaten aus 14 Tagen im Dashboard. Sie können das Dashboard von Amazon S3 Storage Lens nach Übersichtswerten und Kostenoptimierung oder nach Ereignissen sortieren, um bestimmte Metriken zu analysieren.

- Anzeigen von Empfehlungen zur Umfangsanpassung: Anhand der Empfehlungen in AWS Compute Optimizer und dem Amazon-EC2-Tool zur Umfangsanpassung in der Kostenverwaltungskonsole oder durch Prüfen der Umfangsanpassung für Ressourcen in AWS Trusted Advisor können Sie Anpassungen an Ihrer Workload vornehmen. Achten Sie darauf, die [richtigen Tools](#) zur Umfangsanpassung verschiedener Ressourcen zu verwenden, und halten Sie sich an die [Richtlinien für die Dimensionierung](#), abhängig davon, ob es sich um eine Amazon-EC2-Instance, AWS-Speicherklassen oder Amazon-RDS-Instance-Typen handelt. Bei Speicherressourcen können Sie Amazon S3 Storage Lens verwenden. Hiermit erhalten Sie Einblicke in die Objektspeichernutzung und Aktivitätstrends und finden Empfehlungen zur Kostenoptimierung und zum Anwenden von bewährten Methoden zum Schutz der Daten. Anhand der kontextbezogenen Empfehlungen, die [Amazon S3 Storage Lens](#) aus der Analyse von Metriken in Ihrer Organisation ableitet, können Sie direkt Schritte zur Speicheroptimierung ergreifen.
- Automatische Auswahl des Ressourcentyps und des Umfangs basierend auf Metriken: Mithilfe der Workload-Metriken können Sie Ihre Workload-Ressourcen manuell oder automatisch auswählen. Bei Computing-Ressourcen kann die Konfiguration von AWS Auto Scaling oder die Implementierung von Code in Ihrer Anwendung den Aufwand reduzieren, der bei häufigen Änderungen erforderlich ist. So lassen sich Änderungen möglicherweise früher implementieren, als dies mit einem manuellen Prozess der Fall wäre. Sie können eine Flotte von On-Demand-Instances und Spot-Instances innerhalb einer einzigen Auto-Scaling-Gruppe starten und automatisch skalieren. Zusätzlich zum Erhalt von Rabatten für die Verwendung von Spot-Instances können Sie mit Reserved Instances oder einem Savings Plan Rabatte auf die regulären On-Demand-Instance-Preise erhalten. Durch die Kombination dieser Faktoren sparen Sie Kosten für Amazon-EC2-Instances und können die gewünschte Skalierung und Leistung für Ihre Anwendung festlegen. Sie können auch eine [Strategie der attributbasierten Auswahl des Instance-Typs \(ABS\)](#) in [Auto-Scaling-Gruppen \(ASG\)](#) verwenden und so die Instance-Anforderungen in Form einer Gruppe von Attributen ausdrücken, z. B. vCPU, Arbeitsspeicher und Speicher. Mit Amazon EC2 Spot Instances können Sie automatisch Instance-Typen neuerer Generationen verwenden, und mit Amazon EC2 Spot Instances auf ein größeres Kapazitätsangebot zugreifen. Amazon-EC2-Flotte und Amazon EC2 Auto Scaling wählen Instances aus, die den angegebenen Attributen entsprechen, und starten diese. So müssen Sie Instance-Typen nicht mehr manuell auswählen. Für Speicherressourcen können Sie die Features [Amazon S3 Intelligent-Tiering](#) und [Amazon](#)

[EFS Infrequent Access](#) nutzen. Hiermit werden automatisch die Speicherklassen ausgewählt, die automatisch zur Einsparung von Speicherkosten führen, wenn sich Datenzugriffsmuster ändern, ohne Leistungsbeeinträchtigungen oder Betriebsaufwand.

Ressourcen

Zugehörige Dokumente:

- [AWS Auto Scaling](#)
- [AWS Right-Sizing](#)
- [AWS Compute Optimizer](#)
- [Amazon-CloudWatch-Features](#)
- [CloudWatch Getting Set Up](#)
- [CloudWatch Publishing Custom Metrics](#)
- [Erste Schritte mit Amazon EC2 Auto Scaling](#)
- [Amazon S3 Storage Lens](#)
- [Amazon S3 Intelligent Tiering](#)
- [Amazon EFS Infrequent Access](#)
- [Starten einer Amazon-EC2-Instance mit dem SDK](#)

Zugehörige Videos:

- [Right Size Your Services](#)

Zugehörige Beispiele:

- [Attribute based Instance Type Selection for Auto Scaling for Amazon EC2 Fleet](#)
- [Optimizing Amazon Elastic Container Service for cost using scheduled scaling](#)
- [Predictive scaling with Amazon EC2 Auto Scaling](#)
- [Optimieren Sie Kosten und erhalten Sie Einblick in die Nutzung mit Amazon S3 Storage Lens](#)

COST06-BP04 Erwägen Sie die Verwendung gemeinsam genutzter Ressourcen

Für Services, die bereits auf Organisationsebene für mehrere Geschäftseinheiten bereitgestellt werden, sollten Sie die Verwendung gemeinsam genutzter Ressourcen erwägen, um die Auslastung zu erhöhen und die Gesamtbetriebskosten (TCO) zu senken. Die Verwendung gemeinsam genutzter Ressourcen kann eine kostengünstige Option sein, um Verwaltung und Kosten zu zentralisieren, indem bestehende Lösungen oder gemeinsam genutzte Komponenten oder beides verwendet werden. Verwalten Sie allgemeine Funktionen wie Überwachung, Backups und Konnektivität entweder innerhalb einer Kontogrenze oder in einem dedizierten Konto. Sie können auch die Kosten senken, indem Sie Standardisierung implementieren und Doppelarbeit sowie Komplexität reduzieren.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Wenn mehrere Workloads dieselbe Funktion ausführen, verwenden Sie vorhandene Lösungen und gemeinsam genutzte Komponenten, um Verwaltung und Kosten zu optimieren. Erwägen Sie die Nutzung vorhandener Ressourcen (insbesondere gemeinsam genutzter Ressourcen), z. B. Datenbankserver oder Verzeichnisservices, die nicht zur Produktion verwendet werden, um die Cloud-Kosten zu senken, indem Sie bewährte Sicherheitsmethoden und Organisationsvorschriften befolgen. Für eine optimale Wertschöpfung und Effizienz ist entscheidend, die Kosten (mithilfe von Kostenauflistung und Rückbuchung) den relevanten Geschäftsbereichen zuzuordnen, die den Konsum antreiben.

Kostenauflistung bezieht sich auf Berichte, in denen die Cloud-Kosten in zuteilbare Kategorien wie Verbraucher, Geschäftseinheiten, Hauptbuchkonten oder andere verantwortliche Entitäten unterteilt werden. Mit Kostenauflistungen sollen Teams, Geschäftseinheiten oder Einzelpersonen die Kosten ihrer verbrauchten Cloud-Ressourcen mitgeteilt werden.

Rückbuchung bedeutet, zentrale Serviceausgaben den Kostenträgern zuzuordnen, und zwar auf der Grundlage einer Strategie, die für einen bestimmten Finanzmanagementprozess geeignet ist. Für Kunden werden bei einer Rückbuchung die Kosten, die von einem Shared-Services-Konto anfallen, verschiedenen Finanzkostenkategorien zugeordnet, die für einen Kundenberichtsprozess geeignet sind. Durch die Einrichtung von Rückbuchungsmechanismen können Sie die Kosten melden, die verschiedenen Geschäftseinheiten, Produkten und Teams entstanden sind.

Workloads können als kritisch und unkritisch eingestuft werden. Verwenden Sie auf der Grundlage dieser Klassifizierung gemeinsam genutzte Ressourcen mit allgemeinen Konfigurationen für weniger kritische Workloads. Reservieren Sie dedizierte Server ausschließlich für kritische Workloads, um die Kosten weiter zu optimieren. Teilen Sie Ressourcen oder stellen Sie sie für

mehrere Konten bereit, um sie effizient zu verwalten. Selbst in unterschiedlichen Entwicklungs-, Test- und Produktionsumgebungen ist eine sichere gemeinsame Nutzung möglich, ohne die Organisationsstruktur zu beeinträchtigen.

Verwenden Sie Daten zur Zuordnung geteilter Kosten, mit deren Hilfe Sie die Kosten einzelner Geschäftsentitäten basierend auf der Verwendung gemeinsam genutzter Computing- und Speicherressourcen durch die Anwendung zuordnen können, um Ihr Verständnis zu verbessern und die Kosten und Nutzung für containerisierte Anwendungen zu optimieren. Daten zur Zuordnung geteilter Kosten helfen Ihnen dabei, bei Container-Workloads, die in Amazon Elastic Container Service (Amazon ECS) oder Amazon Elastic Kubernetes Service (Amazon EKS) ausgeführt werden, Kostenauflistung und Rückbuchung auf Aufgabenebene zu erreichen.

Erstellen Sie für verteilte Architekturen eine Shared-Services-VPC, die den zentralisierten Zugriff auf gemeinsam genutzte Services ermöglicht, die für Workloads in allen VPCs erforderlich sind. Diese gemeinsam genutzten Services können Ressourcen wie Verzeichnisservices oder VPC-Endpunkte umfassen. Zur Reduzierung des Verwaltungsaufwands und der Kosten empfiehlt die gemeinsame Nutzung von Ressourcen von einem zentralen Standort, anstatt sie in jeder einzelnen VPC zu erstellen.

Durch die Verwendung gemeinsam genutzter Ressourcen können Sie Betriebskosten sparen, die Ressourcenauslastung maximieren und die Konsistenz verbessern. In einem Design mit mehreren Konten können Sie einige AWS-Services zentral hosten und über mehrere Anwendungen und Konten an einem zentralen Punkt darauf zugreifen, um Kosten zu sparen. Sie können mit [AWS Resource Access Manager \(AWS RAM\)](#) weitere gängige Ressourcen freigeben, z. B. [VPC-Subnetze und AWS Transit Gateway-Anhänge](#), [AWS Network Firewall](#) oder [Pipelines in Amazon SageMaker AI](#). In einer Mehrkonten-Umgebung ermöglicht AWS RAM die einmalige Erstellung einer Ressource und ihre Freigabe für andere Konten.

Organisationen sollten die geteilten Kosten effektiv markieren und sicherstellen, dass kein erheblicher Teil ihrer Kosten unmarkiert oder nicht zugewiesen ist. Wenn Sie die gemeinsamen Kosten nicht effektiv verteilen und niemand die Verantwortung für die Verwaltung gemeinsamer übernimmt, können die Kosten für eine gemeinsame Cloud in die Höhe schießen. Sie müssen sich bewusst sein, wo Kosten auf Ressourcen-, Workload-, Team- oder Organisationsebene entstanden sind, da dieses Wissen Ihr Verständnis für den auf der jeweiligen Ebene geschaffenen Mehrwert im Vergleich zu den erzielten Geschäftsergebnissen verbessert. Letztlich profitieren Organisationen von Kosteneinsparungen, die sich aus der gemeinsamen Nutzung der Cloud-Infrastruktur ergeben. Fördern Sie die Kostenzuordnung für gemeinsam genutzte Cloud-Ressourcen, um die Cloud-Ausgaben zu optimieren.

Implementierungsschritte

- Vorhandene Ressourcen bewerten: Prüfen Sie bestehende Workloads, die ähnliche Services für Ihre Workload verwenden. Ziehen Sie abhängig von den Komponenten des Workloads vorhandene Plattformen in Betracht, sofern die Geschäftslogik oder die technischen Anforderungen dies zulassen.
- Gemeinsame Nutzung von Ressourcen in AWS RAM verwenden und entsprechend einschränken: Verwenden Sie AWS RAM, um Ressourcen mit anderen AWS-Konten innerhalb Ihrer Organisation zu teilen. Wenn Sie Ressourcen gemeinsam nutzen, müssen Sie Ressourcen nicht in mehreren Konten duplizieren, wodurch der betriebliche Aufwand der Ressourcenverwaltung minimiert wird. Dieser Prozess unterstützt die sichere Freigabe der Ressourcen, die Sie erstellt haben, an Rollen und Benutzer:innen in Ihrem Konto sowie an andere AWS-Konten.
- Ressourcen taggen: Taggen Sie Ressourcen, die für die Kostenberichterstattung infrage kommen, und kategorisieren Sie sie in Kostenkategorien. Aktivieren Sie diese kostenbezogenen Ressourcen-Tags für die Kostenzuordnung, um sich einen Überblick über den AWS-Ressourcenverbrauch zu verschaffen. Achten Sie darauf, ein angemessenes Maß an Granularität in Bezug auf Kosten- und Nutzungstransparenz zu schaffen, und beeinflussen Sie das Cloud-Nutzungsverhalten durch Kostenzuordnungsberichte und KPI-Tracking.

Ressourcen

Zugehörige bewährte Methoden:

- [SEC03-BP08 Sicheres gemeinsames Nutzen von Ressourcen in Ihrer Organisation](#)

Zugehörige Dokumente:

- [Was ist AWS Resource Access Manager?](#)
- [AWS-Services, die Sie mit AWS Organizations](#) verwenden können
- [Freigabefähige AWS-Ressourcen](#)
- [AWS Cost and Usage \(CUR\) Queries](#)

Zugehörige Videos:

- [AWS Resource Access Manager - granular access control with managed permissions](#)
- [How to design your AWS cost allocation strategy](#)

- [AWS Cost Categories](#)

Zugehörige Beispiele:

- [How-to chargeback shared services: An AWS Transit Gateway example](#)
- [How to build a chargeback/showback model for Savings Plans using the CUR](#)
- [Using VPC Sharing for a Cost-Effective Multi-Account Microservice Architecture](#)
- [Improve cost visibility of Amazon EKS with AWS Split Cost Allocation Data](#)
- [Improve cost visibility of Amazon ECS and AWS Batch with AWS Split Cost Allocation Data](#)

KOSTEN 7. Wie können Sie Kosten mithilfe von Preismodellen senken?

Verwenden Sie das Preismodell, das sich für Ihre Ressourcen am besten eignet. So halten Sie die Ausgaben möglichst niedrig.

Best Practices

- [COST07-BP01 Durchführen einer Preismodellanalyse](#)
- [COST07-BP02 Auswählen von Regionen auf Basis der Kosten](#)
- [COST07-BP03 Auswahl von Drittanbietervereinbarungen mit kosteneffizienten Bedingungen](#)
- [COST07-BP04 Implementieren von Preismodellen für alle Komponenten dieser Workload](#)
- [COST07-BP05 Durchführen einer Preismodellanalyse auf Verwaltungskontoebene](#)

COST07-BP01 Durchführen einer Preismodellanalyse

Analysieren Sie die einzelnen Komponenten der Workload. Stellen Sie fest, ob die Komponente und die Ressourcen über einen längeren Zeitraum (für Bindungsrabatte) oder dynamisch und kurz ausgeführt werden (für Spot- oder On-Demand-Zwecke). Analysieren Sie die Workload mithilfe der Empfehlungen in Tools für die Kostenverwaltung und wenden Sie Geschäftsregeln auf diese Empfehlungen an, um hohe Erträge zu erzielen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

AWS verfügt über mehrere [Preismodelle](#), mit denen Sie für Ihre Ressourcen auf die kostengünstigste Art und Weise bezahlen können, die den Anforderungen Ihrer Organisation entspricht und vom

jeweiligen Produkt abhängt. Arbeiten Sie mit Ihren Teams zusammen, um das am besten geeignete Preismodell zu bestimmen. Häufig besteht das Preismodell aus einer Kombination aus verschiedenen Optionen, die sich nach Ihrer Verfügbarkeit richtet.

Bei On-Demand-Instances zahlen Sie für die Datenverarbeitungs- oder Datenbankkapazitäten auf Stunden- oder Sekundenbasis (mindestens 60 Sekunden), je nachdem, welche Instances Sie ausführen. Es sind keine langfristigen Verpflichtungen oder Vorauszahlungen erforderlich.

Bei Savings Plans handelt es sich um ein flexibles Preismodell, das günstige Preise für die Nutzung von Amazon EC2, Lambda und AWS Fargate bietet. Im Gegenzug verpflichten Sie sich zu einer konstanten Nutzungsmenge (gemessen in Dollar/Stunde) für die Dauer von einem Jahr oder drei Jahren.

Spot Instances sind ein Preismechanismus für Amazon EC2, der es ermöglicht, ohne Vorabverpflichtungen freie Datenverarbeitungskapazität zu einem ermäßigten Stundensatz (bis zu 90 % Rabatt im Vergleich zum On-Demand-Preis) anzufordern.

Mit Reserved Instances zahlen Sie im Voraus für die Kapazität und erhalten bis zu 75 Prozent Rabatt. Weitere Informationen finden Sie unter [Kostenoptimierung mit Reservierungen](#).

Sie könnten Savings Plans für die mit der Produktion, der Qualität und den Entwicklungsumgebungen verbundenen Ressourcen hinzufügen. Da Sandbox-Ressourcen nur bei Bedarf aktiviert werden, könnten Sie alternativ ein On-Demand-Modell für die Ressourcen in dieser Umgebung wählen. Verwenden Sie [Spot Instances](#) von Amazon, um die Kosten für Amazon EC2 zu senken, oder verwenden Sie [Compute Savings Plans](#), um die Kosten für Amazon EC2, Fargate und Lambda zu reduzieren. Das Empfehlungstool [AWS Cost Explorer](#) stellt Möglichkeiten für an feste Kapazität gebundene Rabatte mit Savings Plans vor.

Wenn Sie in der Vergangenheit bereits [Reserved Instances](#) für Amazon EC2 erworben oder in Ihrer Organisation Verfahren zur Kostenzuteilung eingeführt haben, können Sie Amazon EC2 Reserved Instances vorerst weiterhin verwenden. Wir empfehlen jedoch, eine Strategie für die zukünftige Verwendung von Savings Plans als flexibleren Mechanismus zur Kostenreduzierung zu entwickeln. Sie können die Empfehlungen zu Savings Plans (SP) in AWS Cost Management jederzeit aktualisieren, um neue Empfehlungen zu Savings Plans zu generieren. Verwenden Sie Reserved Instances (RI), um die Kosten für Amazon RDS, Amazon Redshift, Amazon ElastiCache und Amazon OpenSearch Service zu reduzieren. Es stehen drei Optionen für Savings Plans und Reserved Instances zur Verfügung: vollständige Vorauszahlung, teilweise Vorauszahlung und keine Vorauszahlung. Nutzen Sie die in AWS Cost Explorer bereitgestellten Kaufempfehlungen für RI und SP.

Um Möglichkeiten für Spot-Workloads zu finden, verwenden Sie eine stündliche Ansicht Ihrer Gesamtnutzung und suchen Sie nach regelmäßigen Zeiträumen mit sich ändernder Nutzung oder Elastizität. Sie können Spot-Instances für verschiedene fehlertolerante und flexible Anwendungen verwenden. Beispiele sind statuslose Webserver, API-Endpunkte, Big-Data- und Analytikanwendungen, containerisierte Workloads, CI/CD und weitere flexible Workloads.

Ermitteln Sie, ob Ihre Amazon-EC2- und Amazon-RDS-Instances deaktiviert werden können, wenn sie nicht genutzt werden (nach Geschäftsschluss und am Wochenende). Dadurch können Sie die Kosten verglichen mit einem Einsatz rund um die Uhr um 70 % oder mehr reduzieren. Wenn Sie über Amazon-Redshift-Cluster verfügen, die nur zu bestimmten Zeiten verfügbar sein müssen, können Sie den Cluster anhalten und zu einem späteren Zeitpunkt neu starten. Wenn der Amazon-Redshift-Cluster oder die Amazon-EC2- und Amazon-RDS-Instances gestoppt werden, fallen keine Datenverarbeitungskosten mehr, sondern nur noch die Speichergebühren an.

Beachten Sie, dass es sich bei [On-Demand-Kapazitätsreservierungen](#) (On-Demand Capacity Reservations, ODCR) nicht um einen Preisnachlass handelt. Für Kapazitätsreservierungen wird Ihnen der entsprechende On-Demand-Tarif in Rechnung gestellt, unabhängig davon, ob Sie Instances mit der reservierten Kapazität ausführen oder nicht. Sie sollten in Betracht gezogen werden, wenn Sie ausreichend Kapazität für die Ressourcen bereitstellen müssen, die Sie ausführen möchten. ODCRs müssen nicht an langfristige Verpflichtungen gebunden sein. Sie können gekündigt werden, wenn Sie sie nicht mehr benötigen. Sie können jedoch auch von den Rabatten profitieren, die Savings Plans oder Reserved Instances bieten.

Implementierungsschritte

- Analysieren der Workload-Elastizität: Verwenden Sie die stündliche Granularität in Cost Explorer oder ein benutzerdefiniertes Dashboard, um die Elastizität Ihrer Workloads zu analysieren. Suchen Sie nach regelmäßigen Änderungen hinsichtlich der Anzahl der Instances, die ausgeführt werden. Instances mit kurzer Dauer sind Kandidaten für Spot Instances oder Spot-Flotte.
 - [Well-Architected Lab: Cost Explorer](#)
 - [Well-Architected Lab: Cost Visualization](#)
- Überprüfen bestehender Preisverträge: Überprüfen Sie laufende Verträge oder Verpflichtungen für langfristige Anforderungen. Analysieren Sie, was Sie aktuell haben und inwiefern diese Verpflichtungen genutzt werden. Nutzen Sie bereits vorhandene vertragliche Rabatte oder Unternehmensverträge. [Unternehmensverträge](#) bieten Kunden die Möglichkeit, Verträge so zu gestalten, dass sie ihren Bedürfnissen am besten entsprechen. Ziehen Sie bei langfristigen Verpflichtungen reservierte Preisrabatte, Reserved Instances oder Savings Plans für den spezifischen Instance-Typ, die Instance-Familie, AWS-Region und Availability Zones in Betracht.

- Durchführen einer Analyse des Bindungsrabatts: Sehen Sie sich unter Verwendung von Cost Explorer in Ihrem Konto die Empfehlungen für Savings Plans und Reserved Instances an. Um sicherzustellen, dass Sie die richtigen Empfehlungen mit den erforderlichen Rabatten und Risiken implementieren, befolgen Sie die [Well-Architected Labs](#).

Ressourcen

Zugehörige Dokumente:

- [Zugriff auf Empfehlungen für Reserved Instances](#)
- [Instance-Kaufoptionen](#)
- [AWS Enterprise](#)

Zugehörige Videos:

- [Save up to 90% and run production workloads on Spot](#)

Zugehörige Beispiele:

- [Well-Architected Lab: Cost Explorer](#)
- [Well-Architected Lab: Cost Visualization](#)
- [Well-Architected Lab: Pricing Models](#)

COST07-BP02 Auswählen von Regionen auf Basis der Kosten

Ressourcenpreise können je nach Region abweichen. Ermitteln Sie regionale Kostenunterschiede und stellen Sie nur in Regionen mit höheren Kosten bereit, um die Anforderungen an Latenzzeiten, Datenresidenz und Datensouveränität zu erfüllen. Die Berücksichtigung der Regionskosten sorgt dafür, dass Sie den niedrigsten Gesamtpreis für diese Workload zahlen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Die [AWS Cloud-Infrastruktur](#) ist global, wird an [mehreren Standorten weltweit](#) gehostet und basiert auf AWS-Regionen, Availability Zones, Local Zones, AWS Outposts und Wavelength Zones. Eine Region ist ein physischer Ort auf der Welt. Jede Region ist ein separates geografisches Gebiet,

in dem AWS mehrere Availability Zones hat. Availability Zones sind mehrere isolierte Standorte innerhalb jeder Region. Sie bestehen aus mindestens einem eigenständigen Rechenzentrum mit einer redundanten Stromversorgung, einem Netzwerk sowie Konnektivität.

Jede AWS-Region wird im Rahmen der jeweilig gültigen lokalen Marktbedingungen betrieben, und die Ressourcenpreise können von Region zu Region variieren, da es beispielsweise Unterschiede bei den Kosten für Land, Glasfaser, Strom und bei den Steuern gibt. Wählen Sie eine spezifische Region aus, in der Sie eine Komponente oder Ihre gesamte Lösung ausführen möchten, sodass Sie weltweit einen Betrieb zu den geringstmöglichen Kosten gewährleisten. Mit [AWS Calculator](#) können Sie die Kosten Ihrer Workload in verschiedenen Regionen einschätzen. Suchen Sie dazu Services nach Standorttyp (Region, Wavelength Zone und Local Zone) und Region.

Wenn Sie die Architektur Ihrer Lösungen aufbauen, hat es sich bewährt zu versuchen, Computing-Ressourcen zugunsten einer geringeren Latenz und einer stärkeren Datensouveränität näher an die Benutzer zu bringen. Wählen Sie den geografischen Standort auf der Grundlage Ihrer Geschäfts-, Datenschutz-, Leistungs- und Sicherheitsanforderungen. Verwenden Sie für Anwendungen mit globalen Endbenutzern mehrere Standorte.

Nutzen Sie Regionen, die niedrigere Preise für AWS-Services anbieten, um Ihre Workloads bereitzustellen, wenn Sie keine Verpflichtungen in Bezug auf Datenschutz, Sicherheit und geschäftliche Anforderungen haben. Wenn Ihre Standardregion zum Beispiel Asien-Pazifik (Sydney) (ap-southwest-2) ist und es keine Einschränkungen (z. B. Datenschutz, Sicherheit) für die Verwendung anderer Regionen gibt, ist die Bereitstellung nicht kritischer Amazon-EC2-Instances (Entwicklung und Test) in der Region USA Ost (Nord-Virginia) (us-east-1) kostengünstiger.

	<i>Compliance</i>	<i>Latenz</i>	<i>Kosten</i>	<i>Services/Funktionen</i>
Region 1	✓	15 ms	\$\$	✓
Region 2	✓	20 ms	\$\$\$	X
Region 3	✓	80 ms	\$	✓
Region 4	✓	15 ms	\$\$	✓
Region 5	✓	20 ms	\$\$\$	X
Region 6	✓	15 ms	\$	✓
Region 7	✓	80 ms	\$	✓
Region 8	✓	15 ms	\$	X

Matrixtabelle für Regionsfeatures

Die obige Matrixtabelle zeigt uns, dass Region 6 die beste Option für dieses gegebene Szenario ist, da die Latenz im Vergleich zu anderen Regionen gering ist, der Service verfügbar ist und es sich um die kostengünstigste Region handelt.

Implementierungsschritte

- **Überprüfen der AWS-Region-Preise:** Analysieren Sie die Workload-Kosten in der aktuellen Region. Berechnen Sie die Kosten in anderen verfügbaren Regionen, beginnend mit den höchsten Kosten nach Service und Verwendungstyp. Migrieren Sie in die neue Region, wenn die prognostizierte Einsparung die Kosten für das Verschieben der Komponente oder der Workload überwiegt.
- **Überprüfen der Anforderungen für Bereitstellungen in mehreren Regionen:** Analysieren Sie Ihre geschäftlichen Anforderungen und Verpflichtungen (Datenschutz, Sicherheit oder Leistung), um herauszufinden, ob für Sie Beschränkungen gelten, sodass Sie nicht mehrere Regionen verwenden können. Wenn Sie sich nicht auf eine einzelne Region beschränken müssen, verwenden Sie mehrere Regionen.
- **Analysieren der erforderlichen Datenübertragungen:** Berücksichtigen Sie bei der Auswahl von Regionen die Datenübertragungskosten. Halten Sie Ihre Daten in der Nähe des Kunden und in der Nähe der Ressourcen. Wählen Sie weniger kostenintensive AWS-Regionen,

in denen ein Datenfluss und nur minimale Datenübertragung besteht. Abhängig von Ihren Geschäftsanforderungen für die Datenübertragung können Sie [Amazon CloudFront](#), [AWS PrivateLink](#), [AWS Direct Connect](#) und [AWS Virtual Private Network](#) verwenden, um Ihre Nettwerkkosten zu senken, die Leistung zu verbessern und die Sicherheit zu erhöhen.

Ressourcen

Zugehörige Dokumente:

- [Zugriff auf Empfehlungen für Reserved Instances](#)
- [Amazon EC2 – Preise](#)
- [Instance-Kaufoptionen](#)
- [Tabelle „Region“](#)

Zugehörige Videos:

- [Save up to 90% and run production workloads on Spot](#)

Zugehörige Beispiele:

- [Überblick über die Datenübertragungskosten für gängige Architekturen](#)
- [Kostenerwägungen für globale Bereitstellungen](#)
- [What to Consider when Selecting a Region for your Workloads](#)

COST07-BP03 Auswahl von Drittanbietervereinbarungen mit kosteneffizienten Bedingungen

Kosteneffiziente Vereinbarungen und Bedingungen stellen sicher, dass die Kosten dieser Services mit den von ihnen bereitgestellten Vorteilen skaliert werden. Wählen Sie Vereinbarungen und Preise aus, die skaliert werden, wenn sie Ihrer Organisation zusätzliche Vorteile bieten.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Es gibt mehrere Produkte auf dem Markt, die Ihnen helfen, die Kosten für Ihre Cloud-Umgebungen zu verwalten. Sie unterscheiden sich teilweise in Bezug auf die Features, die von den Bedürfnissen der Kunden abhängen. So konzentrieren sich einige auf die Kostenkontrolle oder Kostentransparenz

und andere auf die Kostenoptimierung. Ein Schlüsselfaktor für eine effektive Kostenoptimierung und Governance ist die Verwendung des richtigen Tools mit den erforderlichen Features und dem richtigen Preismodell. Diese Produkte unterscheiden sich in ihren Preismodellen. Bei manchen wird ein bestimmter Prozentsatz Ihrer monatlichen Rechnung berechnet, bei anderen ein Prozentsatz Ihrer erzielten Einsparungen. Im Idealfall sollten Sie nur für das bezahlen, was Sie benötigen.

Wenn Sie Lösungen oder Services von Drittanbietern in der Cloud nutzen, ist es wichtig, dass die Preisstrukturen an Ihren gewünschten Ergebnissen ausgerichtet sind. Preise sollten mit den Ergebnissen und dem Wert skaliert werden, den sie bieten. Beispielsweise kostet Software, deren Preis auf einem Prozentsatz der erzielten Einsparungen basiert, umso mehr, je mehr Sie sparen (Ergebnis). Lizenzvereinbarungen, bei denen Sie mit steigenden Ausgaben mehr bezahlen, sind möglicherweise nicht immer in Ihrem Interesse, um die Kosten zu optimieren. Wenn der Anbieter jedoch klare Vorteile für alle Bestandteile Ihrer Rechnung bietet, könnte diese Preisstaffelung gerechtfertigt sein.

So kann beispielsweise eine Lösung, die Empfehlungen für Amazon EC2 bereitstellt und einen Prozentsatz Ihrer gesamten Rechnung berechnet, teurer werden, wenn Sie andere Services nutzen, die keinen Vorteil bieten. Ein weiteres Beispiel ist ein verwalteter Service, der zu einem Prozentsatz der Kosten für verwaltete Ressourcen in Rechnung gestellt wird. Eine höhere Instance-Größe erfordert möglicherweise nicht notwendigerweise mehr Verwaltungsaufwand, kann aber teurer werden. Stellen Sie sicher, dass diese Service-Preisvereinbarungen ein Kostenoptimierungsprogramm oder entsprechende Features in ihrem Service enthalten, um die Effizienz zu steigern.

Die Kunden finden diese auf dem Markt befindlichen Produkte vielleicht fortschrittlicher oder benutzerfreundlicher. Sie müssen die Kosten für diese Produkte berücksichtigen und über mögliche langfristige Kostenoptimierungen nachdenken.

Implementierungsschritte

- **Analyse von Vereinbarungen und Bedingungen Dritter:** Überprüfen Sie die Preise in Drittanbietervereinbarungen. Führen Sie die Modellierung für verschiedene Nutzungsebenen durch und berücksichtigen Sie neue Kosten, wie z. B. die Nutzung neuer Services oder Erweiterungen der aktuellen Services aufgrund des Workload-Wachstums. Entscheiden Sie, ob die zusätzlichen Kosten Ihrem Unternehmen die erforderlichen Vorteile bieten.

Ressourcen

Zugehörige Dokumente:

- [Zugriff auf Empfehlungen für Reserved Instances](#)
- [Instance-Kaufoptionen](#)

Zugehörige Videos:

- [Save up to 90% and run production workloads on Spot](#)

COST07-BP04 Implementieren von Preismodellen für alle Komponenten dieser Workload

Dauerhaft ausgeführte Ressourcen sollten reservierte Kapazität wie Savings Plans oder Reserved Instances nutzen. Die kurzfristige Kapazität wird für die Verwendung von Spot Instances oder einer Spot-Flotte konfiguriert. On-Demand-Instances werden nur für kurzfristige Workloads verwendet, die nicht unterbrochen werden können und nicht lange genug für reservierte Kapazitäten ausgeführt werden – typischerweise 25 bis 75 % des Zeitraums, je nach Ressourcentyp.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Niedrig

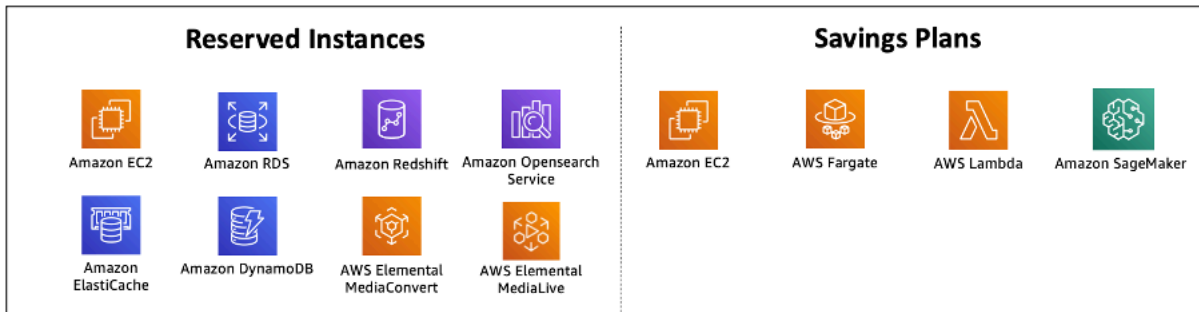
Implementierungsleitfaden

Um die Kosteneffizienz zu verbessern, bietet AWS mehrere Empfehlungen für Verpflichtungen auf Grundlage Ihrer bisherigen Nutzung an. Anhand dieser Empfehlungen können Sie nachvollziehen, was Sie einsparen können und wie die Verpflichtung verwendet wird. Sie können diese Services als On-Demand- oder Spot-Konfiguration nutzen oder sich für einen bestimmten Zeitraum verpflichten und Ihre On-Demand-Kosten mithilfe von Reserved Instances (RIs) und Savings Plans (SPs) reduzieren. Zur Optimierung Ihrer Workload müssen Sie nicht nur die einzelnen Workload-Komponenten und die verschiedenen AWS-Services berücksichtigen, sondern auch die Bindungsrabatte, Kaufoptionen und Spot Instances für diese Services.

Beachten Sie die Anforderungen der jeweiligen Workload-Komponenten sowie die verschiedenen Preismodelle für diese Services. Definieren Sie die Verfügbarkeitsanforderungen dieser Komponenten. Stellen Sie fest, ob mehrere unabhängige Ressourcen vorhanden sind, die die Funktion in der Workload ausführen, und welche Workload-Anforderungen im Laufe der Zeit gelten. Vergleichen Sie die Kosten der Ressourcen unter Verwendung des standardmäßigen On-Demand-Preismodells und anderer anwendbarer Modelle. Beziehen Sie potenzielle Änderungen in Ressourcen oder Workload-Komponenten in Ihre Überlegungen ein.

Sehen wir uns zum Beispiel diese Webanwendungsarchitektur in AWS an. Diese Beispiel-Workload besteht aus mehreren AWS-Services, wie z. B. Amazon Route 53, AWS WAF, Amazon CloudFront, Amazon-EC2-Instances, Amazon-RDS-Instances, Load Balancer, Amazon-S3-Speicher und Amazon

Elastic File System (Amazon EFS). Sie müssen jeden dieser Services überprüfen und mögliche Kosteneinsparungen durch die verschiedenen Preismodelle ermitteln. Einige von ihnen können für RIs oder SPs infrage kommen, während andere nur On-Demand verfügbar sind. Wie die folgende Abbildung zeigt, können einige der AWS-Services mithilfe von RIs oder SPs bereitgestellt werden.



AWS-Services, die über Reserved Instances und Savings Plans bereitgestellt werden

Implementierungsschritte

- Implementieren von Preismodellen: Kaufen Sie anhand Ihrer Analyseergebnisse Savings Plans, Reserved Instances oder implementieren Sie Spot Instances. Wenn Sie sich zum ersten Mal verpflichten, wählen Sie die 5 oder 10 besten Empfehlungen aus der Liste aus und beobachten und analysieren Sie die Ergebnisse in den nächsten ein bis zwei Monaten. AWS Cost Management Console begleitet Sie durch den Prozess. Überprüfen Sie die RI- oder SP-Empfehlungen von der Konsole aus, passen Sie die Empfehlungen an (Typ, Zahlung und Laufzeit) und überprüfen Sie die stündliche Verpflichtung (z. B. 20 USD pro Stunde) und legen Sie sie dann in den Warenkorb. Die Rabatte gelten automatisch für die berechnete Nutzung. Erwerben Sie in regelmäßigen Zyklen eine geringe Anzahl von Bindungsrabatten, (z. B. alle 2 Wochen oder monatlich). Implementieren Sie Spot Instances für Workloads, die unterbrochen werden können oder zustandslos sind. Wählen Sie anschließend On-Demand-Instances von Amazon EC2 aus und weisen Sie Ressourcen für die verbleibenden Anforderungen zu.
- Workload-Überprüfungszyklus: Implementieren Sie einen Überprüfungszyklus für die Workload, der speziell die Abdeckung des Preismodells analysiert. Sobald der Workload den erforderlichen Umfang erreicht hat, können Sie teilweise (alle paar Monate) oder wenn sich die Nutzung Ihrer Organisation ändert, zusätzliche Bindungsrabatte erwerben.

Ressourcen

Zugehörige Dokumente:

- [Understanding your Savings Plans recommendations](#)

- [Zugriff auf Empfehlungen für Reserved Instances](#)
- [Erwerb von Reserved Instances](#)
- [Instance-Kaufoptionen](#)
- [Spot Instances](#)
- [Reservierungsmodelle für andere AWS-Services](#)
- [Savings Plans Supported Services](#)

Zugehörige Videos:

- [Save up to 90% and run production workloads on Spot](#)

Zugehörige Beispiele:

- [Was sollte ich vor dem Kauf eines Savings Plan beachten?](#)
- [Wie kann ich den Cost Explorer verwenden, um meine Ausgaben und Nutzung zu analysieren?](#)

COST07-BP05 Durchführen einer Preismodellanalyse auf Verwaltungskontoebene

Prüfen Sie die Tools für die Fakturierung und Kostenverwaltung und informieren Sie sich über empfohlene Rabatte bei Bindung und Reservierungen, um regelmäßige Analysen auf Ebene des Verwaltungskontos auszuführen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Niedrig

Implementierungsleitfaden

Durch die regelmäßige Kostenmodellierung können Sie Möglichkeiten zur Optimierung über mehrere Workloads hinweg implementieren. Wenn beispielsweise mehrere Workloads On-Demand-Instances verwenden, ist das Änderungsrisiko insgesamt niedriger und die Nutzung eines auf fester Kapazität basierenden Rabatts kann zu niedrigeren Gesamtkosten führen. Es wird empfohlen, Analysen in regelmäßigen Zyklen von zwei Wochen bis zu einem Monat durchzuführen. Auf diese Weise können Sie kleine Anpassungskäufe tätigen, sodass sich die Abdeckung Ihrer Preismodelle mit Ihren sich ändernden Workloads und ihren Komponenten weiter entwickelt.

Verwenden Sie das Empfehlungstool [AWS Cost Explorer](#), um Möglichkeiten für an feste Kapazität gebundene Rabatte in Ihrem Verwaltungskonto zu finden. Empfehlungen auf Ebene

des Verwaltungskontos werden unter Berücksichtigung der Nutzung aller Konten in Ihrer AWS-Organisation berechnet, die über Reserved Instances (RI) oder Savings Plans (SP) verfügen. Sie werden auch berechnet, wenn die Rabattteilung aktiviert ist, um eine Festlegung zu empfehlen, mit der die Ersparnisse auf allen Konten maximiert werden.

Beim Kauf auf Verwaltungskontoebene werden zwar in vielen Fällen maximale Einsparungen erzielt, es kann jedoch Situationen geben, in denen Sie den Kauf von SPs auf der verknüpften Kontoebene in Betracht ziehen könnten, z. B. wenn Sie möchten, dass die Rabatte zuerst für die Nutzung in diesem bestimmten verknüpften Konto gelten. Empfehlungen für Mitgliedskonten werden auf Ebene der einzelnen Konten berechnet, um die Einsparungen für das jeweilige Konto zu maximieren. Wenn Ihr Konto sowohl RI- als auch SP-Bindungen umfasst, werden diese in der folgenden Reihenfolge angewendet:

1. Zonen-RI
2. Standard-RI
3. Konvertierbare RI
4. Instance Savings Plan
5. Compute Savings Plan

Wenn Sie einen SP auf Verwaltungskontoebene erwerben, werden die Einsparungen auf der Grundlage des höchsten bis niedrigsten Rabattprozentsatzes berechnet. SPs auf Verwaltungskontoebene überprüfen alle verknüpften Konten und wenden die Ersparnisse dort an, wo der Rabatt am höchsten ist. Wenn Sie einschränken möchten, wo die Ersparnisse verwendet werden, können Sie auf der verknüpften Kontoebene einen Savings Plan erwerben. Jedes Mal, wenn auf diesem Konto berechnete Computing-Services ausgeführt werden, wird der Rabatt zuerst dort angewendet. Wenn auf dem Konto keine berechtigten Datenverarbeitungsservices ausgeführt werden, wird der Rabatt auf die anderen verknüpften Konten unter demselben Verwaltungskonto aufgeteilt. Die gemeinsame Nutzung von Rabatten ist standardmäßig aktiviert, kann aber bei Bedarf deaktiviert werden.

In einer konsolidierten Abrechnungsfamilie werden Savings Plans zuerst auf die Nutzung des Inhaberkontos und dann auf die Nutzung anderer Konten angewendet. Dies ist nur dann der Fall, wenn Sie das Teilen aktiviert haben. Ihre Savings Plans werden zuerst auf Ihren höchsten Sparprozentsatz angewendet. Wenn es mehrere Nutzungen mit denselben Sparprozentsätzen gibt, werden Savings Plans auf die erste Nutzung mit der niedrigsten Savings-Plans-Rate angewendet. Savings Plans gelten so lange, bis keine Restnutzungen mehr zur Verfügung stehen oder Ihre Bindung ausgeschöpft ist. Jede verbleibende Nutzung wird zu den On-Demand-Tarifen abgerechnet.

Sie können die Empfehlungen zu Savings Plans (SP) in AWS Cost Management jederzeit aktualisieren, um neue Empfehlungen zu Savings Plans zu generieren.

Nach der Analyse der Flexibilität der Instances können Sie sich entsprechend den Empfehlungen festlegen. Erstellen Sie eine Kostenmodellierung, indem Sie die kurzfristigen Kosten der Workload mit möglichen verschiedenen Ressourcenoptionen analysieren und die AWS-Preismodelle analysieren und an Ihren geschäftlichen Anforderungen ausrichten, um die Gesamtbetriebskosten sowie die Möglichkeiten zur [Kostenoptimierung](#) zu ermitteln.

Implementierungsschritte

Durchführen einer Analyse des Bindungsrabatts: Sehen Sie sich unter Verwendung von Cost Explorer in Ihrem Konto die Empfehlungen für Savings Plans und Reserved Instances an. Stellen Sie sicher, dass Sie die Empfehlungen zu Savings Plans verstehen, und schätzen Sie Ihre monatlichen Ausgaben und Einsparungen. Sehen Sie sich die Empfehlungen auf Ebene des Verwaltungskontos an, die unter Berücksichtigung der Nutzung aller Mitgliedskonten in Ihrer AWS-Organisation berechnet werden, für die das Teilen der Rabatte für RI oder Savings Plans aktiviert ist, um maximale Einsparungen über alle Konten hinweg zu ermöglichen. Sie können sicherstellen, dass Sie die richtigen Empfehlungen mit den erforderlichen Rabatten und Risiken implementieren, indem Sie die Well-Architected Labs befolgen.

Ressourcen

Zugehörige Dokumente:

- [Wie funktioniert die Preisgestaltung in AWS?](#)
- [Instance-Kaufoptionen](#)
- [Übersicht über Savings Plans](#)
- [Empfehlungen zu Savings Plans](#)
- [Zugriff auf Empfehlungen für Reserved Instances](#)
- [Understanding your Saving Plans recommendation](#)
- [How Savings Plans apply to your AWS usage](#)
- [Savings Plans mit konsolidierter Fakturierung](#)
- [Aktivieren von geteilten reservierten Instances und Savings Plans-Rabatten](#)

Zugehörige Videos:

- [Save up to 90% and run production workloads on Spot](#)

Zugehörige Beispiele:

- [Was sollte ich vor dem Kauf eines Savings Plan beachten?](#)
- [How can I use rolling Savings Plans to reduce commitment risk?](#)
- [Verwendung von Spot-Instances](#)

KOSTEN 8. Wie können Sie die Kosten für Datenübertragungen planen?

Damit Sie architekturbezogene Entscheidungen zur Kostenminimierung treffen können, müssen Sie unbedingt die Datenübertragungskosten einplanen und überwachen. Eine geringfügige, aber effektive Änderung an der Architektur kann Ihre Betriebskosten über einen längeren Zeitraum hinweg erheblich senken.

Best Practices

- [COST08-BP01 Durchführen einer Datenübertragungsmodellierung](#)
- [COST08-BP02 Auswahl von Komponenten zur Optimierung der Datenübertragungskosten](#)
- [COST08-BP03 Implementieren von Services zur Senkung der Datenübertragungskosten](#)

COST08-BP01 Durchführen einer Datenübertragungsmodellierung

Stellen Sie die Organisationsanforderungen zusammen und führen Sie eine Datenübertragungsmodellierung der Workload und ihrer einzelnen Komponenten durch. Dadurch wird der niedrigste Kostenpunkt für die jeweiligen aktuellen Datenübertragungsanforderungen ermittelt.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Wenn eine Lösung in der Cloud entworfen wird, werden die Gebühren für die Datenübertragung in der Regel vernachlässigt, da die Architektur üblicherweise in On-Premises-Rechenzentren entworfen wird oder es an Kenntnissen mangelt. Die Gebühren für die Datenübertragung in AWS hängen von der Quelle, dem Ziel und der Datenverkehrsmenge ab. Die Berücksichtigung dieser Gebühren in der Entwicklungsphase kann zu Kosteneinsparungen führen. Für eine genaue Einschätzung der Gesamtbetriebskosten (TCO) ist es sehr wichtig zu verstehen, wo die Datenübertragung in Ihrer Workload stattfindet, wie hoch die Kosten der Übertragung sind und welcher Nutzen damit verbunden ist. Auf diese Weise können Sie eine fundierte Entscheidung treffen, die Architekturentscheidung

zu ändern oder zu akzeptieren. Sie können beispielsweise über eine Multi-Availability-Zone-Konfiguration verfügen, in der Sie Daten zwischen den Availability Zones replizieren.

Sie modellieren die Komponenten der Services, die die Daten in Ihrer Workload übertragen, und entscheiden, dass dies akzeptable Kosten sind (ähnlich wie bei der Zahlung für Datenverarbeitung und Speicher in beiden Availability Zones), um die erforderliche Zuverlässigkeit und Ausfallsicherheit zu erreichen. Modellieren Sie die Kosten über verschiedene Nutzungsstufen. Die Workload-Nutzung kann sich im Laufe der Zeit ändern und verschiedene Services können auf verschiedenen Ebenen kostengünstiger sein.

Denken Sie bei der Modellierung Ihrer Datenübertragung daran, wie viele Daten aufgenommen werden und woher diese Daten stammen. Berücksichtigen Sie außerdem, wie viele Daten verarbeitet werden und wie viel Speicher- oder Datenverarbeitungskapazität benötigt wird. Befolgen Sie bei der Modellierung bewährte Netzwerkmethoden für Ihre Workload-Architektur, um Ihre potenziellen Datenübertragungskosten zu optimieren.

AWS Pricing Calculator kann Ihnen helfen, die geschätzten Kosten für bestimmte AWS-Services und den erwarteten Datentransfer einzuschätzen. Wird bereits eine Workload ausgeführt (zu Testzwecken oder in einer Vorproduktionsumgebung), verwenden Sie [AWS Cost Explorer](#) oder [AWS Cost and Usage Report](#) (CUR), um Ihre Datenübertragungskosten zu verstehen und zu modellieren. Konfigurieren Sie einen Machbarkeitsnachweis (PoC) oder testen Sie Ihre Workload und führen Sie einen Test mit einer realistischen simulierten Last aus. Sie können Ihre Kosten bei verschiedenen Workload-Nachfragen modellieren.

Implementierungsschritte

- Ermitteln der Anforderungen: Was ist das primäre Ziel und was sind die geschäftlichen Anforderungen für die geplante Datenübertragung zwischen Quelle und Ziel? Was ist das erwartete Geschäftsergebnis am Ende? Ermitteln Sie die Geschäftsanforderungen und definieren Sie das erwartete Ergebnis.
- Ermitteln von Quelle und Ziel: Was ist die Datenquelle und das Ziel für die Datenübertragung, z. B. innerhalb der AWS-Regionen, in AWS Services oder ins Internet?
 - [Datenübertragung innerhalb einer AWS-Region](#)
 - [Datenübertragung zwischen AWS-Regionen](#)
 - [Datenübertragung ins Internet](#)
- Ermittlung der Datenklassifizierungen: Welche Datenklassifizierung gilt für diese Datenübertragung? Um welche Art von Daten handelt es sich? Um wie viele Daten handelt es sich? Wie häufig müssen die Daten übertragen werden? Handelt es sich um sensible Daten?

- Ermitteln der zu verwendenden AWS-Services oder -Tools: Welche AWS-Services werden für diese Datenübertragung verwendet? Ist es möglich, einen bereits bereitgestellten Service für eine andere Workload zu verwenden?
- Berechnen der Datenübertragungskosten: Verwenden Sie [AWS Pricing](#), die Datenübertragungsmodellierung, die Sie zuvor erstellt haben, um die Datenübertragungskosten für die Workload zu berechnen. Berechnen Sie die Datenübertragungskosten bei verschiedenen Nutzungsstufen, sowohl bei erhöhter als auch bei verringerter Workload-Nutzung. Wenn es mehrere Optionen für die Workload-Architektur gibt, berechnen Sie die Kosten für jede Option zum Vergleich.
- Verknüpfen von Kosten mit Ergebnissen: Geben Sie für alle anfallenden Datenübertragungskosten das Ergebnis an, das damit für die Workload erreicht wird. Erfolgt der Transfer zwischen Komponenten, kann dies für die Entkopplung verwendet werden. Erfolgt der Transfer zwischen Availability Zones, kann dies zur Redundanz verwendet werden.
- Erstellen der Datenübertragungsmodellierung: Nachdem Sie alle Informationen zusammengetragen haben, erstellen Sie eine konzeptionelle Basisdatenübertragungsmodellierung für mehrere Anwendungsfälle und unterschiedliche Workloads.

Ressourcen

Zugehörige Dokumente:

- [AWS-Caching-Lösungen](#)
- [AWS Preise](#)
- [Amazon EC2 – Preise](#)
- [Amazon VPC – Preise](#)
- [Understanding data transfer charges](#)

Zugehörige Videos:

- [Monitoring and Optimizing Your Data Transfer Costs](#)
- [S3 Transfer Acceleration](#)

Zugehörige Beispiele:

- [Überblick über die Datenübertragungskosten für gängige Architekturen](#)

- [AWS Prescriptive Guidance for Networking](#)

COST08-BP02 Auswahl von Komponenten zur Optimierung der Datenübertragungskosten

Alle Komponenten sind ausgewählt und die Architektur ist so konzipiert, dass die Datenübertragungskosten gesenkt werden. Dies umfasst auch die Verwendung von Komponenten wie Wide Area Network (WAN)-Optimierung und Multi-Availability-Zone-Konfigurationen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Eine Architektur für die Datenübertragung minimiert die Kosten für die Datenübertragung. Dies kann auch die Nutzung von CDNs bedeuten, um die Daten näher an den Benutzern zu platzieren, oder die Verwendung spezieller Netzwerklinks von Ihrem Standort zu AWS. Sie können auch WAN-Optimierung und Anwendungsoptimierung verwenden, um die Datenmenge zu reduzieren, die zwischen Komponenten übertragen wird.

Bei der Übertragung von Daten zu oder innerhalb von AWS Cloud ist es wichtig, das Ziel auf der Grundlage der verschiedenen Anwendungsfälle, der Art der Daten und der verfügbaren Netzwerkressourcen zu kennen, um die richtigen AWS-Services zur Optimierung der Datenübertragung auszuwählen. AWS bietet eine Reihe von Datenübertragungsservices, die auf die verschiedenen Anforderungen der Datenmigration zugeschnitten sind. Wählen Sie die richtigen Optionen für die [Datenspeicherung](#) und [Datenübertragung](#) auf Grundlage der geschäftlichen Anforderungen in Ihrer Organisation.

Beachten Sie bei der Planung oder Überprüfung Ihrer Workload-Architektur die folgenden Punkte:

- Verwenden von VPC-Endpunkten in AWS: VPC-Endpunkte ermöglichen private Verbindungen zwischen Ihrer VPC und unterstützten AWS-Services. So vermeiden Sie die Nutzung des öffentlichen Internets, was zu Kosten für die Datenübertragung führen kann.
- Verwenden eines NAT-Gateways: Verwenden Sie ein [NAT-Gateway](#), damit Instances in einem privaten Subnetz eine Verbindung zum Internet oder zu den Services außerhalb Ihrer VPC herstellen können. Überprüfen Sie, ob sich die Ressourcen hinter dem NAT-Gateway, die den meisten Datenverkehr senden, in derselben Availability Zone befinden wie das NAT-Gateway. Wenn dies nicht der Fall ist, erstellen Sie neue NAT-Gateways in derselben Availability Zone wie die Ressource, um die Gebühren für die Datenübertragung zwischen den Zonen zu reduzieren.

- Verwenden von AWS Direct Connect: Direct Connect umgeht das öffentliche Internet und stellt eine direkte, private Verbindung zwischen Ihrem On-Premises-Netzwerk und AWS her. Dies kann kostengünstiger und konsistenter sein als die Übertragung großer Datenmengen über das Internet.
- Vermeidung von Datenübertragungen über regionale Grenzen hinweg: Bei Datenübertragungen zwischen AWS-Regionen (von einer Region in eine andere) fallen in der Regel Gebühren an. Die Entscheidung, einen multiregionalen Weg einzuschlagen, sollte gut überlegt sein. Weitere Informationen finden Sie unter [Szenarien mit mehreren Regionen](#).
- Überwachen der Datenübertragung: Verwenden Sie Amazon CloudWatch und [VPC-Flow-Protokolle](#), um Einzelheiten über Ihre Datenübertragung und Netzwerknutzung zu erfassen. Analysieren Sie den erfassten Netzwerkverkehr in Ihren VPCs, z. B. die IP-Adresse oder den Bereich, der von und zu Netzwerkschnittstellen geht.
- Analysieren Ihrer Netzwerknutzung: Verwenden Sie Mess- und Berichtstools wie AWS Cost Explorer, CUDOS-Dashboards oder CloudWatch, um die Datenübertragungskosten Ihrer Workload zu verstehen.

Implementierungsschritte

- Auswählen der Komponenten für die Datenübertragung: Konzentrieren Sie sich anhand der in [COST08-BP01 Durchführen einer Datenübertragungsmodellierung](#) erläuterten Datenübertragungsmodellierung darauf, wo die größten Datenübertragungskosten anfallen oder anfallen würden, wenn sich die Workload-Nutzung ändert. Suchen Sie nach alternativen Architekturen oder zusätzlichen Komponenten, die den Datenübertragungsbedarf beseitigen oder reduzieren (oder die Kosten senken).

Ressourcen

Zugehörige bewährte Methoden:

- [COST08-BP01 Durchführen einer Datenübertragungsmodellierung](#)
- [COST08-BP03 Implementieren von Services zur Senkung der Datenübertragungskosten](#)

Zugehörige Dokumente:

- [Cloud-Datenmigration](#)
- [AWS-Caching-Lösungen](#)
- [Schnellere Bereitstellung von Inhalten mit Amazon CloudFront](#)

Zugehörige Beispiele:

- [Überblick über die Datenübertragungskosten für gängige Architekturen](#)
- [AWS Network Optimization Tips](#)
- [Die Leistung optimieren und die Kosten für Netzwerkanalytik mit VPC-Ablaufprotokollen im Apache-Parquet-Format senken](#)

COST08-BP03 Implementieren von Services zur Senkung der Datenübertragungskosten

Implementieren Sie Services zur Verringerung der Datenübertragung. Verwenden Sie beispielsweise Edge-Standorte oder Content Delivery Networks (CDN), um Inhalte für Endbenutzer bereitzustellen, erstellen Sie Caching-Ebenen vor Ihren Anwendungsservern oder Datenbanken und verwenden Sie dedizierte Netzwerkverbindungen anstelle von VPNs für die Konnektivität zur Cloud.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Es gibt verschiedene AWS-Services, mit denen Sie die Nutzung Ihrer Netzwerkdatenübertragung optimieren können. Abhängig von den Komponenten, dem Typ und der Cloud-Architektur Ihrer Workload können diese Services Sie bei der Komprimierung, beim Caching sowie der gemeinsamen Nutzung und Verteilung Ihres Datenverkehrs in der Cloud unterstützen.

- [Amazon CloudFront](#) ist ein weltweites Inhaltsbereitstellungnetzwerk, das Daten bei niedriger Latenz und hohen Datenübertragungsgeschwindigkeiten bereitstellt. Es stellt Daten an Edge-Standorten rund um die Welt in den Cache und reduziert damit die Belastung Ihrer Ressourcen. Durch die Verwendung von CloudFront können Sie den administrativen Aufwand für die Bereitstellung von Inhalten für eine große Anzahl an Benutzern weltweit bei minimaler Latenz reduzieren. Das [Paket mit Sicherheitseinsparungen](#) kann Ihnen dabei helfen, bis zu 30 % Ihrer CloudFront-Nutzung einzusparen, wenn Sie planen, Ihre Nutzung im Laufe der Zeit zu erhöhen.
- [AWS Direct Connect](#) ermöglicht es Ihnen, eine dedizierte Netzwerkverbindung zu AWS aufzubauen. Damit können Sie Netzwerkkosten reduzieren, die Bandbreite erhöhen und eine im Vergleich zu Internet-basierten Verbindungen gleichbleibendere Netzwerkerfahrung bieten.
- [Site-to-Site VPN](#) ermöglicht es Ihnen, eine sichere und private Verbindung zwischen Ihrem privaten Netzwerk und dem globalen AWS-Netzwerk aufzubauen. Dies ist ideal für kleine Niederlassungen oder Geschäftspartner, da es vereinfachte Konnektivität bietet und ein vollständig verwalteter und elastischer Service ist.

- [VPC-Endpunkte](#) ermöglichen Konnektivität zwischen AWS-Services über private Netzwerke und können verwendet werden, um Kosten für öffentliche Datenübertragungen und [NAT-Gateways](#) zu reduzieren. Für [Gateway-VPC-Endpunkte](#) fallen keine stündlichen Gebühren an und sie unterstützen Amazon S3 und Amazon DynamoDB. [Schnittstellen-VPC-Endpunkte](#) werden von [AWS PrivateLink](#) bereitgestellt und für sie fällt eine Gebühr pro Stunde und Nutzungskosten pro GB an.
- [NAT-Gateways](#) bieten integrierte Skalierung und Verwaltung, wodurch die Kosten im Vergleich zu einer eigenständigen NAT-Instance reduziert werden. Platzieren Sie NAT-Gateways in denselben Availability Zones wie Instances mit hohem Datenverkehr und ziehen Sie die Verwendung von VPC-Endpunkten für die Instances in Betracht, die auf Amazon DynamoDB oder Amazon S3 zugreifen müssen, um die Datenübertragungs- und Verarbeitungskosten zu senken.
- Verwenden Sie [AWS Snow Family](#)-Geräte, die über Datenverarbeitungsressourcen zum Erfassen und Verarbeiten von Daten am Netzwerk-Edge verfügen. Mit AWS Snow Family-Geräten ([Snowball Edge](#), [Snowball Edge](#) und [Snowmobile](#)) können Sie Petabytes an Daten kostengünstig und offline in die AWS Cloud verschieben.

Implementierungsschritte

- Services implementieren: Wählen Sie für Ihren Service und Workload-Typ anhand der Datenübertragungsmodellierung und der Informationen in den VPC-Flow-Protokollen die entsprechenden AWS-Netzwerkservices aus. Sehen Sie sich an, wo sich die höchsten Kosten und Volumenströme befinden. Überprüfen Sie die AWS-Services und ermitteln Sie, ob es einen Service gibt, der die Übertragung reduziert oder entfernt, insbesondere die Netzwerk- und Inhaltsbereitstellung. Suchen Sie auch nach Caching-Services, bei denen wiederholt auf Daten oder große Datenmengen zugegriffen wird.

Ressourcen

Zugehörige Dokumente:

- [AWS Direct Connect](#)
- [Unsere AWS-Produkte entdecken](#)
- [AWS-Caching-Lösungen](#)
- [Amazon CloudFront](#)
- [AWS Snow Family](#)
- [Amazon CloudFront-Sicherheitssparpaket](#)

Zugehörige Videos:

- [Monitoring and Optimizing Your Data Transfer Costs](#)
- [AWS Cost Optimization Series: CloudFront](#)
- [Wie reduziere ich die Datenübertragungsgebühren für mein NAT-Gateway in Amazon VPC?](#)

Zugehörige Beispiele:

- [How-to chargeback shared services: An AWS Transit Gateway example](#)
- [Understand AWS data transfer details in depth from cost and usage report using Athena query and QuickSight](#)
- [Überblick über die Datenübertragungskosten für gängige Architekturen](#)
- [Using AWS Cost Explorer to analyze data transfer costs](#)
- [Cost-Optimizing your AWS architectures by utilizing Amazon CloudFront features](#)
- [Wie reduziere ich die Datenübertragungsgebühren für mein NAT-Gateway in Amazon VPC?](#)

Verwaltung von Nachfrage und Bereitstellung von Ressourcen

Frage

- [KOSTEN 9. Wie verwalten Sie die Nachfrage und stellen Ressourcen bereit?](#)

KOSTEN 9. Wie verwalten Sie die Nachfrage und stellen Ressourcen bereit?

Stellen Sie bei einer Workload mit ausgewogenen Ausgaben und Leistungen sicher, dass alles, wofür Sie bezahlen, genutzt wird, und vermeiden Sie eine erhebliche Unterauslastung der Instances. Eine verschobene Auslastungsmetrik in einer der Richtungen wirkt sich nachteilig auf Ihr Unternehmen aus, entweder im Hinblick auf die Betriebskosten (verschlechterte Leistung aufgrund von Überbelegung) oder auf die verschwendeten AWS-Ausgaben (aufgrund von Überversorgung).

Best Practices

- [COST09-BP01 Analyse des Workload-Bedarfs](#)
- [COST09-BP02 Implementieren eines Puffers oder einer Drosselung zur Bedarfsverwaltung](#)
- [COST09-BP03 Dynamische Bereitstellung von Ressourcen](#)

COST09-BP01 Analyse des Workload-Bedarfs

Analysieren Sie den Bedarf des Workloads im gesamten Zeitverlauf. Stellen Sie sicher, dass die Analyse saisonale Trends berücksichtigt und die Betriebsbedingungen über die gesamte Lebensdauer des Workloads genau wiedergibt. Der Analyseaufwand sollte in einem angemessenen Verhältnis zum potenziellen Nutzen stehen, z. B. muss der Zeitaufwand den Workload-Kosten entsprechen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Hoch

Implementierungsleitfaden

Die Analyse des Workload-Bedarfs für Cloud-Computing erfordert ein Verständnis der Muster und Eigenschaften von Computing-Aufgaben, die in der Cloud-Umgebung initiiert werden. Diese Analyse hilft Benutzern, die Ressourcenzuweisung zu optimieren, Kosten zu verwalten und sicherzustellen, dass die Leistung den Anforderungen entspricht.

Informieren Sie sich über die Anforderungen des Workloads. Die Anforderungen Ihrer Organisation sollten die Workload-Reaktionszeiten für Anforderungen angeben. Anhand der Reaktionszeit kann bestimmt werden, ob der Bedarf gut gesteuert wird oder ob das Ressourcenangebot geändert werden sollte, um der Nachfrage gerecht zu werden.

Die Analyse sollte die Vorhersehbarkeit und Wiederholbarkeit der Nachfrage, die Änderungsrate der Nachfrage und die Menge der Nachfrageänderungen umfassen. Die Analyse muss über einen ausreichend langen Zeitraum ausgeführt werden, um saisonale Abweichungen wie Arbeiten am Ende des Monats oder Feiertagsspitzen einzubeziehen.

Der Analyseaufwand sollte die potenziellen Vorteile der Implementierung einer Skalierung widerspiegeln. Betrachten Sie die erwarteten Gesamtkosten der Komponente sowie etwaige Erhöhungen oder Verringerungen der Nutzung und der Kosten während der Workload-Lebensdauer.

Im Folgenden sind einige wichtige Aspekte aufgeführt, die bei der Durchführung der Workload-Bedarfsanalyse für Cloud-Computing zu berücksichtigen sind:

1. **Ressourcennutzung und Metriken:** Analysieren Sie, wie AWS-Ressourcen im Laufe der Zeit genutzt werden. Ermitteln Sie Nutzungsmuster während und außerhalb der Spitzenzeiten, um die Ressourcenzuweisung und Skalierungsstrategien zu optimieren. Überwachen Sie Leistungskennzahlen wie Reaktionszeiten, Latenz, Durchsatz und Fehlerraten. Diese Kennzahlen helfen bei der Bewertung des Gesamtzustands und der Effizienz der Cloud-Infrastruktur.

2. Skalierungsverhalten von Benutzern und Anwendungen: Verstehen Sie das Benutzerverhalten und wie es sich auf den Workload-Bedarf auswirkt. Das Untersuchen von Mustern beim Benutzerdatenverkehr trägt dazu bei, die Bereitstellung von Inhalten und die Reaktionsfähigkeit von Anwendungen zu verbessern. Analysieren Sie, wie Workloads mit steigender Nachfrage skaliert werden. Bestimmen Sie, ob die Parameter für die automatische Skalierung korrekt und effektiv für den Umgang mit Auslastungsschwankungen konfiguriert sind.
3. Workload-Typen: Identifizieren Sie die verschiedenen Typen von Workloads, die in der Cloud ausgeführt werden, wie Batch-Verarbeitung, Echtzeitdatenverarbeitung, Webanwendungen, Datenbanken oder Machine Learning. Jeder Workload-Typ kann unterschiedliche Ressourcenanforderungen und Leistungsprofile aufweisen.
4. Service Level Agreements (SLAs): Vergleichen Sie die tatsächliche Leistung mit den SLAs, um deren Einhaltung sicherzustellen und Bereiche zu identifizieren, die verbessert werden müssen.

Sie können [Amazon CloudWatch](#) verwenden, um Metriken zu erfassen und nachzuverfolgen, Protokolldateien zu sammeln und zu überwachen, Alarme festzulegen und automatisch auf Änderungen Ihrer AWS-Ressourcen zu reagieren. Amazon CloudWatch bietet Ihnen auch einen systemweiten Einblick in die Auslastung Ihrer Ressourcen, die Anwendungsleistung und die Integrität Ihrer Betriebsabläufe.

Mit [AWS Trusted Advisor](#) können Sie Ihre Ressourcen gemäß bewährten Methoden bereitstellen und so die Systemleistung und -zuverlässigkeit verbessern, die Sicherheit erhöhen und nach Einsparungsmöglichkeiten suchen. Darüber hinaus können Sie Nicht-Produktions-Instances deaktivieren und Amazon CloudWatch und Auto Scaling verwenden, um auf Steigerungen oder Reduzierungen des Bedarfs zu reagieren.

Schließlich können Sie [AWS Cost Explorer](#) oder [Quick](#) mit der AWS Cost and Usage Report (CUR)-Datei oder Ihren Anwendungsprotokollen verwenden, um eine erweiterte Analyse des Workload-Bedarfs durchzuführen.

Insgesamt ermöglicht eine umfassende Analyse des Workload-Bedarfs Unternehmen fundierte Entscheidungen zur Bereitstellung, Skalierung und Optimierung von Ressourcen, was zu einer höheren Leistung, Kosteneffizienz und Benutzerzufriedenheit führt.

Implementierungsschritte

- Vorhandene Workload-Daten analysieren: Analysieren Sie Daten aus der vorhandenen Workload, früheren Versionen der Workload oder vorhergesagten Nutzungsmustern. Verwenden Sie Amazon CloudWatch, Protokolldateien und Überwachungsdaten, um einen Einblick in die Nutzung der

Workload zu erhalten. Analysieren Sie einen vollständigen Workload-Zyklus und erfassen Sie Daten für alle saisonalen Änderungen, z. B. Ereignisse am Monatsende oder am Ende des Jahres. Der in der Analyse reflektierte Aufwand sollte die Workload-Merkmale widerspiegeln. Der größte Aufwand sollte für hochwertige Workloads mit den größten Nachfrageänderungen betrieben werden. Der geringste Aufwand sollte für Workloads mit niedrigem Wert und geringfügigen Nachfrageänderungen betrieben werden.

- Vorhersage externer Einflüsse: Treffen Sie Teammitglieder aus der gesamten Organisation, die die Nachfrage in der Workload beeinflussen oder ändern können. Häufig betroffene Teams sind Vertrieb, Marketing oder Business Development. Arbeiten Sie mit ihnen zusammen, um die Zyklen kennenzulernen, in denen sie arbeiten, und um zu erfahren, ob es Ereignisse gibt, die die Nachfrage des Workloads ändern könnten. Erstellen Sie eine Prognose des Workload-Bedarfs anhand dieser Daten.

Ressourcen

Zugehörige Dokumente:

- [Amazon CloudWatch](#)
- [AWS Trusted Advisor](#)
- [AWS X-Ray](#)
- [AWS Auto Scaling](#)
- [AWS Instance Scheduler](#)
- [Erste Schritte mit Amazon SQS](#)
- [AWS Cost Explorer](#)
- [Schnell](#)

Zugehörige Beispiele:

- [Monitor, Track and Analyze for cost optimization](#)
- [Searching and analyzing logs in CloudWatch](#)

COST09-BP02 Implementieren eines Puffers oder einer Drosselung zur Bedarfsverwaltung

Pufferung und Drosselung ändern den Bedarf Ihres Workloads und glätten alle Spitzen.

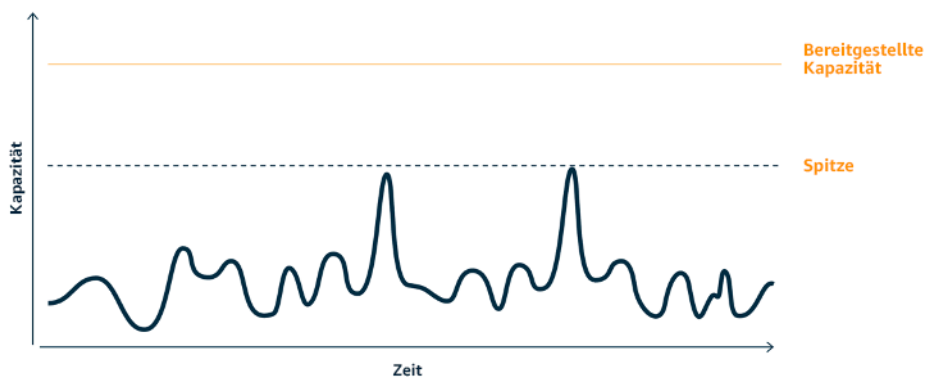
Implementieren Sie die Drosselung, wenn Ihre Clients Wiederholungen durchführen. Implementieren

Sie die Pufferung, um die Anforderung zu speichern und die Verarbeitung auf einen späteren Zeitpunkt zu verschieben. Stellen Sie sicher, dass Ihre Drosselungen und Puffer so konzipiert sind, dass Clients in der erforderlichen Zeit eine Antwort erhalten.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Die Implementierung einer Pufferung oder Drosselung ist beim Cloud-Computing von entscheidender Bedeutung, um die Nachfrage zu steuern und die für den Workload benötigte bereitgestellte Kapazität zu reduzieren. Für eine optimale Leistung ist es unerlässlich, die Gesamtnachfrage, einschließlich der Spitzen, sowie die Geschwindigkeit, mit der sich die Anfragen ändern, und die erforderliche Reaktionszeit zu messen. Wenn Clients die Möglichkeit haben, ihre Anfragen erneut zu senden, ist es praktisch, eine Drosselung vorzunehmen. Umgekehrt ist für Clients ohne Wiederholungsfunktionen die Implementierung einer Pufferlösung der ideale Ansatz. Solche Puffer rationalisieren den Eingang von Anfragen und optimieren die Interaktion von Anwendungen mit unterschiedlichen Betriebsgeschwindigkeiten.



Bedarfskurve mit zwei deutlichen Spitzen, die hohe bereitgestellte Kapazität erfordern

Nehmen wir einen Workload mit der nachfolgend gezeigten Bedarfskurve. Dieser Workload hat zwei Spitzen und um damit umzugehen, wird die Ressourcenkapazität bereitgestellt, die hier durch die orangefarbene Linie angezeigt wird. Die für diesen Workload aufgewendeten Ressourcen und die eingesetzte Energie werden nicht durch die Fläche unter der Bedarfskurve, sondern von der Linie für die bereitgestellte Kapazität angezeigt, da die bereitgestellte Kapazität zur Bewältigung dieser beiden Spitzen benötigt wird. Die Verflachung der Bedarfskurve kann Ihnen dabei helfen, die bereitgestellte Kapazität für einen Workload zu verringern und dessen Umweltauswirkungen zu reduzieren. Um die Spitzen abzuflachen, sollten Sie eine Lösung zur Drosselung oder Pufferung in Betracht ziehen.

Um dies besser zu verstehen, werden wir uns kurz die Drosselung und Pufferung ansehen.

Drosselung: Wenn die Quelle der Nachfrage über eine Wiederholungsfunktion verfügt, können Sie die Drosselung implementieren. Die Drosselung teilt der Quelle mit, dass wenn sie die Anfrage zum aktuellen Zeitpunkt nicht bedienen kann, sie es später erneut versuchen sollte. Die Quelle wartet einen bestimmten Zeitraum und wiederholt die Anfrage. Die Implementierung der Drosselung hat den Vorteil, dass die maximale Menge an Ressourcen und Kosten des Workloads begrenzt wird. In AWS können Sie [Amazon API Gateway](#) verwenden, um Drosselung zu implementieren.

Pufferbasiert: Ein pufferbasierter Ansatz verwendet Produzenten (Komponenten, die Nachrichten an die Warteschlange senden), Verbraucher (Komponenten, die Nachrichten aus der Warteschlange empfangen) und eine Warteschlange (die Nachrichten enthält), um die Nachrichten zu speichern. Nachrichten können dadurch von Verbrauchern in der für ihre Geschäftsanforderungen passenden Geschwindigkeit gelesen und verarbeitet werden. Durch die Verwendung einer pufferbasierten Methodik werden die Nachrichten von den Produzenten in Warteschlangen oder Streams gespeichert und können von den Verbrauchern in einem Tempo abgerufen werden, das sich an deren betrieblichen Anforderungen orientiert.

In AWS können Sie zur Implementierung eines pufferbasierten Ansatzes aus mehreren Services wählen. [Amazon Simple Queue Service \(Amazon SQS\)](#) ist ein verwalteter Service, der Warteschlangen bietet, die es einem einzelnen Verbraucher ermöglichen, individuelle Nachrichten zu lesen. [Amazon Kinesis](#) stellt einen Stream bereit, mit dem viele Verbraucher dieselben Nachrichten lesen können.

Durch Pufferung und Drosselung können Spitzenwerte abgeflacht werden, indem die Anforderungen an Ihren Workload angepasst werden. Verwenden Sie die Drosselung, wenn Clients Aktionen wiederholen, und nutzen Sie die Pufferung, um Anfragen zurückzuhalten und später zu verarbeiten. Stellen Sie bei der Architektur mit einem pufferbasierten Ansatz sicher, dass Sie Ihren Workload so gestalten, dass er die Anfrage in der erforderlichen Zeit erfüllt, und dass Sie doppelte Arbeitsanfragen verarbeiten können. Analysieren Sie den Gesamtbedarf, die Änderungsrate und die erforderliche Reaktionszeit, um die korrekte Größe der erforderlichen Drosselung oder des Puffers zu bestimmen.

Implementierungsschritte

- **Analysieren der Client-Anforderungen:** Analysieren Sie die Client-Anforderungen, um zu bestimmen, ob sie Wiederholungen durchführen können. Für Clients, die keine Wiederholungen durchführen können, müssen Puffer implementiert werden. Analysieren Sie den Gesamtbedarf, die Änderungsrate und die erforderliche Reaktionszeit, um die Größe der erforderlichen Drosselung oder des Puffers zu bestimmen.

- Implementieren eines Puffers oder einer Drosselung: Implementieren Sie einen Puffer oder eine Drosselung in der Workload. Eine Warteschlange wie Amazon Simple Queue Service (Amazon SQS) kann für Ihre Workload-Komponenten einen Puffer bereitstellen. Amazon API Gateway kann eine Drosselung für Ihre Workload-Komponenten bereitstellen.

Ressourcen

Zugehörige bewährte Methoden:

- [SUS02-BP06 Implementierung von Pufferung oder Drosselung, um die Bedarfskurve zu verflachen](#)
- [REL05-BP02 Drosselung von Anfragen](#)

Zugehörige Dokumente:

- [AWS Auto Scaling](#)
- [AWS Instance Scheduler](#)
- [Amazon API Gateway](#)
- [Amazon Simple Queue Service](#)
- [Erste Schritte mit Amazon SQS](#)
- [Amazon Kinesis](#)

Zugehörige Videos:

- [Choosing the Right Messaging Service for Your Distributed App](#)

Zugehörige Beispiele:

- [Managing and monitoring API throttling in your workloads](#)
- [Throttling a tiered, multi-tenant REST API at scale using API Gateway](#)
- [Enabling Tiering and Throttling in a Multi-Tenant Amazon EKS SaaS Solution Using Amazon API Gateway](#)
- [Application integration Using Queues and Messages](#)

COST09-BP03 Dynamische Bereitstellung von Ressourcen

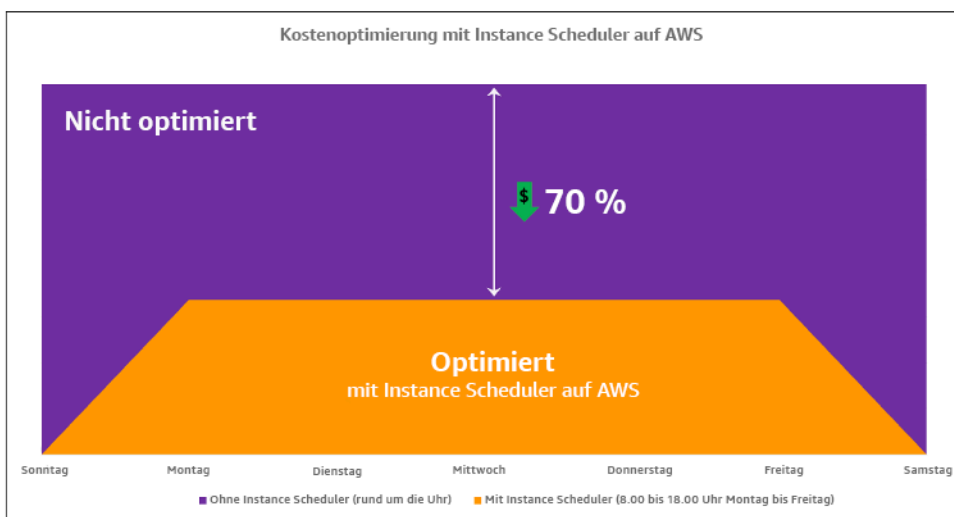
Ressourcen werden geplant bereitgestellt. Dies kann bedarfsbasiert sein, z. B. durch automatische Skalierung, oder zeitbasiert, wobei der Bedarf vorhersehbar ist und Ressourcen basierend auf der Zeit bereitgestellt werden. Diese Methoden führen zur geringsten Über- oder Unterversorgung.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Niedrig

Implementierungsleitfaden

Es gibt verschiedene Möglichkeiten für AWS-Kunden, die für ihre Anwendungen verfügbaren Ressourcen zu erhöhen und Ressourcen bereitzustellen, um der Nachfrage gerecht zu werden. Eine dieser Optionen ist die Verwendung von AWS Instance Scheduler, der das Starten und Stoppen von Amazon Elastic Compute Cloud (Amazon EC2)- und Amazon Relational Database Service (Amazon RDS)-Instances automatisiert. Die andere Option ist die Verwendung von AWS Auto Scaling, mit der Sie Ihre Computing-Ressourcen automatisch an die Anforderungen Ihrer Anwendung oder Ihres Services anpassen können. Wenn Sie Ressourcen bedarfsgerecht bereitstellen, zahlen Sie nur für die Ressourcen, die Sie tatsächlich nutzen. So senken Sie die Kosten, indem Sie Ressourcen bereitstellen, wenn sie benötigt werden, und sie beenden, wenn sie nicht mehr benötigt werden.

Mit [AWS Instance Scheduler](#) können Sie das Stoppen und Startn Ihrer Amazon-EC2- und Amazon-RDS-Instances zu definierten Zeiten konfigurieren. So können Sie die Nachfrage nach denselben Ressourcen innerhalb eines konsistenten Zeitmusters befriedigen. Beispiel: Ein Benutzer greift jeden Tag um acht Uhr morgens auf Amazon-EC2-Instances zu, die er nach sechs Uhr abends nicht mehr benötigt. Durch diese Lösung lassen sich die Betriebskosten senken, indem sie nicht genutzte Ressourcen stoppt und sie bei Bedarf wieder startet.



Kostenoptimierung mit AWS Instance Scheduler

Mit AWS Systems Manager Quick Setup können Sie mithilfe einer einfachen Benutzeroberfläche auch ganz einfach Zeitpläne für Ihre Amazon-EC2-Instances in Ihren Konten und Regionen konfigurieren. Sie können Amazon-EC2- oder Amazon-RDS-Instances mit AWS Instance Scheduler planen und bestehende Instances stoppen und starten. Sie können jedoch keine Instances stoppen und starten, die Teil Ihrer Auto-Scaling-Gruppe (ASG) sind oder die Services wie Amazon Redshift oder Amazon OpenSearch Service verwalten. Auto-Scaling-Gruppen haben ihre eigene Planung für die Instances in der Gruppe und diese Instances werden erstellt.

Mit [AWS Auto Scaling](#) können Sie Ihre Kapazität anpassen, um eine stabile, vorhersehbare Leistung zu möglichst niedrigen Kosten aufrechtzuerhalten. Es handelt sich um einen vollständig verwalteten und kostenlosen Service zur Skalierung Ihrer Anwendung, der sich in Amazon-EC2-Instances und Spot-Flotten, Amazon ECS, Amazon DynamoDB und Amazon Aurora integrieren lässt. Auto Scaling bietet eine automatische Ressourcenerkennung, um zu helfen, Ressourcen in Ihrer Workload zu finden, die konfiguriert werden können. Es verfügt über integrierte Skalierungsstrategien zur Optimierung der Leistung, der Kosten oder eines Gleichgewichts zwischen beiden Ressourcen und bietet eine prädiktive Skalierung, um regelmäßig auftretende Spitzen zu unterstützen.

Für die Skalierung Ihrer Auto-Scaling-Gruppe haben Sie mehrere Skalierungsoptionen:

- Fortlaufende Nutzung derselben Anzahl an Instances
- Manuelles Skalieren
- Skalierung nach Zeitplan
- Skalierung nach Bedarf
- Verwendung prädiktiver Skalierung

Auto-Scaling-Richtlinien unterscheiden sich und können in dynamische und geplante Skalierungsrichtlinien unterteilt werden. Dynamische Richtlinien sind für manuelle oder dynamische Skalierung, die geplant oder prädiktiv sein kann. Sie können Skalierungsrichtlinien für dynamische, geplante und prädiktive Skalierung verwenden. Sie können auch Metriken und Alarme von [Amazon CloudWatch](#) verwenden, um Skalierungsereignisse für Ihre Workload auszulösen. Wir empfehlen Ihnen die Verwendung von [Startvorlagen](#), mit denen Sie auf die neuesten Features und Verbesserungen zugreifen können. Nicht alle Auto-Scaling-Features sind verfügbar, wenn Sie Startkonfigurationen verwenden. Sie können beispielsweise keine Auto-Scaling-Gruppe erstellen, die Spot- und On-Demand-Instances startet oder mehrere Instance-Typen angibt. Sie müssen eine Startvorlage verwenden, um diese Funktionen zu konfigurieren. Wenn Sie Startvorlagen verwenden, empfehlen wir Ihnen, jede einzelne davon zu versionieren. Mit dem Versioning von Startvorlagen

können Sie eine Teilmenge des vollständigen Satzes an Parametern erstellen. Anschließend können Sie es erneut verwenden, um andere Versionen derselben Startvorlage zu erstellen.

Verwenden Sie AWS Auto Scaling oder implementieren Sie die Skalierung mit den [AWS-APIs oder -SDKs](#) in Ihren Code. Dies reduziert Ihre Gesamtkosten für die Workload, da die Betriebskosten durch manuelle Änderungen an Ihrer Umgebung wegfallen, und kann viel schneller durchgeführt werden. So können Sie sicherstellen, dass Ihre Workload-Ressourcen jederzeit mit Ihrem Bedarf übereinstimmen. Damit Sie diese bewährte Methode befolgen und Ressourcen dynamisch für Ihre Organisation bereitstellen können, sollten Sie die horizontale und vertikale Skalierung in der AWS Cloud sowie die Art der auf den Amazon-EC2-Instances ausgeführten Anwendungen verstehen. Ihr Team für Cloud Financial Management sollte am besten mit den technischen Teams zusammenarbeiten, um diese bewährte Methode zu befolgen.

[Elastic Load Balancing](#) unterstützt Sie bei der Skalierung durch die Verteilung der Nachfrage auf mehrere Ressourcen. Mithilfe von ASG und Elastic Load Balancing können Sie eingehende Anforderungen verwalten, indem Sie den Datenverkehr optimal weiterleiten, sodass keine Instance in einer Auto-Scaling-Gruppe überlastet wird. Die Anforderungen werden nacheinander auf alle Ziele einer Zielgruppe verteilt, ohne Rücksicht auf Kapazität oder Auslastung.

Typische Metriken können Amazon-EC2-Standardmetriken sein, z. B. CPU-Auslastung, Netzwerkdurchsatz und von Elastic Load Balancing beobachtete Anforderungs- und Antwortlatenz. Wenn möglich, sollten Sie eine Metrik verwenden, die auf das Kundenerlebnis hinweist. In der Regel handelt es sich um eine benutzerdefinierte Metrik, die aus Anwendungscode innerhalb Ihrer Workload stammen kann. Um in diesem Dokument zu erläutern, wie die Nachfrage dynamisch gedeckt werden kann, werden wir Auto Scaling in zwei Kategorien einteilen, nämlich nachfragebasierte und zeitbasierte Angebotsmodelle, und uns eingehend mit den einzelnen Modellen befassen.

Nachfragebasiertes Angebot: Nutzen Sie die Elastizität der Cloud, um Ressourcen bereitzustellen, die sich ändernden Anforderungen gerecht werden, indem Sie sich auf den Nachfragestatus nahezu in Echtzeit verlassen. Verwenden Sie für nachfragebasierte Bereitstellung APIs oder Servicefeatures, um die Menge der Cloud-Ressourcen in Ihrer Architektur programmgesteuert zu variieren. Auf diese Weise können Sie Komponenten in Ihrer Architektur skalieren und die Anzahl der Ressourcen in Bedarfsspitzenzeiten zur Aufrechterhalten der Leistung erhöhen und die Kapazität zur Reduzierung der Kosten herabsetzen, wenn der Bedarf abklingt.

Bedarfsorientiertes Angebot (dynamische Skalierungsrichtlinien)

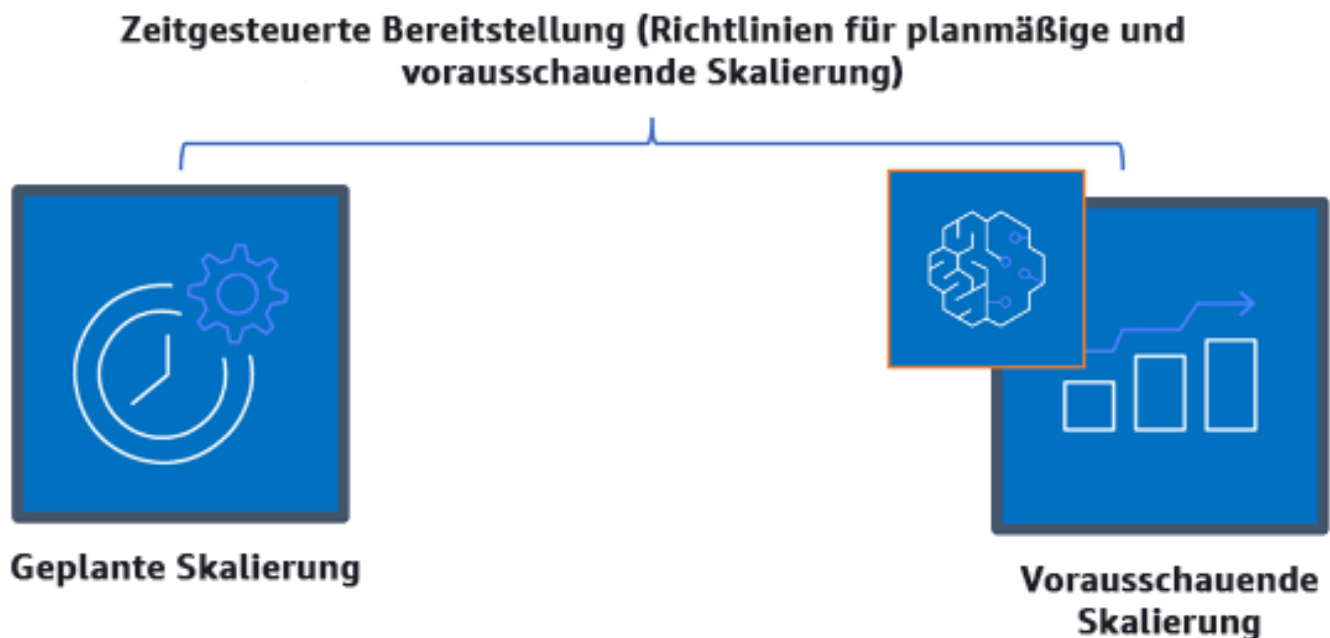


Bedarfsbasierte dynamische Skalierungsrichtlinien

- Einfache/schrittweise Skalierung: Überwacht Metriken und fügt Instances gemäß den vom Kunden manuell definierten Schritten hinzu oder entfernt sie.
- Zielverfolgung: Ein thermostatähnlicher Steuermechanismus, der automatisch Instances hinzufügt oder entfernt, um die Metriken an einem vom Kunden definierten Ziel zu halten.

Beim Aufbau der Architektur mit einem bedarfsbasierten Ansatz sollten Sie die folgenden beiden wichtigen Aspekte berücksichtigen: 1. Machen Sie sich damit vertraut, wie schnell Sie neue Ressourcen bereitstellen müssen. 2. Machen Sie sich damit vertraut, dass sich die Größe der Marge zwischen Angebot und Nachfrage ändern wird. Sie müssen darauf vorbereitet sein, das Intervall der Änderung in Bezug auf die Nachfrage zu verarbeiten, und auch Ressourcenfehler einkalkulieren.

Zeitbasiertes Angebot: Ein zeitbasierter Ansatz richtet die Ressourcenkapazität an der Nachfrage aus, die vorhersehbar oder eindeutig durch die Zeit definiert ist. Dieser Ansatz ist in der Regel nicht abhängig vom Nutzungsgrad der Ressourcen. Mit einem zeitbasierten Ansatz können Sie sicherstellen, dass Ressourcen zu dem Zeitpunkt zur Verfügung stehen, zu dem sie benötigt werden, und ohne Verzögerung aufgrund von Startverfahren und System- oder Konsistenzprüfungen bereitgestellt werden können. Durch die Verwendung eines zeitbasierten Ansatzes können Sie zusätzliche Ressourcen hinzufügen oder die Kapazität in Spitzenzeiten erhöhen.



Zeitbasierte Skalierungsrichtlinien

Sie können geplantes oder vorausschauendes Auto Scaling verwenden, um einen zeitbasierten Ansatz zu implementieren. Workloads können zu bestimmten Zeiten auf Basis eines Zeitplans auf- oder abskaliert werden, z. B. zu Beginn der Geschäftszeiten. Dadurch sind ausreichende Ressourcen verfügbar, wenn die Benutzer ankommen oder die Nachfrage steigt. Die vorausschauende Skalierung verwendet Muster zum Aufskalieren, während bei der geplanten Skalierung vordefinierte Zeiten für die Aufskalierung verwendet werden. Sie können auch eine [Strategie der attributbasierten Auswahl des Instance-Typs](#) in Auto-Scaling-Gruppen verwenden und so die Instance-Anforderungen in Form einer Gruppe von Attributen ausdrücken, z. B. vCPU, Arbeitsspeicher und Speicher. Darüber hinaus können Sie automatisch Instance-Typen neuerer Generationen verwenden, sobald sie veröffentlicht werden, und mit Amazon EC2 Spot Instances auf ein größeres Kapazitätsangebot zugreifen. Amazon-EC2-Flotte und Amazon EC2 Auto Scaling wählen Instances aus, die den angegebenen Attributen entsprechen, und starten diese. So müssen Sie Instance-Typen nicht mehr manuell auswählen.

Sie können auch die [AWS-APIs und -SDKs](#) und [AWS CloudFormation](#) nutzen, um vollständige Umgebungen nach Bedarf automatisch bereitzustellen oder zu deaktivieren. Dieser Ansatz eignet sich hervorragend für Entwicklungs- und Testumgebungen, die nur zu Geschäftszeiten oder in bestimmten Zeiträumen ausgeführt werden. Mit APIs können Sie die Größe der Ressourcen

innerhalb einer Umgebung skalieren (vertikale Skalierung). So könnten Sie beispielsweise eine Produktions-Workload hochskalieren, indem Sie die Instance-Größe oder -Klasse ändern. Stoppen und starten Sie dazu die Instance, und wählen Sie eine andere Instance-Größe oder -Klasse aus. Diese Technik kann auch auf andere Ressourcen angewendet werden, z. B. Amazon EBS Elastic Volumes, bei denen Sie im laufenden Betrieb die Größe ändern, die Leistung anpassen (IOPS) oder den Volume-Typ ändern können.

Beim Aufbau der Architektur mit einem zeitbasierten Ansatz sollten Sie die beiden folgenden wichtigen Aspekte berücksichtigen: 1: Wie konsistent ist das Nutzungsmuster? 2. Wie wirken sich Musteränderungen aus? Sie können die Treffergenauigkeit für Prognosen durch die Überwachung Ihrer Workloads und die Verwendung von Business Intelligence erhöhen. Wenn Sie signifikante Änderungen im Nutzungsmuster erkennen, können Sie die Zeiten ändern, um eine Deckung zu gewährleisten.

Implementierungsschritte

- Konfigurieren der geplanten Skalierung: Für vorhersehbare Änderungen des Bedarfs kann die zeitbasierte Skalierung die richtige Anzahl an Ressourcen in einem angemessenen Zeitraum bereitstellen. Es ist auch nützlich, wenn die Ressourcenerstellung und -konfiguration nicht schnell genug ist, um bei Bedarf auf Änderungen zu reagieren. Mithilfe der Workload-Analyse konfigurieren Sie die geplante Skalierung mithilfe von AWS Auto Scaling. Zur Konfiguration der zeitbasierten Planung können Sie die vorausschauende Skalierung der geplanten Skalierung verwenden, um im Vorfeld die Anzahl der Amazon-EC2-Instances in Ihren Auto-Scaling-Gruppen entsprechend den erwarteten oder prognostizierbaren Lastveränderungen zu erhöhen.
- Konfigurieren der vorausschauenden Skalierung: Mit der vorausschauenden Skalierung können Sie im Voraus die Amazon-EC2-Instances in Ihrer Auto-Scaling-Gruppe anhand von täglichen und wöchentlichen Mustern in Datenverkehrsflüssen erhöhen. Wenn Sie regelmäßige Spitzen beim Datenverkehr sowie Anwendungen haben, die lange brauchen, um zu starten, sollten Sie die vorausschauende Skalierung in Betracht ziehen. Die vorausschauende Skalierung kann Ihnen helfen, schneller zu skalieren, indem die Kapazität vor der prognostizierten Last initialisiert wird, im Gegensatz zur dynamischen Skalierung, die nur reaktiv ist. Wenn die Benutzer Ihre Workloads beispielsweise mit Beginn der Geschäftszeiten nutzen und sie nach Geschäftsschluss nicht mehr brauchen, kann die vorausschauende Skalierung die Kapazität vor den Geschäftszeiten erhöhen. Die Verzögerung, die bei der dynamischen Skalierung entsteht, bis sie auf den veränderten Datenverkehr reagiert, entfällt somit.
- Konfigurieren von dynamischer automatischer Skalierung: Verwenden Sie Auto Scaling, um die Skalierung auf der Grundlage von aktiven Workload-Metriken zu konfigurieren. Verwenden Sie

die Analyse und konfigurieren Sie Auto Scaling so, dass es auf den richtigen Ressourcenebenen gestartet wird. Achten Sie darauf, dass die Workload in der erforderlichen Zeit skaliert wird. Sie können eine Flotte von On-Demand-Instances und Spot-Instances innerhalb einer einzigen Auto-Scaling-Gruppe starten und automatisch skalieren. Zusätzlich zum Erhalt von Rabatten für die Verwendung von Spot-Instances können Sie mit Reserved Instances oder einem Savings Plan Rabatte auf die regulären On-Demand-Instance-Preise erhalten. Alle diese Faktoren zusammengenommen helfen Ihnen dabei, Ihre Kosteneinsparungen für Amazon-EC2-Instances zu optimieren und sicherzustellen, dass Sie die gewünschte Skalierung und Leistung für Ihre Anwendung erreichen.

Ressourcen

Zugehörige Dokumente:

- [AWS Auto Scaling](#)
- [AWS Instance Scheduler](#)
- Skalieren der Größe Ihrer Auto-Scaling-Gruppe
- [Erste Schritte mit Amazon EC2 Auto Scaling](#)
- [Erste Schritte mit Amazon SQS](#)
- [Scheduled Scaling for Amazon EC2 Auto Scaling](#)
- [Predictive scaling for Amazon EC2 Auto Scaling](#)

Zugehörige Videos:

- [Target Tracking Scaling Policies for Auto Scaling](#)
- [AWS Instance Scheduler](#)

Zugehörige Beispiele:

- [Attribute based Instance Type Selection for Auto Scaling for Amazon EC2 Fleet](#)
- [Optimizing Amazon Elastic Container Service for cost using scheduled scaling](#)
- [Predictive Scaling with Amazon EC2 Auto Scaling](#)
- [Wie verwende ich den Instance Scheduler mit CloudFormation, um Amazon-EC2-Instances zu planen?](#)

Optimierung im Laufe der Zeit

Fragen

- [KOSTEN 10. Wie können Sie neue Services bewerten?](#)
- [KOSTEN 11. Wie bewerten Sie die Kosten des Aufwands?](#)

KOSTEN 10. Wie können Sie neue Services bewerten?

Im Zuge der Veröffentlichung neuer Services und Features durch AWS empfiehlt es sich, dass Sie Ihre bestehenden Entscheidungen zur Architektur überdenken, um sicherzustellen, dass diese weiterhin so kostengünstig wie möglich sind.

Best Practices

- [COST10-BP01 Entwickeln Sie einen Prozess zur Überprüfung der Arbeitslast](#)
- [COST10-BP02 Überprüfen und analysieren Sie diesen Workload regelmäßig](#)

COST10-BP01 Entwickeln Sie einen Prozess zur Überprüfung der Arbeitslast

Entwickeln Sie einen Prozess, der die Kriterien und den Prozess für die Workload-Prüfung definiert. Der Überprüfungsaufwand sollte in einem angemessenen Verhältnis zum potenziellen Nutzen stehen. Beispielsweise ist es sinnvoll, zentrale Workloads oder Workloads, deren Wert mehr als 10 % der Rechnung ausmacht, vierteljährlich oder alle sechs Monate zu prüfen, während Workloads mit einem Wert von weniger als 10 % der Rechnung jährlich überprüft werden sollten.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Hoch

Implementierungsleitfaden

Um sicherzustellen, dass Sie immer die kosteneffizienteste Workload haben, müssen Sie die Workload regelmäßig überprüfen, um zu wissen, ob es Möglichkeiten gibt, neue Services, Features und Komponenten zu implementieren. Damit Sie insgesamt niedrigere Kosten erzielen, muss der Prozess proportional zu den potenziellen Einsparungen sein. Beispielsweise sollten Workloads, die 50 % Ihrer Gesamtausgaben ausmachen, regelmäßiger und gründlicher überprüft werden als Workloads, die 5 % Ihrer Gesamtausgaben ausmachen. Lassen Sie auch externe Faktoren oder Volatilität in Ihre Überlegungen einfließen. Wenn die Workload eine bestimmte Geografie oder ein bestimmtes Marktsegment bedient und Änderungen in diesem Bereich vorhergesagt werden, können häufigere Überprüfungen zu Kosteneinsparungen führen. Ein weiterer Faktor bei der Überprüfung ist

die Implementierung von Änderungen. Wenn es erhebliche Kosten für das Testen und Validieren von Änderungen gibt, sollten Überprüfungen seltener erfolgen.

Denken Sie an die langfristigen Kosten für die Wartung veralteter Komponenten und Ressourcen sowie die Unfähigkeit, neue Features in diese zu implementieren. Die aktuellen Kosten für Tests und Validierung können den vorgeschlagenen Vorteil übersteigen. Doch mit der Zeit können sich die Kosten für die Änderung erheblich erhöhen, da die Lücke zwischen der Workload und den aktuellen Technologien zunimmt, was zu noch größeren Kosten führt. Beispielsweise sind die Kosten für den Wechsel zu einer neuen Programmiersprache derzeit möglicherweise nicht günstig. In fünf Jahren können sich jedoch die Kosten für Personen, die in dieser Sprache qualifiziert sind, erhöhen. Aufgrund des Wachstums der Workload würden Sie ein noch größeres System in die neue Sprache verlagern, was noch mehr Aufwand erfordert als zuvor.

Unterteilen Sie Ihre Workload in Komponenten, weisen Sie die Kosten der Komponente zu (eine Schätzung reicht aus) und listen Sie dann die Faktoren (z. B. Aufwand und externe Märkte) neben den einzelnen Komponenten auf. Verwenden Sie diese Indikatoren, um eine Überprüfungshäufigkeit für jede Workload zu bestimmen. Zum Beispiel können bei Webservern hohe Kosten, geringer Änderungsaufwand und hohe externe Faktoren anfallen, was zu einer hohen Überprüfungshäufigkeit führt. Bei einer zentralen Datenbank können mittlere Kosten, hoher Änderungsaufwand und niedrige externe Faktoren anfallen, was zu einer mittleren Überprüfungshäufigkeit führt.

Definieren Sie einen Prozess, mit dem sich neu verfügbare Services, Designmuster, Ressourcentypen und Konfigurationen zur Optimierung Ihrer Workload-Kosten bewerten lassen. Ähnlich wie bei der [Prüfung der Säule „Leistungseffizienz“](#) und der [Prüfung der Säule „Zuverlässigkeit“](#) identifizieren, validieren und priorisieren Sie Optimierungs- und Verbesserungsmaßnahmen. Führen Sie eine Problembehandlung durch und nehmen Sie diese in Ihr Backlog auf.

Implementierungsschritte

- Festlegen der Überprüfungshäufigkeit: Legen Sie fest, wie häufig die Workload und ihre Komponenten überprüft werden sollen. Reservieren Sie Zeit und Ressourcen, um eine kontinuierliche Verbesserungen zu ermöglichen, und prüfen Sie die Häufigkeit, um Ihre Workload zu optimieren und effizienter zu gestalten. Dies ist eine Kombination von Faktoren und kann sich von Workload zu Workload innerhalb Ihrer Organisation und zwischen Komponenten in der Workload unterscheiden. Häufige Faktoren sind u. a. die Bedeutung für die Organisation, gemessen in Bezug auf Umsatz oder Marke, die Gesamtkosten für die Ausführung der Workload (einschließlich Betriebs- und Ressourcenkosten), die Komplexität der Workload, wie einfach es ist, eine Änderung zu implementieren, Softwarelizenzvereinbarungen sowie Änderungen,

die aufgrund mangelhafter Lizenzen erhebliche Erhöhungen der Lizenzkosten verursachen würden. Komponenten können funktional oder technisch definiert werden, z. B. Webserver und Datenbanken oder Datenverarbeitungs- und Speicherressourcen. Gleichen Sie die Faktoren entsprechend aus und entwickeln Sie einen Zeitraum für die Workload und ihre Komponenten. Sie können sich entscheiden, die vollständigen Workload alle 18 Monate, die Webserver alle 6 Monate, die Datenbank alle 12 Monate, die Datenverarbeitungs- und Kurzzeitspeicherung alle 6 Monate und die Langzeitspeicherung alle 12 Monate zu überprüfen.

- Festlegen der Prüfungsgründlichkeit: Legen Sie den Aufwand für die Prüfung der Workload oder der Workload-Komponenten fest. Ähnlich wie bei der Überprüfungsfrequenz geht es hier um mehrere Faktoren, die ausgeglichen sein müssen. Bewerten und priorisieren Sie regelmäßig Verbesserungsmöglichkeiten, um die Maßnahmen dort zu intensivieren, wo sie den größten Nutzen bringen. So erfahren Sie auch, wie viel Aufwand für diese Aktivitäten erforderlich ist. Wenn die erwarteten Ergebnisse die Ziele nicht erfüllen und der Aufwand mehr kostet, wiederholen Sie den Versuch mit alternativen Vorgehensweisen. Bei Ihren Prüfungen sollten auch Zeit und Ressourcen genutzt werden, um kontinuierliche, schrittweise Verbesserungen zu ermöglichen. Sie können beispielsweise entscheiden, für die Analyse der Datenbankkomponente eine Woche, für die Analyse von Datenverarbeitungsressourcen eine Woche und für die Analyse von Speicherprüfungen vier Stunden aufzuwenden.

Ressourcen

Zugehörige Dokumente:

- [AWS Nachrichten-Blog](#)
- [Arten von Cloud Computing](#)
- [Neuerungen bei AWS](#)

Zugehörige Beispiele:

- [AWS Support Sie proaktive Services](#)
- [Regelmäßige Workload-Überprüfungen von SAP Workloads](#)

COST10-BP02 Überprüfen und analysieren Sie diesen Workload regelmäßig

Bestehende Workloads werden basierend auf den einzelnen definierten Prozessen regelmäßig überprüft, um zu ermitteln, ob neue Services übernommen, vorhandene Services ersetzt oder die Architektur von Workloads geändert werden können.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

AWS fügt ständig neue Funktionen hinzu, sodass Sie mit der neuesten Technologie schneller experimentieren und Innovationen umsetzen können. [AWS Was AWS ist neu?](#) beschreibt, wie das funktioniert, und bietet einen schnellen Überblick über AWS Dienste, Funktionen und Ankündigungen regionaler Erweiterungen, sobald diese veröffentlicht werden. Sie können sich eingehender über die angekündigten Veröffentlichungen informieren und diese zur Prüfung und Analyse Ihrer bestehenden Workloads verwenden. Um die Vorteile neuer AWS Dienste und Funktionen zu nutzen, überprüfen Sie Ihre Workloads und implementieren bei Bedarf neue Dienste und Funktionen. Das bedeutet, dass Sie möglicherweise bestehende Dienste, die Sie für Ihre Workloads verwenden, ersetzen oder Ihren Workload modernisieren müssen, um diese neuen AWS Services einzuführen. Sie können beispielsweise Ihre Workloads überprüfen und die Messaging-Komponente durch Amazon Simple Email Service ersetzen. Dadurch entfallen die Kosten für den Betrieb und die Verwaltung einer Flotte von Instances, während die gesamte Funktionalität zu geringeren Kosten bereitgestellt wird.

Bei der Analyse Ihrer Workload und der Ermittlung potenzieller Chancen sollten Sie nicht nur neue Services, sondern auch neue Möglichkeiten zur Entwicklung von Lösungen berücksichtigen. Sehen Sie sich die Videos von [This is My Architecture](#) an AWS, um mehr über die Architekturentwürfe anderer Kunden, ihre Herausforderungen und Lösungen zu erfahren. Schauen Sie sich die [All-In-Serie an](#), um mehr über reale Anwendungen von AWS Diensten und Kundenberichte zu erfahren. Sie können sich auch die Videoreihe [Back to Basics](#) (Zurück zu den Grundlagen) ansehen, in der bewährte Methoden zum grundlegenden Cloud-Architekturmuster erklärt, untersucht und aufgeschlüsselt werden. Eine weitere Quelle sind die Videos „[How to Build This](#)“, die Menschen mit großen Ideen dabei helfen sollen, ihr Produkt (MVP) mit Hilfe von AWS Dienstleistungen zum Leben zu erwecken. Auf diese Weise können sich Bauherren aus der ganzen Welt, die eine starke Idee haben, architektonische Beratung von erfahrenen AWS Solutions Architects einholen. In unseren Ressourcenmaterialien unter [Erste Schritte](#) finden Sie darüber hinaus ausführliche Tutorials.

Befolgen Sie vor dem Starten Ihres Überprüfungsprozesses die Anforderungen Ihres Unternehmens in Bezug auf die Workload, die Sicherheit und den Datenschutz, um einen spezifischen Service oder

eine spezifische Region zu nutzen. Befolgen Sie während des vereinbarten Überprüfungsprozesses die Leistungsanforderungen.

Implementierungsschritte

- **Regelmäßige Überprüfung der Workload:** Führen Sie mit Ihrem definierten Prozess Überprüfungen mit der angegebenen Häufigkeit durch. Stellen Sie sicher, dass Sie den richtigen Aufwand für jede Komponente aufwenden. Dieser Prozess ähnelt dem anfänglichen Designprozess, bei dem Sie Services für die Kostenoptimierung ausgewählt haben. Analysieren Sie die Services und die Vorteile, die sie mit sich bringen würden, sowie den Zeitfaktor bei den Änderungskosten. Analysieren Sie nicht nur die langfristigen Vorteile.
- **Implementieren neuer Services:** Wenn es das Ziel der Analyse ist, Änderungen zu implementieren, führen Sie zunächst eine Analyse der Basisanforderungen der Workload durch, um die aktuellen Kosten für jede Ausgabe festzustellen. Implementieren Sie die Änderungen und führen Sie dann eine Analyse durch, um die neuen Kosten für jede Ausgabe zu bestätigen.

Ressourcen

Zugehörige Dokumente:

- [AWS Nachrichten-Blog](#)
- [Neuerungen bei AWS](#)
- [AWS Dokumentation](#)
- [AWS Erste Schritte](#)
- [AWS Allgemeine Ressourcen](#)

Zugehörige Videos:

- [AWS - Das ist meine Architektur](#)
- [AWS - Zurück zu den Grundlagen](#)
- [AWS - All-In-Serie](#)
- [How to Build This](#)

KOSTEN 11. Wie bewerten Sie die Kosten des Aufwands?

Best Practices

- [COST11-BP01 Durchführen von Automatisierungen für Betriebsabläufe](#)

COST11-BP01 Durchführen von Automatisierungen für Betriebsabläufe

Bewerten Sie die Betriebskosten in der Cloud und konzentrieren Sie sich dabei auf die Quantifizierung der Zeit- und Aufwandsersparnisse bei administrativen Aufgaben und Bereitstellungen, der Minimierung des Risikos menschlicher Fehler, Compliance und anderen Abläufen durch Automatisierung. Ermitteln Sie den Zeitaufwand und die damit verbundenen Kosten, die für den betrieblichen Aufwand erforderlich sind, und implementieren Sie die Automatisierung von Verwaltungsaufgaben, um den manuellen Aufwand zu minimieren, wo immer dies möglich ist.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Niedrig

Implementierungsleitfaden

Die Automatisierung von Abläufen reduziert die Häufigkeit von manuellen Aufgaben, optimiert die Effizienz und bietet Kunden Vorteile, indem sie eine konsistente und zuverlässige Umgebung bei der Bereitstellung, Verwaltung oder dem Betrieb von Workloads ermöglicht. Sie können Infrastrukturressourcen von manuellen Betriebsaufgaben entlasten und für hochwertigere Aufgaben sowie Innovationen einsetzen, was den Unternehmenswert verbessert. Unternehmen benötigen eine bewährte, getestete Möglichkeit, ihre Workloads in der Cloud zu verwalten. Diese Lösung muss sicher, schnell und kostengünstig sein, mit minimalem Risiko und maximaler Zuverlässigkeit.

Beginnen Sie damit, Ihre Betriebsabläufe basierend auf dem erforderlichen Aufwand zu priorisieren, indem Sie sich die Gesamtbetriebskosten ansehen. Beispiel: Wie lange dauert es, neue Ressourcen in der Cloud bereitzustellen, vorhandene Ressourcen zu optimieren oder die notwendigen Konfigurationen zu implementieren? Sehen Sie sich die Gesamtkosten für die menschliche Arbeitskraft an und berücksichtigen Sie dabei die Kosten für Betriebsabläufe und Verwaltung. Priorisieren Sie die Automatisierung von Verwaltungsaufgaben, um die menschliche Arbeitskraft zu reduzieren.

Der Überprüfungsaufwand sollte in einem angemessenen Verhältnis zum potenziellen Nutzen stehen. Beispiel: Untersuchen Sie den Zeitaufwand für das manuelle im Vergleich zum automatischen Ausführen von Aufgaben. Priorisieren Sie die Automatisierung sich wiederholender, hochwertiger, zeitaufwändiger und komplexer Aktivitäten. Aufgaben mit hohem Wert oder einem hohen Risiko von menschlichen Fehlern sind in der Regel der bessere Ausgangspunkt für Automatisierungen, da das Risiko oft unerwünschte zusätzliche Betriebskosten (z. B. für Überstunden des Betriebsteams) mit sich bringt.

Verwenden Sie Automatisierungstools wie AWS Systems Manager oder AWS Config, um Betriebs-, Compliance-, Überwachungs-, Lebenszyklus- und Beendigungsprozesse zu optimieren. Mit AWS-Services und -Tools und Produkten von Drittanbietern können Sie die von Ihnen implementierten Automatisierungen an Ihre spezifischen Anforderungen anpassen. Die folgende Tabelle zeigt einige der zentralen Betriebsfunktionen der AWS-Services, mit denen Sie die Verwaltung und die Betriebsabläufe automatisieren können:

- [AWS Audit Manager](#): Kontinuierliche Überprüfung Ihrer AWS-Nutzung, um die Risiko- und Compliance-Bewertung zu vereinfachen
- [AWS Backup](#): Zentrale Verwaltung und Automatisierung des Datenschutzes
- [AWS Config](#): Konfiguration von Datenverarbeitungsressourcen, Bewertung, Prüfung und Auswertung von Konfigurationen und Ressourceninventar
- [AWS CloudFormation](#): Starten von hochverfügbaren Ressourcen mit Infrastructure as Code
- [AWS CloudTrail](#): IT-Änderungsmanagement, Compliance und Kontrolle
- [Amazon EventBridge](#): Planen von Ereignissen und Auslösen von Maßnahmen durch AWS Lambda
- [AWS Lambda](#): Automatisieren sich wiederholender Prozesse, indem sie durch Ereignisse ausgelöst oder mit AWS nach einem festen Zeitplan ausgeführt werden
- [AWS Systems Manager](#): Starten und Stoppen von Workloads, Patchen von Betriebssystemen, automatische Konfiguration und dauerhafte Verwaltung
- [AWS Step Functions](#): Planen von Aufträgen und Automatisieren von Workflows
- [AWS Service Catalog](#): Vorlagennutzung und Infrastructure as Code mit Compliance und Kontrolle

Wenn Sie sofort Automatisierungen mit AWS-Produkten und -Services einführen möchten, in Ihrer Organisation jedoch nicht über die erforderliche Kompetenz verfügen, wenden Sie sich an [AWS Managed Services \(AMS\)](#), [AWS Professional Services](#) oder [AWS-Partner](#), um die Automatisierung in höherem Umfang zu nutzen und Ihre operative Exzellenz in der Cloud zu verbessern.

AWS Managed Services (AMS) ist ein Service, der die AWS-Infrastruktur für Unternehmenskunden und -partner betreibt. Er bietet eine sichere und konforme Umgebung, in der Sie Ihre Workloads bereitstellen können. AMS verwendet Enterprise-Cloud-Betriebsmodelle mit Automatisierung, damit Sie Ihre Organisationsanforderungen erfüllen, schneller in die Cloud wechseln und Ihre laufenden Verwaltungskosten senken können.

AWS Professional Services kann Sie auch dabei unterstützen, die gewünschten Geschäftsziele zu erreichen und Betriebsabläufe mit AWS zu automatisieren. Sie unterstützen die Kunden bei der Bereitstellung von automatisierten, robusten und agilen IT-Abläufen sowie für die Cloud optimierten

Governance-Funktionen. Detaillierte Überwachungsbeispiele und empfohlene bewährte Methoden finden Sie im Whitepaper zur Säule für die betriebliche Effizienz.

Implementierungsschritte

- Einmal entwickeln und mehrmals bereitstellen: Verwenden Sie Infrastructure as Code wie beispielsweise CloudFormation, AWS SDK oder AWS CLI zur einmaligen Bereitstellung und mehrfachen Nutzung für ähnliche Umgebungen oder für die Notfallwiederherstellung. Nutzen Sie während der Bereitstellung Tags, um die Nutzung wie in anderen bewährten Methoden beschrieben zu verfolgen. Verwenden Sie [AWS Launch Wizard](#), um die erforderliche Zeit für die Bereitstellung vieler beliebter Unternehmens-Workloads zu reduzieren. AWS Launch Wizard leitet Sie durch die Dimensionierung, Konfiguration und Bereitstellung von Unternehmens-Workloads gemäß den bewährten Methoden von AWS. Sie können auch den [Servicekatalog](#) verwenden. Dieser unterstützt Sie bei der Erstellung und Verwaltung von genehmigten Vorlagen für Infrastructure as Code zur Verwendung in AWS, sodass alle Mitarbeiter genehmigte Selfservice-Cloud-Ressourcen erkunden können.
- Automatisieren von kontinuierlicher Compliance: Erwägen Sie, die Bewertung und Korrektur aufgezeichneter Konfigurationen anhand vordefinierter Standards zu automatisieren. Durch die Kombination von AWS Organizations mit den Funktionen von AWS Config und [AWS CloudFormation](#) können Sie die Konfigurations-Compliance für Hunderte von Mitgliedskonten in großem Umfang effizient verwalten und automatisieren. Sie können Änderungen an Konfigurationen und Beziehungen zwischen AWS-Ressourcen überprüfen und den Verlauf einer Ressourcenkonfiguration nachverfolgen.
- Automatisieren von Überwachungsaufgaben: AWS bietet verschiedene Tools, mit denen Sie Services überwachen können. Sie können diese Tools so konfigurieren, dass sie Überwachungsaufgaben automatisieren. Erstellen und implementieren Sie einen Überwachungsplan, der Überwachungsdaten aus allen Teilen Ihrer Workload erfasst, sodass Sie einen Mehrpunktausfall, falls ein solcher auftritt, einfacher debuggen können. Beispielsweise können Sie die automatisierten Überwachungstools verwenden, um Amazon EC2 zu beobachten und benachrichtigt zu werden, wenn bei Systemstatusprüfungen, Instance-Statusprüfungen und Amazon-CloudWatch-Alarmen etwas nicht stimmt.
- Automatisieren von Wartung und Betrieb: Führen Sie Routineaufgaben automatisch ohne menschliche Eingriffe aus. Mithilfe von AWS-Services und -Tools können Sie auswählen, welche AWS-Automatisierungen Sie implementieren und an Ihre spezifischen Anforderungen anpassen möchten. Verwenden Sie beispielsweise [EC2 Image Builder](#) zum Entwickeln, Testen und Bereitstellen von virtuellen Maschinen und Container-Images zur Verwendung in AWS oder On-Premises oder zum Patchen Ihrer EC2-Instances mit AWS SSM. Wenn die gewünschte

Aktion nicht mit AWS-Services ausgeführt werden kann oder Sie komplexere Aktionen mit Filterung der Ressourcen benötigen, automatisieren Sie Ihre Betriebsabläufe mit [AWS Command Line Interface](#) (AWS CLI) oder AWS-SDK-Tools. AWS CLI bietet die Möglichkeit, die gesamte Kontrolle und Verwaltung von AWS-Services mit Skripten zu automatisieren, ohne dass die AWS-Managementkonsole verwendet werden muss. Wählen Sie Ihre bevorzugten AWS-SDKs aus, um mit den AWS-Services zu interagieren. Weitere Codebeispiele finden Sie im [Repository mit AWS-SDK-Codebeispielen](#).

- Schaffen eines kontinuierlichen Lebenszyklus mit Automatisierungen: Es ist wichtig, dass Sie ausgereifte Lebenszyklusrichtlinien einrichten und beibehalten, nicht nur für Vorschriften oder Redundanz, sondern auch für die Kostenoptimierung. Sie können AWS Backup verwenden, um den Datenschutz von Datenspeichern wie Buckets, Volumes, Datenbanken und Dateisystemen zentral zu verwalten und zu automatisieren. Sie können auch Amazon Data Lifecycle Manager verwenden, um die Erstellung, Aufbewahrung und Löschung von EBS-Snapshots und EBS-gestützten AMIs zu automatisieren.
- Löschen von unnötigen Ressourcen: Es ist durchaus üblich, ungenutzte Ressourcen in der Sandbox oder in Entwicklungs-AWS-Konten anzusammeln. Entwickler erstellen und experimentieren im Rahmen des normalen Entwicklungszyklus mit verschiedenen Services und Ressourcen, und dann löschen sie diese Ressourcen nicht, wenn sie nicht mehr benötigt werden. Ungenutzte Ressourcen können unnötige und manchmal hohe Kosten für die Organisation verursachen. Durch das Löschen dieser Ressourcen können die Kosten für den Betrieb dieser Umgebungen gesenkt werden. Vergewissern Sie sich im Zweifelsfall, dass diese Daten nicht benötigt werden oder gesichert sind. Sie können AWS CloudFormation verwenden, um bereitgestellte Stapel zu bereinigen, wodurch die meisten in der Vorlage definierten Ressourcen automatisch gelöscht werden. Alternativ können Sie mit Tools wie [aws-nuke](#) eine Automatisierung zum Löschen von AWS-Ressourcen einrichten.

Ressourcen

Zugehörige Dokumente:

- [Modernizing operations in the AWS Cloud](#)
- [AWS Services for Automation](#)
- [Integration & Automation](#)
- [AWS Systems Manager Automation](#)
- [Automated and manual monitoring](#)
- [AWS automations for SAP administration and operations](#)

- [AWS Managed Services](#)
- [AWS Professional Services](#)

Zugehörige Videos:

- [Automate Continuous Compliance at Scale in AWS](#)
- [AWS Backup Demo: Cross-Account & Cross-Region Backup](#)
- [Patching for your Amazon EC2 Instances](#)

Zugehörige Beispiele:

- [Reinventing automated operations \(Part I\)](#)
- [Reinventing automated operations \(Part II\)](#)
- [Automate deletion of AWS resources by using aws-nuke](#)
- [Delete unused Amazon EBS volumes by using AWS Config and AWS SSM](#)
- [Automate continuous compliance at scale in AWS](#)
- [IT Automations with AWS Lambda](#)

Nachhaltigkeit

Bei der Säule „Nachhaltigkeit“ geht es darum, die Auswirkungen der genutzten Services zu verstehen, diese über den gesamten Workload-Lebenszyklus hinweg zu quantifizieren sowie konzeptionelle Grundsätze und bewährte Methoden einzusetzen, die dabei helfen, diese Auswirkungen zu reduzieren, wenn Cloud-Workloads erstellt werden. Verbindliche Anleitungen zur Implementierung finden Sie im [Whitepaper „Säule der Nachhaltigkeit“](#).

Bereiche für bewährte Methoden

- [Auswahl der Region](#)
- [Ausrichtung am Bedarf](#)
- [Software und Architektur](#)
- [Daten](#)
- [Hardware und Services](#)
- [Prozess und Kultur](#)

Auswahl der Region

Frage

- [SUS 1 Wie wählen Sie Regionen für Ihre Workload aus?](#)

SUS 1 Wie wählen Sie Regionen für Ihre Workload aus?

Welche Region Sie für Ihre Workload auswählen, hat signifikante Auswirkungen auf deren KPIs, u. a. Leistung, Kosten und CO2-Bilanz. Um diese KPIs effizient zu verbessern, sollten Sie die Regionen für Ihre Workloads in Abhängigkeit von den Unternehmensanforderungen und Nachhaltigkeitszielen auswählen.

Best Practices

- [SUS01-BP01 Auswählen der Region auf Grundlage von Unternehmensanforderungen und Nachhaltigkeitszielen](#)

SUS01-BP01 Auswählen der Region auf Grundlage von Unternehmensanforderungen und Nachhaltigkeitszielen

Wählen Sie für Ihre Workload eine Region auf der Grundlage Ihrer Geschäftsanforderungen und Nachhaltigkeitsvorgaben aus, um KPIs wie Leistung, Kosten und CO2-Bilanz zu optimieren.

Typische Anti-Muster:

- Sie wählen die Region der Workload auf der Grundlage Ihres eigenen Standorts aus.
- Sie konsolidieren alle Workload-Ressourcen an einem einzelnen geografischen Standort.

Vorteile der Nutzung dieser bewährten Methode: Wenn Sie eine Workload in der Nähe von Amazon-Projekten für erneuerbare Energien oder in Regionen mit nachweislich niedrigen Kohlendioxidemissionen platzieren, kann die CO2-Bilanz einer Clouds-Workload gesenkt werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Die AWS Cloud ist ein ständig wachsendes Netzwerk aus Regionen und Points of Presence (PoP), die durch eine globale Netzwerkinfrastruktur verbunden sind. Die Wahl der Region für Ihre Workload

hat erhebliche Auswirkungen auf zugehörige KPIs wie Leistung, Kosten und CO2-Bilanz. Um diese KPIs effizient zu verbessern, sollten Sie Regionen für Ihre Workload abhängig von den Unternehmensanforderungen und Nachhaltigkeitszielen auswählen.

Implementierungsschritte

- Erstellen einer Shortlist potenzieller Regionen: Führen Sie die folgenden Schritte aus, um Regionen zu bewerten und eine Shortlist potenzieller Regionen für Ihren Workload zu erstellen, basierend auf Ihren geschäftlichen Anforderungen, einschließlich Compliance, verfügbarer Features, Kosten und Latenz:
 - Überprüfen Sie, ob diese Regionen mit den lokalen Vorschriften konform sind (z. B. hinsichtlich der Datensouveränität).
 - Prüfen Sie anhand der [Liste der verfügbaren AWS-Services nach Regionen](#), ob die Regionen über die für Ihre Workload erforderlichen Services und Funktionen verfügen.
 - Berechnen Sie die Kosten der Workload in jeder Region mithilfe des [AWS Pricing Calculator](#).
 - Testen Sie die Netzwerklatenz zwischen den Standorten Ihrer Endbenutzer und jeder AWS-Region.
- Auswählen von Regionen: Wählen Sie Regionen in der Nähe von Amazon-Projekten für erneuerbare Energien und Regionen mit geringeren Kohlendioxidemissionen als an anderen Standorten (oder in anderen Regionen) aus.
 - Ermitteln Sie die relevanten Nachhaltigkeitsrichtlinien, um die jährlichen CO2-Emissionen gemäß dem [Greenhouse Gas Protocol](#) nachzuverfolgen und zu vergleichen (marktbasierte und standortbasierte Verfahren).
 - Wählen Sie die Region entsprechend der Methode aus, mit der Sie CO2-Emissionen nachverfolgen. Weitere Informationen zum Auswählen einer Region anhand von Nachhaltigkeitsrichtlinien finden Sie im [Artikel zum Auswählen einer Region für Ihre Workload auf der Grundlage von Nachhaltigkeitszielen](#).

Ressourcen

Zugehörige Dokumente:

- [Grundlegendes zu CO2-Emissionsschätzungen](#)
- [Amazon weltweit](#)
- [Methodik für erneuerbare Energien](#)
- [„Relevante Aspekte bei der Wahl einer Region für Ihre Workloads“ erläutert](#)

Zugehörige Videos:

- [AWS re:Invent 2023 – Nachhaltigkeitsinnovationen in der globalen AWS-Infrastruktur](#)
- [AWS re:Invent 2023 – Nachhaltige Architektur: Vergangenheit, Gegenwart und Zukunft](#)
- [AWS re:Invent 2022 – Bereitstellung nachhaltiger, leistungsstarker Architekturen](#)
- [AWS re:Invent 2022 – Nachhaltige Architektur und Reduzieren der AWS-CO2-Bilanz](#)
- [AWS re:Invent 2022 – Nachhaltigkeit in der globalen AWS-Infrastruktur](#)

Ausrichtung am Bedarf

Frage

- [SUS 2 Wie richten Sie Cloud-Ressourcen am Bedarf aus?](#)

SUS 2 Wie richten Sie Cloud-Ressourcen am Bedarf aus?

Wenn Sie berücksichtigen, wie Benutzer und Anwendungen Ihre Workloads und andere Ressourcen nutzen, können Sie auf diese Weise Verbesserungsmöglichkeiten ermitteln, um Nachhaltigkeitsziele zu erreichen. Skalieren Sie Ihre Infrastruktur so, dass Sie den Bedarf kontinuierlich anpassen können. Sorgen Sie zudem dafür, dass zur Unterstützung Ihrer Benutzer nicht mehr Ressourcen verwendet werden als unbedingt nötig. Richten Sie Service-Levels an den Kundenanforderungen aus. Positionieren Sie Ressourcen so, dass die Netzwerkkapazitäten, die für Benutzer und Anwendungen erforderlich sind, begrenzt werden. Entfernen Sie ungenutzte Komponenten. Stellen Sie Teammitgliedern Geräte zur Verfügung, die ihre Anforderungen bei geringstmöglichen Auswirkungen auf die Nachhaltigkeit erfüllen.

Best Practices

- [SUS02-BP01 Dynamisches Skalieren der Workload-Infrastruktur](#)
- [SUS02-BP02 An den Nachhaltigkeitszielen SLAs ausrichten](#)
- [SUS02-BP03 Beenden der Erstellung und Wartung nicht verwendeter Komponenten](#)
- [SUS02-BP04 Optimieren Sie die geografische Verteilung von Workloads auf der Grundlage ihrer Netzwerkanforderungen](#)
- [SUS02-BP05 Optimieren von Ressourcen für Teammitglieder im Hinblick auf die ausgeführten Aktivitäten](#)
- [SUS02-BP06 Implementieren Sie Pufferung oder Drosselung, um die Nachfragekurve abzuflachen](#)

SUS02-BP01 Dynamisches Skalieren der Workload-Infrastruktur

Nutzen Sie die Elastizität der Cloud und skalieren Sie Ihre Infrastruktur dynamisch, um das Angebot an Cloud-Ressourcen an den Bedarf anzupassen und eine Überbereitstellung von Kapazitäten in Ihrer Workload zu vermeiden.

Typische Anti-Muster:

- Sie skalieren Ihre Infrastruktur nicht mit der Benutzerlast.
- Sie skalieren Ihre Infrastruktur immer manuell.
- Sie behalten die erhöhte Kapazität nach dem Hochskalieren bei, anstatt sie wieder herunterzuskalieren.

Vorteile der Nutzung dieser bewährten Methode: Das Konfigurieren und Testen der Workload-Elastizität trägt dazu bei, das Angebot an Cloud-Ressourcen effizient an den Bedarf anzupassen und eine Überbereitstellung von Kapazitäten zu vermeiden. Sie können die Vorteile der Elastizität in der Cloud nutzen, um die Kapazität während und nach Bedarfsspitzen automatisch zu skalieren und so sicherzustellen, dass Sie nur die Menge an Ressourcen nutzen, die für die Erfüllung Ihrer Geschäftsanforderungen erforderlich ist.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Die Cloud bietet Ihnen die Flexibilität, Ressourcen dynamisch durch verschiedene Mechanismen zu erweitern oder zu reduzieren, um einem veränderten Bedarf gerecht zu werden. Eine optimale Abstimmung von Angebot und Bedarf führt zu den geringsten Auswirkungen auf die Umgebung für eine Workload.

Der Bedarf kann fest oder variabel sein und erfordert Metriken und Automatisierung, um sicherzustellen, dass die Verwaltung nicht zur Last wird. Anwendungen können vertikal (hoch oder herunter) und/oder horizontal (ab oder auf) skaliert werden. Bei der vertikalen Skalierung wird die Instance-Größe geändert, bei der horizontalen Skalierung die Anzahl von Instances.

Sie können verschiedene Ansätze nutzen, um das Angebot an Ressourcen auf den Bedarf abzustimmen.

- Zielverfolgungsansatz: Überwachen Sie Ihre Skalierungsmetriken und erhöhen oder verringern Sie die Kapazität automatisch nach Bedarf.

- **Prädiktives Skalieren:** Skalieren Sie auf der Grundlage erwarteter täglicher und wöchentlicher Trends.
- **Zeitplanbasierter Ansatz:** Legen Sie einen eigenen Skalierungszeitplan auf der Grundlage vorhersehbarer Laständerungen fest.
- **Service-Skalierung:** Wählen Sie Services (beispielsweise Serverless) aus, die nativ von Natur aus skalierbar sind oder Auto Scaling als Feature bieten.

Identifizieren Sie Zeiträume mit geringer oder gar keiner Nutzung und skalieren Sie Ressourcen, um überschüssige Kapazitäten zu entfernen und die Effizienz zu verbessern.

Implementierungsschritte

- Elastizität ermöglicht die Anpassung der verfügbaren Ressourcen an den Bedarf. Instances, Container und Funktionen bieten Mechanismen für Elastizität, entweder in Kombination mit Auto Scaling oder als Feature des Service. AWS bietet eine Reihe von Mechanismen für Auto Scaling, um sicherzustellen, dass Workloads in Zeiten geringer Benutzerlast schnell und einfach herunterskaliert werden können. Hier sind einige Beispiele für Auto-Scaling-Mechanismen:

Auto-Scaling-Mechanismus	Verwendung
Amazon EC2 Auto Scaling	Verwenden Sie diesen Mechanismus, um zu überprüfen, ob die richtige Menge an Amazon-EC2-Instances zur Verfügung steht, um die Benutzerlast für Ihre Anwendung zu bewältigen.
Application Auto Scaling	Verwenden Sie diesen Mechanismus, um die Ressourcen für einzelne AWS-Services über Amazon EC2 hinaus automatisch zu skalieren – beispielsweise Lambda-Funktionen oder Amazon Elastic Container Service (Amazon ECS)-Services.
Kubernetes Cluster Autoscaler	Verwenden Sie diesen Mechanismus, um Kubernetes-Cluster in AWS automatisch zu skalieren.

- Skalieren wird häufig im Zusammenhang mit Datenverarbeitungs-Services wie Amazon-EC2-Instances oder AWS Lambda-Funktionen behandelt. Ziehen Sie die Konfiguration von nicht Daten verarbeitenden Services wie Lese- und Schreibkapazitätseinheiten von [Amazon DynamoDB](#) oder Shards von [Amazon Kinesis Data Streams](#) in Betracht, um den Bedarf zu decken.
- Vergewissern Sie sich, dass die Metriken zum Hoch- oder Herunterskalieren für die jeweilige Art der bereitgestellten Workload überprüft werden. Wenn Sie eine Anwendung zur Video-Transkodierung bereitstellen, wird eine CPU-Auslastung von 100 % erwartet, weshalb dies nicht die Hauptmetrik sein sollte. Sie können bei Bedarf eine [benutzerdefinierte Metrik](#) (etwa die Speicherauslastung) für Ihre Skalierungsrichtlinie verwenden. Beachten Sie bei der Wahl geeigneter Metriken die folgenden Hinweise für Amazon EC2:
 - Es muss sich um eine gültige Nutzungsmetrik handeln, die beschreibt, wie stark eine Instance genutzt wird.
 - Der Wert der Metrik muss sich proportional zur Anzahl der Instances in der Auto-Scaling-Gruppe erhöhen oder verringern.
- Verwenden Sie für Ihre Auto-Scaling-Gruppe eine [dynamische Skalierung](#) anstelle einer [manuellen Skalierung](#). Außerdem empfiehlt es sich, bei der dynamischen Skalierung [Skalierungsrichtlinien zur Zielverfolgung](#) zu verwenden.
- Vergewissern Sie sich, dass Workload-Bereitstellungen sowohl Hoch- als auch Herunterskalierungsereignisse behandeln können. Erstellen Sie Testszenarien für Herunterskalierungsereignisse, um sich zu vergewissern, dass sich die Workload wie erwartet verhält und die Benutzererfahrung nicht beeinträchtigt wird (etwa durch den Verlust von Sticky Sessions). Sie können den [Aktivitätsverlauf](#) verwenden, um eine Skalierungsaktivität für eine Auto-Scaling-Gruppe zu überprüfen.
- Überprüfen Sie Ihre Workload auf vorhersagbare Muster und skalieren Sie proaktiv, wenn Sie vorhergesagte und geplante Bedarfsänderungen erwarten. Mit der prädiktiven Skalierung können Sie die Notwendigkeit einer Überbereitstellung von Kapazitäten vermeiden. Weitere Einzelheiten finden Sie unter [Prädiktive Skalierung mit Amazon EC2 Auto Scaling](#).

Ressourcen

Zugehörige Dokumente:

- [Erste Schritte mit Amazon EC2 Auto Scaling](#)
- [Prädiktive Skalierung für EC2, unterstützt von Machine Learning](#)

- [Analysieren des Benutzerverhaltens mit Amazon OpenSearch Service, Amazon Data Firehose und Kibana](#)
- [Was ist Amazon CloudWatch?](#)
- [Überwachen der DB-Last mit Performance Insights in Amazon RDS](#)
- [Vorstellung von nativer Unterstützung für die prädiktive Skalierung mit Amazon EC2 Auto Scaling](#)
- [Vorstellung von Karpenter – Open-Source-Kubernetes-Cluster-Autoscaler mit hoher Leistung](#)
- [Detaillierte Einblicke in Amazon ECS Cluster Auto Scaling](#)

Zugehörige Videos:

- [AWS re:Invent 2023 – Skalierung in AWS für die ersten 10 Millionen Benutzer](#)
- [AWS re:Invent 2023 – Nachhaltige Architektur: Vergangenheit, Gegenwart und Zukunft](#)
- [AWS re:Invent 2022 – Entwickeln einer kosten-, energie- und ressourceneffizienten Computing-Umgebung](#)
- [AWS re:Invent 2022 – Container-Skalierung von einem einzelnen Benutzer auf mehrere Millionen](#)
- [AWS re:Invent 2023 – Skalierung der FM-Inferenz auf Hunderte von Modellen mit Amazon SageMaker AI](#)
- [AWS re:Invent 2023 – Nutzung der Leistungsfähigkeit von Karpenter für die Skalierung, Optimierung und Aktualisierung von Kubernetes](#)

Zugehörige Beispiele:

- [Auto Scaling](#)

SUS02-BP02 An den Nachhaltigkeitszielen SLAs ausrichten

Überprüfen und optimieren Sie die Service-Level-Vereinbarungen (SLA) für die Arbeitslast auf der Grundlage Ihrer Nachhaltigkeitsziele, um den Ressourcenaufwand für Ihre Arbeitslast zu minimieren und gleichzeitig die Geschäftsanforderungen zu erfüllen.

Typische Anti-Muster:

- SLAsDie Arbeitslast ist unbekannt oder mehrdeutig.
- Sie definieren Ihren SLA nur aus Gründen der Verfügbarkeit und Leistung.
- Sie verwenden für alle Ihre Workloads die gleichen Designmuster (wie etwa Multi-AZ-Architektur).

Vorteile der Einführung dieser bewährten Methode: Die Ausrichtung an SLAs Nachhaltigkeitszielen führt zu einer optimalen Ressourcennutzung bei gleichzeitiger Erfüllung der Geschäftsanforderungen.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Niedrig

Implementierungsleitfaden

SLAsdefinieren Sie das Serviceniveau, das von einem Cloud-Workload erwartet wird, z. B. Reaktionszeit, Verfügbarkeit und Datenspeicherung. Sie beeinflussen die Architektur, die Ressourcennutzung und die Umweltauswirkungen einer Cloud-Workload. Prüfen Sie in regelmäßigen Abständen SLAs und gehen Sie Kompromisse ein, die den Ressourcenverbrauch deutlich reduzieren und im Gegenzug akzeptable Verringerungen der Serviceniveaus ermöglichen.

Implementierungsschritte

- Nachhaltigkeitsziele verstehen: Identifizieren Sie Nachhaltigkeitsziele in Ihrer Organisation – etwa eine Reduzierung des CO2-Ausstoßes oder eine bessere Ressourcennutzung.
- ÜberprüfungSLAs: Evaluieren Sie Ihre ProdukteSLAs, um zu beurteilen, ob sie Ihren Geschäftsanforderungen entsprechen. Wenn Sie diese Werte überschreitenSLAs, führen Sie eine weitere Überprüfung durch.
- Kompromisse verstehen: Machen Sie sich ein Bild von den Kompromissen zwischen der Komplexität (zum Beispiel hohe Anzahl gleichzeitiger Benutzer), der Leistung (zum Beispiel Latenz) und den Auswirkungen auf die Nachhaltigkeit Ihrer Workloads (zum Beispiel Ressourcenbedarf). In der Regel geht die Priorisierung von zwei der Faktoren auf Kosten des dritten.
- AnpassungSLAs: Passen Sie Ihre Situation an, SLAs indem Sie Kompromisse eingehen, die die Auswirkungen auf die Nachhaltigkeit deutlich reduzieren, und im Gegenzug akzeptable Verringerungen des Serviceniveaus vornehmen.
 - Nachhaltigkeit und Zuverlässigkeit: Workloads mit hoher Verfügbarkeit verbrauchen in der Regel mehr Ressourcen.
 - Nachhaltigkeit und Leistung: Die Nutzung von mehr Ressourcen, um die Leistung zu steigern, führt unter Umständen zu einer höheren Umweltbelastung.
 - Nachhaltigkeit und Sicherheit: Übermäßig sichere Workloads haben möglicherweise eine höhere Umweltbelastung zur Folge.
- Definieren Sie Nachhaltigkeit, SLAs wenn möglich: Beziehen Sie Nachhaltigkeit in Ihren Workload mit SLAs ein. Definieren Sie beispielsweise ein Mindestauslastungsniveau als Nachhaltigkeit SLA für Ihre Recheninstanzen.

- Verwenden Sie effiziente Entwurfsmuster: Verwenden Sie Entwurfsmuster wie Microservices AWS , die geschäftskritischen Funktionen Priorität einräumen und niedrigere Service-Levels (wie Reaktionszeit- oder Wiederherstellungszeitziele) für unkritische Funktionen ermöglichen.
- Kommunizieren Sie und legen Sie Rechenschaftspflichten fest: Teilen Sie diese Informationen SLAs mit allen relevanten Stakeholdern, einschließlich Ihrem Entwicklungsteam und Ihren Kunden. Verwenden Sie die Berichterstattung, um die zu verfolgen und zu überwachen. SLAs Weisen Sie Rechenschaftspflicht zu, um die Nachhaltigkeitsziele für Sie SLAs zu erreichen.
- Nutzen Sie Anreize und Prämien: Nutzen Sie Anreize und Prämien, um Nachhaltigkeitsziele zu erreichen oder zu übertreffenSLAs.
- Überprüfen und wiederholen: Überprüfen Sie Ihre Ziele regelmäßig und passen Sie sie anSLAs, um sicherzustellen, dass sie mit den sich entwickelnden Nachhaltigkeits- und Leistungszielen in Einklang stehen.

Ressourcen

Zugehörige Dokumente:

- [Grundlegendes zu Resilienzmustern und Kompromissen, um eine effiziente Architektur in der Cloud zu entwickeln](#)
- [Bedeutung von Service Level Agreements für SaaS-Anbieter](#)

Zugehörige Videos:

- [AWS re:Invent 2023 — Kapazität, Verfügbarkeit, Kosteneffizienz: Wählen Sie drei](#)
- [AWS re:INVENT 2023 — Nachhaltige Architektur: Vergangenheit, Gegenwart und future](#)
- [AWS re:Invent 2023 — Fortschrittliche Integrationsmuster und Kompromisse für lose gekoppelte Systeme](#)
- [AWS re:Invent 2022 — Bereitstellung nachhaltiger, leistungsstarker Architekturen](#)
- [AWS re:Invent 2022 — Schaffen Sie eine kosten-, energie- und ressourceneffiziente Computerumgebung](#)

SUS02-BP03 Beenden der Erstellung und Wartung nicht verwendeter Komponenten

Nehmen Sie nicht verwendete Ressourcen in Ihrer Workload außer Betrieb, um die Anzahl der Cloud-Ressourcen zu verringern, die zur Unterstützung Ihres Bedarfs und zur Minimierung von Verschwendung erforderlich sind.

Typische Anti-Muster:

- Sie analysieren Ihre Anwendung nicht auf Ressourcen, die redundant sind oder nicht mehr benötigt werden.
- Sie entfernen keine redundanten oder nicht mehr benötigten Ressourcen.

Vorteile der Nutzung dieser bewährten Methode: Das Entfernen nicht genutzter Ressourcen setzt Kapazitäten frei und verbessert die allgemeine Effizienz der Workload.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Niedrig

Implementierungsleitfaden

Nicht verwendete Ressourcen verbrauchen Cloud-Kapazitäten wie Speicherplatz oder Rechenleistung. Wenn Sie solche Ressourcen identifizieren und eliminieren, können Sie diese Kapazitäten freisetzen, was zu einer effizienteren Cloud-Architektur führt. Analysieren Sie Anwendungsressourcen (wie vorab kompilierte Berichte, Datensätze, statische Bilder) sowie Zugriffsmuster für Komponenten, um Redundanzen, eine zu geringe Auslastung und mögliche Kandidaten für die Außerbetriebnahme zu identifizieren. Entfernen Sie diese redundanten Ressourcen, um die Ressourcenverschwendung in Ihrer Workload zu reduzieren.

Implementierungsschritte

- Bestandsaufnahme durchführen: Führen Sie eine umfassende Bestandsaufnahme durch, um alle Komponenten innerhalb Ihrer Workload zu identifizieren.
- Nutzung analysieren: Verwenden Sie die kontinuierliche Überwachung, um statische Komponenten zu identifizieren, die nicht mehr benötigt werden.
- Ungenutzte Komponenten entfernen: Entwickeln Sie einen Plan, um Komponenten zu entfernen, die nicht mehr benötigt werden.
 - Prüfen Sie vor dem Entfernen einer Ressource die Auswirkungen dieser Maßnahme auf die Architektur.
 - Konsolidieren Sie sich überschneidende generierte Komponenten, um eine redundante Verarbeitung zu entfernen.

- Aktualisieren Sie Ihre Anwendungen, damit diese nicht mehr benötigte Ressourcen nicht weiter produzieren und speichern.
- Mit Dritten kommunizieren: Weisen Sie Dritte an, die Erstellung und Speicherung von Komponenten einzustellen, die in Ihrem Auftrag verwaltet und nicht mehr benötigt werden. Bitten Sie darum, dass redundante Komponenten konsolidiert werden.
- Lebenszyklusrichtlinien verwenden: Verwenden Sie Lebenszyklusrichtlinien, damit ungenutzte Komponenten automatisch gelöscht werden.
 - Mit [Amazon-S3-Lebenszyklen](#) können Sie Ihre Objekte während ihres gesamten Lebenszyklus verwalten.
 - Mit [Amazon Data Lifecycle Manager](#) lassen sich das Erstellen, Aufbewahren und Löschen von Amazon-EBS-Snapshots und Amazon-EBS-gestützten AMIs automatisieren.
- Überprüfen und optimieren: Überprüfen Sie regelmäßig Ihre Workload, um ungenutzte Komponenten zu identifizieren und zu entfernen.

Ressourcen

Zugehörige Dokumente:

- [Optimizing your AWS Infrastructure for Sustainability, Part II: Storage](#)
- [How do I terminate active resources that I no longer need on my AWS-Konto?](#)

Zugehörige Videos:

- [AWS re:Invent 2.023 - Sustainable architecture: Past, present, and future](#)
- [AWS re:Invent 2.022 - Preserving and maximizing the value of digital media assets using Amazon S3](#)
- [AWS re:Invent 2.023 - Optimize costs in your multi-account environments](#)

SUS02-BP04 Optimieren Sie die geografische Verteilung von Workloads auf der Grundlage ihrer Netzwerkanforderungen

Wählen Sie Cloud-Standorte und -Services für Ihre Workload, die die Entfernungen reduzieren, über die Netzwerkdatenverkehr übertragen werden muss, um die Zahl der Netzwerkressourcen zu verringern, die zur Unterstützung Ihrer Workload erforderlich sind.

Typische Anti-Muster:

- Sie wählen die Region der Workload auf der Grundlage Ihres eigenen Standorts aus.
- Sie konsolidieren alle Workload-Ressourcen an einem geografischen Standort.
- Der gesamte Datenverkehr fließt durch Ihre bestehenden Rechenzentren.

Vorteile der Nutzung dieser bewährten Methode: Die Platzierung von Workloads in der Nähe der Benutzer bietet die geringstmögliche Latenz und verringert gleichzeitig die Bewegung der Daten durch das Netzwerk und damit die Umweltauswirkungen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Die AWS Cloud Infrastruktur basiert auf Standortoptionen wie Regionen, Availability Zones, Platzierungsgruppen und Edge-Standorten wie [AWS Outposts](#) [AWS Local Zones](#). Diese Standortoptionen stellen die Konnektivität zwischen Anwendungskomponenten, Cloud-Services, Edge-Netzwerken und On-Premises-Rechenzentren sicher.

Analysieren Sie die Netzwerkzugriffsmuster in Ihrer Workload, um festzustellen, wie diese verwendet werden können, um die Entfernungen für den Netzwerkdatenverkehr zu reduzieren.

Implementierungsschritte

- Analysieren Sie die Netzwerkzugriffsmuster in Ihrer Workload, um zu ermitteln, wie die Benutzer Ihre Anwendung verwenden.
 - Verwenden Sie Überwachungstools wie [Amazon CloudWatch](#) und [AWS CloudTrail](#), um Daten zu Netzwerkaktivitäten zu sammeln.
 - Analysen Sie die Daten, um das Netzwerkzugriffsmuster zu identifizieren.
- Wählen Sie die Regionen für Ihre Workload-Bereitstellung auf der Grundlage der folgenden zentralen Elemente aus:
 - Ihr Nachhaltigkeitsziel: wie unter [Regionsauswahl](#) erklärt.
 - Standort Ihrer Daten: Für datenintensive Anwendungen (wie etwa Big Data oder Machine Learning) sollte der Anwendungscode so nahe wie möglich zu den Daten ausgeführt werden.
 - Standort Ihrer Benutzer: Wählen Sie für benutzerseitige Anwendungen eine Region (oder Regionen) in der Nähe der Benutzer der Workload.
 - Weitere Einschränkungen: Berücksichtigen Sie Einschränkungen wie Kosten und Compliance, wie unter [Relevante Aspekte bei der Wahl einer Region für Ihre Workloads](#) erläutert.

- Verwenden Sie lokale Zwischenspeicherung oder [AWS -Zwischenspeicherung](#) für häufig genutzte Ressourcen zur Verbesserung der Leistung, zur Verringerung von Datenverschiebungen und zur Reduzierung der Umweltauswirkungen.

Service	Wann sollte dies verwendet werden?
Amazon CloudFront	Wird verwendet, um statische Inhalte wie Bilder, Skripte und Videos sowie dynamische Inhalte wie API Antworten oder Webanwendungen zwischenzuspeichern.
Amazon ElastiCache	Verwenden Sie dies für die Zwischenspeicherung von Inhalten für Webanwendungen.
DynamoDB Accelerator	Verwenden Sie dies für die Add-in-Speicher-Beschleunigung für Ihre DynamoDB-Tabellen.

- Nutzen Sie Services, die Ihnen dabei helfen können, Code näher an den Benutzern Ihrer Workload auszuführen:

Service	Wann sollte dies verwendet werden?
Lambda@Edge	Verwenden Sie dies für rechenintensive Anwendungen, die initiiert werden, wenn sich Objekte nicht im Zwischenspeicher befinden.
CloudFront Amazon-Funktionen	Wird für einfache Anwendungsfälle wie HTTP Bearbeitungen von Anfragen oder Antworten verwendet, die durch kurzlebige Funktionen ausgelöst werden können.
AWS IoT Greengrass	Verwenden Sie dies für die Ausführung lokaler Rechenoperationen, Messaging sowie die Datenzwischenspeicherung für verbundene Geräte.

- Nutzen Sie Verbindungspooling, um die erneute Nutzung von Verbindungen zu ermöglichen und die Zahl der erforderlichen Ressourcen zu reduzieren.
- Verwenden Sie verteilte Datenspeicher, die nicht auf persistente Verbindungen und synchrone Updates angewiesen sind, um regionale Benutzergruppen zu unterstützen.
- Ersetzen Sie vorab bereitgestellte statische Netzwerkkapazität durch geteilte dynamische Kapazitäten und teilen Sie die Auswirkungen von Netzwerkkapazitäten auf die Nachhaltigkeit mit anderen Abonnenten.

Ressourcen

Zugehörige Dokumente:

- [Optimierung Ihrer AWS Infrastruktur im Hinblick auf Nachhaltigkeit, Teil: Netzwerke III](#)
- [ElastiCache Amazon-Dokumentation](#)
- [Was ist Amazon CloudFront?](#)
- [Die CloudFront wichtigsten Funktionen von Amazon](#)
- [AWS Globale Infrastruktur](#)
- [AWS Local Zones und AWS Outposts Auswahl der richtigen Technologie für Ihren Edge-Workload](#)
- [Platzierungsgruppen](#)
- [AWS Local Zones](#)
- [AWS Outposts](#)

Zugehörige Videos:

- [Entmystifizierung der Datenübertragung am AWS](#)
- [Skalierung der Netzwerkleistung auf EC2 Amazon-Instances der nächsten Generation](#)
- [AWS Erklärvideo zu Local Zones](#)
- [AWS Outposts: Overview and How it Works](#)
- [AWS re:Invent 2023 — Eine Migrationsstrategie für Edge-Workloads und lokale Workloads](#)
- [AWS re:Invent 2021 — AWS Outposts: Das Erlebnis vor Ort umsetzen AWS](#)
- [AWS re:Invent 2020 — AWS Wavelength: Führen Sie Apps mit extrem niedriger Latenz am 5G-Edge aus](#)
- [AWS re:Invent 2022 — AWS Local Zones: Entwicklung von Anwendungen für eine verteilte Kante](#)

- [AWS re:Invent 2021 — Websites mit niedriger Latenz mit Amazon erstellen CloudFront](#)
- [AWS re:Invent 2022 — Verbessern Sie Leistung und Verfügbarkeit mit AWS Global Accelerator](#)
- [AWS re:Invent 2022 — Bauen Sie Ihr globales Wide Area Network auf mit AWS](#)
- [AWS re:Invent 2020: Globales Verkehrsmanagement mit Amazon Route 53](#)

Zugehörige Beispiele:

- [AWS Netzwerk-Workshops](#)
- [Architecting for sustainability - Minimize data movement across networks](#)

SUS02-BP05 Optimieren von Ressourcen für Teammitglieder im Hinblick auf die ausgeführten Aktivitäten

Optimieren Sie die Ressourcen, die Teammitgliedern zur Verfügung gestellt werden, um negative Auswirkungen auf die Nachhaltigkeit zu minimieren und gleichzeitig ihre Anforderungen zu erfüllen.

Typische Anti-Muster:

- Sie berücksichtigen nicht die Auswirkungen der von Ihren Teammitgliedern verwendeten Geräte auf die Gesamteffizienz Ihrer Cloud-Anwendung.
- Sie verwalten und aktualisieren die von Teammitgliedern verwendeten Ressourcen manuell.

Vorteile der Nutzung dieser bewährten Methode: Die Optimierung der Teammitglieder-Ressourcen verbessert die allgemeine Effizienz Cloud-fähiger Anwendungen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Niedrig

Implementierungsleitfaden

Verstehen Sie die Ressourcen, mit denen Ihre Teammitglieder Ihre Services nutzen, deren erwartete Lebensdauer sowie die finanziellen und nachhaltigkeitsbezogenen Auswirkungen. Implementieren Sie Strategien zur Optimierung dieser Ressourcen. Beispielsweise können Sie komplexe Vorgänge wie Rendering und Kompilierung auf intensiv genutzter und skalierbarer Infrastruktur anstatt auf weniger ausgelasteten Einzelbenutzersystemen mit hohem Energieverbrauch ausführen.

Implementierungsschritte

- Energieeffiziente Workstations verwenden: Stellen Sie den Teammitgliedern energieeffiziente Workstations und Peripheriegeräte zur Verfügung. Verwenden Sie effiziente Energiemanagementfeatures (wie den Energiesparmodus) auf diesen Geräten, um ihren Energieverbrauch zu reduzieren.
- Virtualisierung verwenden: Verwenden Sie virtuelle Desktops und Anwendungs-Streaming, um Upgrade- und Geräteanforderungen zu begrenzen.
- Remote-Zusammenarbeit fördern: Ermutigen Sie die Teammitglieder, Tools für die Remote-Zusammenarbeit wie [Amazon Chime](#) oder [AWS Wickr](#) zu verwenden, um den Reisebedarf und die damit verbundenen CO₂-Emissionen zu reduzieren.
- Energieeffiziente Software verwenden: Stellen Sie den Teammitgliedern energieeffiziente Software zur Verfügung, indem Sie nicht benötigte Features und Prozesse entfernen oder deaktivieren.
- Lebenszyklen verwalten: Evaluieren Sie die Auswirkungen von Prozessen und Systemen auf die Lebenszyklen von Geräten. Wählen Sie Lösungen aus, die den Bedarf für Geräte austauschvorgänge minimieren und gleichzeitig die geschäftlichen Anforderungen erfüllen. Pflegen und aktualisieren Sie regelmäßig Workstations oder Software, um die Effizienz aufrechtzuerhalten und zu verbessern.
- Remote-Verwaltung für Geräte: Implementieren Sie die Remote-Verwaltung für Geräte, um die Anzahl der erforderlichen Geschäftsreisen zu reduzieren.
 - [AWS Systems Manager Fleet Manager](#) ist eine vereinheitlichte UI-Umgebung, mit der Sie Ihre in AWS oder On-Premises ausgeführten Knoten aus der Ferne überwachen können.

Ressourcen

Zugehörige Dokumente:

- [What is Amazon WorkSpaces?](#)
- [Cost Optimizer für Amazon WorkSpaces](#)
- [Dokumentation zu Amazon AppStream 2.0](#)
- [NICE DCV](#)

Zugehörige Videos:

- [Managing cost for Amazon WorkSpaces on AWS](#)

SUS02-BP06 Implementieren Sie Pufferung oder Drosselung, um die Nachfragekurve abzuflachen

Pufferung und Drosselung verflachen die Bedarfskurve und reduzieren die erforderliche bereitgestellte Kapazität für Ihre Workload.

Typische Anti-Muster:

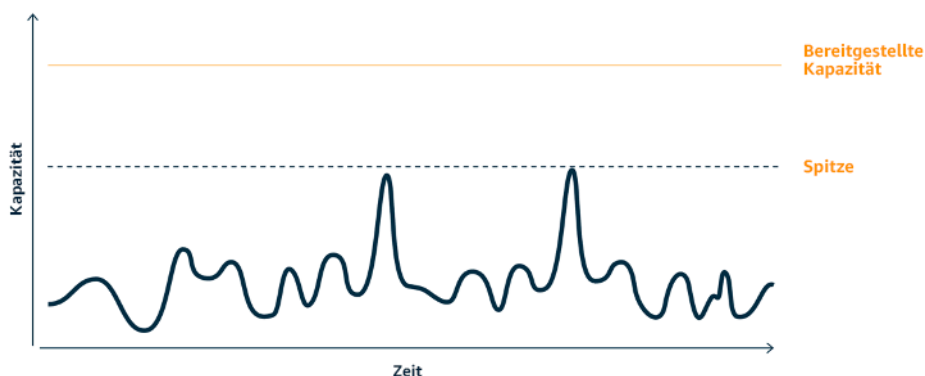
- Sie verarbeiten die Client-Anfragen sofort, obwohl dies nicht erforderlich ist.
- Sie analysieren die Anforderungen für Client-Anfragen nicht.

Vorteile der Nutzung dieser bewährten Methode: Das Verflachen der Bedarfskurve reduziert die erforderliche bereitgestellte Kapazität für die Workload. Die Reduzierung der bereitgestellten Kapazität bedeutet geringeren Energieverbrauch und geringere Umweltauswirkungen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Niedrig

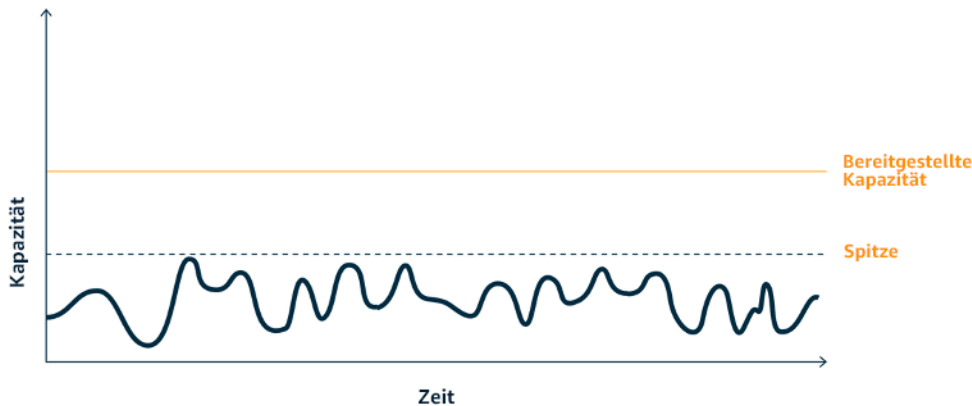
Implementierungsleitfaden

Die Verflachung der Bedarfskurve kann Ihnen dabei helfen, die bereitgestellte Kapazität für eine Workload zu verringern und dessen Umweltauswirkungen zu reduzieren. Nehmen wir eine Workload mit der nachfolgend gezeigten Bedarfskurve. Diese Workload hat zwei Spitzen und um damit umzugehen, wird die Ressourcenkapazität bereitgestellt, die hier durch die orangefarbene Linie angezeigt wird. Die für diese Workload aufgewendeten Ressourcen und die eingesetzte Energie werden nicht durch die Fläche unter der Bedarfskurve, sondern von der Linie für die bereitgestellte Kapazität angezeigt, da für den Umgang mit den beiden Spitzen bereitgestellte Kapazität erforderlich ist.



Bedarfskurve mit zwei deutlichen Spitzen, die hohe bereitgestellte Kapazität erfordern

Sie können Pufferung oder Drosselung verwenden, um die Bedarfskurve zu beeinflussen und die Spitzen abzumildern, was weniger bereitgestellte Kapazität und einen geringeren Energieverbrauch bedeutet. Implementieren Sie Drosselung, wenn Ihre Clients wiederholte Versuche durchführen können. Implementieren Sie die Pufferung, um die Anforderung zu speichern und die Verarbeitung auf einen späteren Zeitpunkt zu verschieben.



Auswirkung der Drosselung auf die Nachfragekurve und die bereitgestellte Kapazität

Implementierungsschritte

- Analysieren Sie die Client-Anfragen, um festzulegen, wie darauf zu reagieren ist. Wichtige Faktoren dabei sind:
 - Kann diese Anfrage in asynchroner Weise verarbeitet werden?
 - Kann der Client die Anfrage erneut versuchen?
- Wenn dies der Fall ist, können Sie Drosselung verwenden, die der Quelle mitteilt, dass wenn sie die Anfrage zum aktuellen Zeitpunkt nicht bedienen kann, es später erneut versucht werden sollte.
 - Sie können [Amazon API Gateway](#) verwenden, um Drosselung zu implementieren.
- Für Clients, die Anfragen nicht erneut versuchen können, muss zur Verflachung der Bedarfskurve ein Puffer implementiert werden. Ein Puffer verschiebt die Anforderungsverarbeitung, so dass Anwendungen, die mit unterschiedlichen Raten ausgeführt werden, effektiv kommunizieren können. Bei der Pufferung werden Nachrichten von Produzenten in eine Warteschlange oder einen Stream gestellt. Nachrichten können dadurch von Verbrauchern in der für ihre Geschäftsanforderungen passenden Geschwindigkeit gelesen und verarbeitet werden.
 - [Amazon Simple Queue Service \(AmazonSQS\)](#) ist ein verwalteter Service, der Warteschlangen bereitstellt, die es einem einzelnen Verbraucher ermöglichen, einzelne Nachrichten zu lesen.

- [Amazon Kinesis](#) stellt einen Stream bereit, mit dem viele Verbraucher dieselben Nachrichten lesen können.
- Analysieren Sie den Gesamtbedarf, die Änderungsrate und die erforderliche Reaktionszeit, um die korrekte Größe der erforderlichen Drosselung oder des Puffers zu bestimmen.

Ressourcen

Zugehörige Dokumente:

- [Erste Schritte mit Amazon SQS](#)
- [Application integration Using Queues and Messages](#)
- [Verwaltung und Überwachung der API Drosselung Ihrer Workloads](#)
- [Drosselung einer mehrstufigen Lösung mit mehreren Mandanten im großen Maßstab mithilfe von Gateway REST API](#)
- [Application integration Using Queues and Messages](#)

Zugehörige Videos:

- [AWS re:Invent 2022 — Anwendungsintegrationsmuster für Microservices](#)
- [AWS re:Invent 2023 — Intelligentes Sparen: Strategien zur Kostenoptimierung von Amazon EC2](#)
- [AWS re:Invent 2023 — Fortschrittliche Integrationsmuster und Kompromisse für lose gekoppelte Systeme](#)

Software und Architektur

Frage

- [SUS 3 Wie können Sie Software- und Architekturmuster zur Unterstützung Ihrer Nachhaltigkeitsziele nutzen?](#)

SUS 3 Wie können Sie Software- und Architekturmuster zur Unterstützung Ihrer Nachhaltigkeitsziele nutzen?

Implementieren Sie Muster für den Lastausgleich und die Wahrung einer konsistent hohen Nutzung der bereitgestellten Ressourcen, um die Zahl der genutzten Ressourcen zu minimieren. Komponenten werden möglicherweise aufgrund von Änderungen des Benutzerverhaltens

über die Zeit nicht mehr genutzt. Prüfen Sie Muster und Architekturen, um nicht ausreichend genutzte Komponenten zu konsolidieren und so die Nutzung insgesamt zu erhöhen. Nehmen Sie Komponenten außer Betrieb, die nicht mehr benötigt werden. Identifizieren Sie die Leistung Ihrer Workload-Komponenten und optimieren Sie die Komponenten, die die meisten Ressourcen verbrauchen. Achten Sie auf die Geräte, mit denen Ihre Kunden auf Ihre Services zugreifen, und implementieren Sie Muster, um den Bedarf für Geräte-Upgrades zu minimieren.

Best Practices

- [SUS03-BP01 Optimieren von Software und Architektur für asynchrone und geplante Aufträge](#)
- [SUS03-BP02 Entfernen oder Faktorwechsel von Workload-Komponenten mit geringer oder keiner Nutzung](#)
- [SUS03-BP03 Optimieren von Codebereichen, die die meiste Zeit oder die meisten Ressourcen verbrauchen](#)
- [SUS03-BP04 Optimieren der Auswirkungen auf Geräte und Ausrüstung von Kunden](#)
- [SUS03-BP05 Verwenden von Softwaremustern und Architekturen, die Datenzugriffs- und Speichermuster optimal unterstützen](#)

SUS03-BP01 Optimieren von Software und Architektur für asynchrone und geplante Aufträge

Verwenden Sie effiziente Software- und Architekturmuster wie Warteschlangenbasierte Systeme, um eine durchgängig hohe Auslastung von bereitgestellten Ressourcen zu erzielen.

Typische Anti-Muster:

- Sie stellen zu viele Ressourcen in der Cloud-Workload bereit, um auf unerwartete Nachfragesteigerungen reagieren zu können.
- In Ihrer Architektur werden Absender und Empfänger von asynchronen Nachrichten nicht durch eine Messaging-Komponente entkoppelt.

Vorteile der Nutzung dieser bewährten Methode:

- Durch effiziente Software- und Architekturmuster werden ungenutzte Ressourcen in Ihrer Workload minimiert und die allgemeine Effizienz gesteigert.
- Sie können die Verarbeitung unabhängig vom Empfang asynchroner Nachrichten skalieren.
- Durch eine Messaging-Komponente gelten weniger strenge Verfügbarkeitsanforderungen, die mit weniger Ressourcen erfüllt werden können.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Verwenden Sie effiziente Architekturmuster wie eine [ereignisgesteuerte Architektur](#), die zu einer gleichmäßigen Nutzung der Komponenten führen und die Überbereitstellung in Ihrer Workload minimieren. Durch die Verwendung effizienter Architekturmuster werden ungenutzte Ressourcen, die aufgrund von Änderungen der Nachfrage im Laufe der Zeit nicht genutzt werden, minimiert.

Analysieren Sie die Anforderungen Ihrer Workload-Komponenten und führen Sie Architekturmuster ein, mit denen die allgemeine Auslastung der Ressourcen gesteigert wird. Nehmen Sie Komponenten außer Betrieb, die nicht mehr benötigt werden.

Implementierungsschritte

- Analysieren Sie die Nachfrage für Ihre Workload, um zu bestimmen, wie diese erfüllt werden kann.
- Verwenden Sie für Anfragen oder Aufträge, für die keine synchronen Antworten erforderlich sind, warteschlangenbasierte Architekturen und Worker mit Auto Scaling, durch die die Auslastung maximiert wird. Hier finden Sie einige Beispiele für Situationen, in denen Sie eine warteschlangenbasierte Architektur in Erwägung ziehen sollten:

Warteschlangenmechanismus	Beschreibung
AWS Batch-Auftragswarteschlangen	AWS Batch-Aufträge werden an eine Auftragswarteschlange gesendet, in der sie bleiben, bis ihre Ausführung in einer Datenverarbeitungsumgebung geplant werden kann.
Amazon Simple Queue Service und Amazon EC2 Spot Instances	Durch das Koppeln von Amazon SQS und Spot Instances lassen sich fehlertolerante und effiziente Architekturen erstellen.

- Verwenden Sie für Anfragen oder Aufträge, die jederzeit verarbeitet werden können, Planungsmechanismen zur Auftragsverarbeitung in Batches, um die Effizienz zu steigern. Hier sind einige Beispiele für Planungsmechanismen in AWS:

Zeitplanungsmechanismus	Beschreibung
Amazon EventBridge Scheduler	Eine Funktion von Amazon EventBridge , mit der Sie in großem Umfang geplante Aufgaben erstellen, ausführen und verwalten können.
Zeitbasierter AWS Glue-Plan	Hiermit definieren Sie einen zeitbasierten Plan für Crawler und Aufträge in AWS Glue.
Geplante Aufgaben von Amazon Elastic Container Service (Amazon ECS)	Amazon ECS unterstützt das Erstellen geplanter Tasks. Geplante Tasks verwenden Amazon EventBridge Regeln, um Tasks entweder nach einem Zeitplan oder als Antwort auf ein EventBridge-Ereignis auszuführen.
Instance Scheduler	Konfigurieren Sie Zeitpläne zum Starten und Beenden Ihrer Instances von Amazon EC2 und Amazon Relational Database Service.

- Wenn Sie Abfrage- und Webhook-Mechanismen in Ihrer Architektur verwenden, ersetzen Sie diese durch Ereignisse. Erstellen Sie mit [ereignisgesteuerten Architekturen](#) hocheffiziente Workloads.
- Nutzen Sie [Serverless in AWS](#), um eine übermäßige Bereitstellung in einer Infrastruktur zu eliminieren.
- Wählen Sie die richtige Größe für Ihre Architektur, um zu vermeiden, dass ungenutzte Ressourcen auf Eingaben warten.
 - Sie können die [Empfehlungen zur Dimensionierung in AWS Cost Explorer](#) oder [AWS Compute Optimizer](#) zur Identifizierung von Dimensionierungsmöglichkeiten verwenden.
- Weitere Informationen finden Sie unter [Größenanpassung: Bereitstellung von an Workloads angepassten Instances](#).

Ressourcen

Zugehörige Dokumente:

- [What is Amazon Simple Queue Service?](#)

- [What is Amazon MQ?](#)
- [Scaling based on Amazon SQS](#)
- [Was ist AWS Step Functions?](#)
- [Was ist AWS Lambda?](#)
- [Using AWS Lambda with Amazon SQS](#)
- [What is Amazon EventBridge?](#)
- [Managing Asynchronous Workflows with a REST API](#)

Zugehörige Videos:

- [AWS re:Invent 2.023 - Navigating the journey to serverless event-driven architecture](#)
- [AWS re:Invent 2.023 - Using serverless for event-driven architecture & domain-driven design](#)
- [AWS re:Invent 2.023 - Advanced event-driven patterns with Amazon EventBridge](#)
- [AWS re:Invent 2.023 - Sustainable architecture: Past, present, and future](#)
- [Asynchronous Message Patterns | AWS Events](#)

Zugehörige Beispiele:

- [Event-driven architecture with AWS Graviton Processors and Amazon EC2 Spot Instances](#)

SUS03-BP02 Entfernen oder Faktorwechsel von Workload-Komponenten mit geringer oder keiner Nutzung

Entfernen Sie ungenutzte Komponenten, die nicht mehr benötigt werden, und führen Sie einen Faktorwechsel für Komponenten mit geringer Nutzung durch, um die Verschwendung von Ressourcen in Ihrer Workload zu begrenzen.

Typische Anti-Muster:

- Sie prüfen den Nutzungsgrad der einzelnen Komponenten Ihrer Workload nicht regelmäßig.
- Sie prüfen und analysieren nicht die Empfehlungen von AWS-Dimensionierungstools wie etwa [AWS Compute Optimizer](#).

Vorteile der Nutzung dieser bewährten Methode: Das Entfernen nicht genutzter Komponenten minimiert Ausschuss und verbessert die allgemeine Effizienz Ihrer Workload.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Ungenutzte oder nur wenig genutzte Komponenten in einem Cloud-Workload verbrauchen unnötigerweise Rechen-, Speicher- oder Netzwerkressourcen. Entfernen Sie diese Komponenten oder führen Sie einen Faktorwechsel für diese Ressourcen durch, um die Verschwendung direkt zu reduzieren und die allgemeine Effizienz eines Cloud-Workloads zu verbessern. Dies ist ein sich wiederholender Verbesserungsprozess, der von Änderungen beim Bedarf oder der Einführung eines neuen Cloud-Services ausgelöst werden kann. Beispielsweise kann eine deutliche Reduzierung der Laufzeit der [AWS Lambda](#)-Funktion darauf hindeuten, dass die Arbeitsspeichergöße reduziert werden sollte. Außerdem können sich die optimalen Services und die Architektur für Ihre Workload ändern, wenn AWS neue Services und Features veröffentlicht.

Überwachen Sie kontinuierlich die Workload-Aktivität und suchen Sie nach Möglichkeiten zur Verbesserung des Nutzungsgrads einzelner Komponenten. Wenn Sie nicht genutzte Komponenten entfernen und Dimensionierungsaktivitäten durchführen, erreichen Sie Ihre geschäftlichen Ziele mit der geringstmöglichen Menge von Cloud-Ressourcen.

Implementierungsschritte

- Inventarisieren Sie Ihre AWS-Ressourcen: Erstellen Sie eine Bestandsliste Ihrer AWS-Ressourcen. In AWS können Sie [AWS Resource Explorer](#) einschalten, um Ihre AWS-Ressourcen zu erkunden und zu organisieren. Weitere Informationen finden Sie unter [AWS re:Invent 2.022 - How to manage resources and applications at scale on AWS](#).
- Überwachen der Nutzung: Überwachen und erfassen Sie die Nutzungsmetriken für kritische Komponenten Ihres Workloads (z. B. CPU-Nutzung, Arbeitsspeichernutzung oder Netzwerkdurchsatz in [Amazon-CloudWatch-Metriken](#)).
- Identifizieren ungenutzter Komponenten: Identifizieren Sie ungenutzte oder nur wenig genutzte Komponenten in Ihrer Architektur.
 - Prüfen Sie für stabile Workloads regelmäßig AWS-Dimensionierungstools wie [AWS Compute Optimizer](#), um nicht oder wenig genutzte Komponenten zu identifizieren.
 - Prüfen Sie für kurzzeitige Workloads die Nutzungsmetriken, um nicht oder wenig genutzte Komponenten zu identifizieren.
- Entfernen ungenutzter Komponenten: Nehmen Sie nicht mehr benötigte und zugehörige Ressourcen (z. B. Amazon-ECR-Images) außer Betrieb.
 - [Automatisierte Bereinigung von nicht verwendeten Images in Amazon ECR](#)

- [Löschen von ungenutzten Amazon Elastic Block Store \(Amazon EBS\)-Volumes mit AWS Config und AWS Systems Manager](#)
- Führen Sie einen Faktorwechsel für wenig genutzte Komponenten aus: Führen Sie einen Faktorwechsel für wenig genutzte Ressourcen aus oder konsolidieren Sie diese mit anderen Ressourcen, um die Nutzungseffizienz zu verbessern. Sie können beispielsweise mehrere kleine Datenbanken auf einer einzelnen Datenbank-Instance in [Amazon RDS](#) bereitstellen, anstatt Datenbanken auf einzelnen, wenig genutzten Instances auszuführen.
- Evaluieren der Verbesserungen: Informieren Sie sich über die [Ressourcen, die von Ihrem Workload bereitgestellt werden, um eine Arbeitseinheit zu erledigen](#). Mit diesen Informationen können Sie die Verbesserungen bewerten, die durch das Entfernen oder einen Faktorwechsel von Komponenten erzielt wurden.
 - [Messen und Verfolgen der Cloud-Effizienz mit Proxymetriken für Nachhaltigkeit, Teil I: Was sind Proxymetriken?](#)
 - [Messen und Verfolgen der Cloud-Effizienz mit Proxymetriken für Nachhaltigkeit, Teil II: Einrichten einer Metrik-Pipeline](#)

Ressourcen

Zugehörige Dokumente:

- [AWS Trusted Advisor](#)
- [Was ist Amazon CloudWatch?](#)
- [Größenanpassung: Bereitstellung von an Workloads angepassten Instances](#)
- [Optimizing your cost with Rightsizing Recommendations](#)

Zugehörige Videos:

- [AWS re:Invent 2023 – Kapazität, Verfügbarkeit, Kosteneffizienz: Wählen Sie drei Optionen aus](#)

SUS03-BP03 Optimieren von Codebereichen, die die meiste Zeit oder die meisten Ressourcen verbrauchen

Optimieren Sie den Code, der innerhalb der verschiedenen Komponenten Ihrer Architektur ausgeführt wird, um die Ressourcennutzung zu minimieren und die Leistung zu maximieren.

Typische Anti-Muster:

- Sie versäumen die Optimierung Ihres Codes für die Ressourcennutzung.
- Sie reagieren auf Leistungsprobleme normalerweise mit Erhöhung des Ressourceneinsatzes.
- Ihr Code-Prüfungs- und -Entwicklungsprozess verfolgt keine Leistungsänderungen.

Vorteile der Nutzung dieser bewährten Methode: Die Verwendung effizienten Codes minimiert die Ressourcennutzung und verbessert die Leistung.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Es ist sehr wichtig, jeden funktionalen Bereich, einschließlich des Codes einer für die Cloud erstellten Anwendung, zu untersuchen, um ihre Ressourcennutzung und Leistung zu optimieren. Überwachen Sie kontinuierlich die Leistung Ihrer Workload in Build-Umgebungen und Produktionsbereichen und suchen Sie nach Möglichkeiten, Codeausschnitte zu verbessern, die einen besonders hohen Ressourcenverbrauch haben. Führen Sie einen regelmäßigen Prüfungsprozess ein, um Fehler oder Anti-Muster in Ihrem Code zu identifizieren, die Ressourcen in ineffizienter Weise nutzen. Nutzen Sie einfache und effiziente Algorithmen, die dieselben Ergebnisse für Ihre Anwendungsfälle liefern.

Implementierungsschritte

- Effiziente Programmiersprache verwenden: Verwenden Sie das jeweils effizienteste Betriebssystem und die optimale Programmiersprache für die Workload. Weitere Informationen zu energieeffizienten Programmiersprachen (einschließlich Rust) finden Sie unter [Sustainability with Rust](#).
- KI-Programmierungsbegleiter verwenden: Ziehen Sie die Verwendung eines KI-Programmierungsbegleiters wie [Amazon Q Developer](#) in Betracht, um Code auf effiziente Weise zu schreiben.
- Code-Überprüfungen automatisieren: Führen Sie bei der Entwicklung Ihrer Workloads einen automatischen Code-Prüfungsprozess ein, um die Qualität zu verbessern sowie Fehler und Anti-Muster zu identifizieren.
 - [Automatisieren von Code-Reviews mit Amazon CodeGuru Reviewer](#)
 - [Erkennen von Parallelitätsfehlern mit Amazon CodeGuru](#)
 - [Verbessern der Codequalität für Python-Anwendungen mit Amazon CodeGuru](#)
- Code-Profiler verwenden: Verwenden Sie einen Code-Profiler für Code-Prüfungen, um die Codebereiche als Optimierungsziele zu identifizieren, die die meiste Zeit oder die meisten Ressourcen verwenden.

- [Reduzieren des CO₂-Fußabdrucks Ihrer Organisation mit Amazon CodeGuru Profiler](#)
- [Verständnis der Speichernutzung in Ihrer Java-Anwendung mit Amazon CodeGuru Profiler](#)
- [Verbessern des Kundenkomforts und Senken von Kosten mit Amazon CodeGuru Profiler](#)
- Überwachen und optimieren: Verwenden Sie Ressourcen für die kontinuierliche Überwachung, um Komponenten mit hohem Ressourcenbedarf oder suboptimaler Konfiguration zu identifizieren.
 - Ersetzen Sie rechenintensive Algorithmen durch einfachere und effizientere Versionen, die dieselben Ergebnisse liefern.
 - Entfernen Sie unnötigen Code und überflüssige Formatierungen.
- Code-Faktorwechsel oder -Transformation verwenden: Erkunden Sie die Möglichkeiten der [Amazon-Q-Codetransformation](#) für die Wartung und Aktualisierung von Anwendungen.
 - [Sprachversionen mit Amazon-Q-Codetransformation aktualisieren](#)
 - [AWSre:Invent 2023 – Automatisierung von App-Upgrades und -Wartung mithilfe von Amazon Q-Codetransformation](#)

Ressourcen

Zugehörige Dokumente:

- [What is Amazon CodeGuru Profiler?](#)
- [FPGA instances](#)
- [Die AWS-SDKs Tools zum Entwickeln in AWS](#)

Zugehörige Videos:

- [Improve Code Efficiency Using Amazon CodeGuru Profiler](#)
- [Automate Code Reviews and Application Performance Recommendations with Amazon CodeGuru](#)

SUS03-BP04 Optimieren der Auswirkungen auf Geräte und Ausrüstung von Kunden

Verstehen Sie die in Ihrer Architektur verwendeten Geräte und nutzen Sie Strategien, um ihre Nutzung zu reduzieren. Dies kann die Umweltauswirkungen Ihrer Cloud-Workload insgesamt verringern.

Typische Anti-Muster:

- Sie ignorieren die Umweltauswirkungen der Geräte, die Ihre Kunden verwenden.
- Sie verwalten und aktualisieren die von Kunden verwendeten Ressourcen manuell.

Vorteile der Nutzung dieser bewährten Methode: Die Implementierung von Softwaremustern und Features, die für Kundengeräte optimiert sind, können die Umweltauswirkungen von Cloud-Workloads insgesamt verringern.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Die Implementierung für Kundengeräte optimierter Softwaremuster und Features können die Umweltauswirkungen auf unterschiedliche Weise reduzieren:

- Die Implementierung neuer abwärtskompatibler Features kann die Anzahl der Hardwareaustauschvorgänge verringern.
- Die Optimierung einer Anwendung, so dass sie effizient auf Geräten ausgeführt werden kann, kann bei der Reduzierung des Energieverbrauchs helfen und die Batterielaufzeit verlängern (falls Batterien zum Einsatz kommen).
- Die Optimierung einer Anwendung für Geräte kann auch Datenübertragungen über das Netzwerk verringern.

Verstehen Sie die in Ihrer Architektur verwendeten Geräte, ihre erwartete Lebensdauer und die Auswirkungen des Austauschs dieser Komponenten. Implementieren Sie Softwaremuster und Features, die dabei helfen, den Energieverbrauch von Geräten zu senken, und den Austausch von Geräten sowie manuelle Upgrades durch Kunden seltener erforderlich machen.

Implementierungsschritte

- Bestandsaufnahme durchführen: Inventarisieren Sie die in ihrer Architektur verwendeten Geräte. Dabei kann es sich um Mobilgeräte, Tablets, IOT-Geräte, Smart Light- oder auch Smartgeräte in einer Fabrik handeln.
- Energieeffiziente Geräte verwenden: Erwägen Sie den Einsatz energieeffizienter Geräte in Ihrer Architektur. Verwenden Sie Energieverwaltungskonfigurationen auf Geräten, um in den Energiesparmodus zu wechseln, wenn sie nicht verwendet werden.
- Effiziente Anwendungen ausführen: Optimieren Sie die Anwendung, die auf den Geräten ausgeführt wird:

- Verwenden Sie Strategien wie die Ausführung von Aufgaben im Hintergrund, um den Energieverbrauch zu verringern.
- Berücksichtigen Sie beim Erstellen von Nutzlasten Netzwerkbandbreite und Latenz und implementieren Sie Funktionen, mit denen Ihre Anwendungen auch über Verbindungen mit geringer Bandbreite und hoher Latenz gut funktionieren.
- Wandeln Sie Nutzlasten und Dateien in von den Geräten benötigte optimierte Formate um. Sie können beispielsweise [Amazon Elastic Transcoder](#) oder [AWS Elemental MediaConvert](#) verwenden, um große, qualitativ hochwertige Digitalmediendateien in Formate umzuwandeln, die Benutzer auf Mobilgeräten abspielen können.
- Führen Sie rechenintensive Aktivitäten (z. B. das Rendern von Bildern) serverseitig aus oder nutzen Sie Anwendungs-Streaming, um den Benutzerkomfort auf älteren Geräten zu verbessern.
- Segmentieren und paginieren Sie Ausgaben, besonders für interaktive Sitzungen, um Nutzlasten zu verwalten und lokale Speicheranforderungen zu begrenzen.
- Anbieter einbeziehen: Arbeiten Sie mit Geräteanbietern zusammen, die nachhaltige Materialien verwenden und für Transparenz in ihren Lieferketten und Umweltzertifizierungen sorgen.
- Over-the-Air (OTA)-Updates verwenden: Verwenden Sie den automatisierten Over-the-Air (OTA)-Mechanismus, um Updates auf einem oder mehreren Geräten bereitzustellen.
- Mit einer [CI/CD-Pipeline](#) können Sie mobile Anwendungen aktualisieren.
- Mit [AWS IoT Device Management](#) können Sie verbundene Geräte in großem Umfang aus der Ferne verwalten.
- Verwaltete Gerätefarmen verwenden: Verwenden Sie zum Testen neuer Features und Updates verwaltete Gerätefarmen mit repräsentativen Sätzen von Hardwaregeräten, um den Umfang der unterstützten Geräte zu maximieren. Weitere Details finden Sie unter [SUS06-BP04 Verwenden von verwalteten Gerätefarmen für Tests](#).
- Kontinuierliche Überwachung und Verbesserung: Verfolgen Sie den Energieverbrauch von Geräten, um Verbesserungsmöglichkeiten zu identifizieren. Verwenden Sie neue Technologien oder bewährte Methoden, um die Umweltauswirkungen dieser Geräte zu verbessern.

Ressourcen

Zugehörige Dokumente:

- [Was ist AWS Device Farm?](#)
- [Dokumentation zu WorkSpaces-Anwendungen](#)

- [NICE DCV](#)
- [OTA-Tutorial zum Aktualisieren der Firmware auf Geräten, auf denen FreeRTOS ausgeführt wird](#)
- [Optimizing Your IoT Devices for Environmental Sustainability](#)

Zugehörige Videos:

- [AWS re:Invent 2.023 - Improve your mobile and web app quality using AWS Device Farm](#)

SUS03-BP05 Verwenden von Softwaremustern und Architekturen, die Datenzugriffs- und Speichermuster optimal unterstützen

Identifizieren Sie, wie Daten in Ihrer Workload verwendet, von Benutzern genutzt, übertragen und gespeichert werden. Verwenden Sie Softwaremuster und Architekturen, die den Datenzugriff und die Speicherung optimal unterstützen, um die zur Unterstützung der Workload erforderlichen Datenverarbeitungs-, Netzwerk- und Speicherressourcen zu reduzieren.

Typische Anti-Muster:

- Sie gehen davon aus, dass für alle Workloads ähnliche Datenspeicher- und Zugriffsmuster gelten.
- Sie verwenden nur eine Speicherebene, vorausgesetzt, dass alle Workloads in diese Ebene passen.
- Sie gehen davon aus, dass Datenzugriffsmuster im Laufe der Zeit konsistent bleiben.
- Ihre Architektur unterstützt potenzielle hohe Bursts beim Datenzugriff, was dazu führt, dass die Ressourcen die meiste Zeit ungenutzt bleiben.

Vorteile der Nutzung dieser bewährten Methode: Die Auswahl und Optimierung Ihrer Architektur auf der Grundlage von Datenzugriffs- und Speichermustern hilft bei der Reduzierung der Entwicklungskomplexität und der Steigerung der allgemeinen Nutzung. Das Verständnis, wann globale Tabellen, Datenpartitionen und Caching verwendet werden sollen, hilft Ihnen dabei, den Betriebsaufwand zu verringern und basierend auf Ihren Workload-Anforderungen zu skalieren.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Verwenden Sie zur Verbesserung der langfristigen Workload-Nachhaltigkeit Architekturmuster, die Datenzugriff- und Speichereigenschaften für Ihren Workload unterstützen. Diese Muster helfen

Ihnen, Daten effizient abzurufen und zu verarbeiten. Sie können beispielsweise eine [moderne Datenarchitektur in AWS](#) mit speziell entwickelten Services verwenden, die für Ihre spezifischen Analytiksanwendungsfälle optimiert sind. Diese Architekturmuster ermöglichen die effiziente Datenverarbeitung und verringern die Ressourcennutzung.

Implementierungsschritte

- Informationen zu Dateneigenschaften: Analysieren Sie die Eigenschaften ihrer Daten und die Zugriffsmuster, um die korrekte Konfiguration für Ihre Cloud-Ressourcen zu identifizieren. Zu den berücksichtigenden Schlüsselmerkmalen gehören:
 - Datentyp: strukturiert, semistrukturiert, unstrukturiert
 - Datenwachstum: begrenzt, unbegrenzt
 - Lebensdauer von Daten: anhaltend, flüchtig, vorübergehend
 - Zugriffsmuster: Lese- oder Schreibzugriff, Häufigkeit von Aktualisierungen, schwankend oder konsistent
- Verwenden optimaler Architekturmuster: Verwenden Sie Architekturmuster, die Datenzugriffs- und Speichermuster optimal unterstützen.
 - [Muster zur Aktivierung der Datenpersistenz](#)
 - [Let's Architect! Moderne Datenarchitekturen](#)
 - [Datenbanken in AWS: Das richtige Tool für jede Aufgabe](#)
- Verwenden speziell entwickelter Services: Verwenden Sie Technologien, die für den jeweiligen Zweck geeignet sind.
 - Nutzen Sie Technologien, die nativ mit komprimierten Daten funktionieren.
 - [Dateiformate von Athena Compression Support](#)
 - [Mögliche Formate für ETL-Eingaben und -Ausgaben in AWS Glue](#)
 - [Laden komprimierter Datendateien aus Amazon S3 mit Amazon Redshift](#)
- Verwenden Sie zweckgerichtet erstellte [Analytikservices](#) für die Datenverarbeitung in Ihrer Architektur. Ausführlichere Informationen zu speziell entwickelten Analytikservices von AWS finden Sie unter [AWS re:Invent 2.022 - Building modern data architectures on AWS](#).
- Verwenden Sie die Datenbank-Engine, die das dominierende Abfragemuster jeweils am besten unterstützt. Verwalten Sie Ihre Datenbankindizes im Hinblick auf die effiziente Ausführung von Abfragen. Weitere Details finden Sie unter [AWS-Datenbanken](#) und [AWS re:Invent 2.022 - Modernize apps with purpose-built databases](#).

- Minimieren von Datenübertragungen: Wählen Sie Netzwerkprotokolle aus, die die Menge der Netzwerkkapazitäten reduzieren, die in Ihrer Architektur verbraucht werden.

Ressourcen

Zugehörige Dokumente:

- [COPY from columnar data formats with Amazon Redshift](#)
- [Converting Your Input Record Format in Firehose](#)
- [Improve query performance on Amazon Athena by Converting to Columnar Formats](#)
- [Monitoring DB load with Performance Insights on Amazon Aurora](#)
- [Monitoring DB load with Performance Insights on Amazon RDS](#)
- [Speicherklasse Amazon S3 Intelligent-Tiering](#)
- [Build a CQRS event store with Amazon DynamoDB](#)

Zugehörige Videos:

- [AWS re:Invent 2.022 - Building data mesh architectures on AWS](#)
- [AWS re:Invent 2.023 - Deep dive into Amazon Aurora and its innovations](#)
- [AWS re:Invent 2.023 - Improve Amazon EBS efficiency and be more cost-efficient](#)
- [AWS re:Invent 2.023 - Optimizing storage price and performance with Amazon S3](#)
- [AWS re:Invent 2.023 - Building and optimizing a data lake on Amazon S3](#)
- [AWS re:Invent 2023 – Advanced event-driven patterns with Amazon EventBridge](#)

Zugehörige Beispiele:

- [AWS Purpose Built Databases Workshop](#)
- [AWS Modern Data Architecture Immersion Day](#)
- [Build a Data Mesh on AWS](#)

Daten

Frage

- [SUS 4 Wie können Sie Datenverwaltungsrichtlinien und -muster zur Unterstützung Ihrer Nachhaltigkeitsziele nutzen?](#)

SUS 4 Wie können Sie Datenverwaltungsrichtlinien und -muster zur Unterstützung Ihrer Nachhaltigkeitsziele nutzen?

Implementieren Sie Verfahren für die Datenverwaltung, die den zur Unterstützung Ihrer Workload bereitgestellten Speicher und die für dessen Nutzung erforderlichen Ressourcen reduzieren. Verstehen Sie Ihre Daten und setzen Sie Speichertechnologien und -konfigurationen ein, die den geschäftlichen Mehrwert der Daten und deren Nutzung besser fördern. Verschieben Sie die Daten während des Lebenszyklus zu effizienteren Speichern mit geringerer Leistung, wenn die Anforderungen abnehmen. Löschen Sie Daten, die nicht mehr benötigt werden.

Best Practices

- [SUS04-BP01 Implementieren Sie eine Datenklassifizierungsrichtlinie](#)
- [SUS04-BP02 Verwenden von Technologien, die Datenzugriff und Speichermuster unterstützen](#)
- [SUS04-BP03 Verwalten des Lebenszyklus von Datensätzen mithilfe von Richtlinien](#)
- [SUS04-BP04 Nutzen Sie Elastizität und Automatisierung, um den Blockspeicher oder das Dateisystem zu erweitern](#)
- [SUS04-BP05 Entfernen nicht benötigter oder redundanter Daten](#)
- [SUS04-BP06 Verwenden geteilter Dateisysteme oder Objektspeicher für den Zugriff auf allgemeine Daten](#)
- [SUS04-BP07 Minimieren von Datenübertragungen zwischen Netzwerken](#)
- [SUS04-BP08 Sichern von Daten nur in dem Fall, wenn ihre erneute Erstellung schwierig ist](#)

SUS04-BP01 Implementieren Sie eine Datenklassifizierungsrichtlinie

Klassifizieren Sie die Daten, um zu verstehen, wie wichtig sie für die Geschäftsergebnisse sind, und wählen Sie die richtige energieeffiziente Speicherebene zur Speicherung der Daten.

Typische Anti-Muster:

- Sie identifizieren keine Datenbestände mit ähnlichen Merkmalen (z. B. Sensibilität, Geschäftskritikalität oder gesetzliche Anforderungen), die verarbeitet oder gespeichert werden.
- Sie haben keinen Datenkatalog zur Inventarisierung Ihrer Datenbestände eingeführt.

Vorteile der Nutzung dieser bewährten Methode: Durch die Implementierung einer Datenklassifizierungsrichtlinie können Sie die energieeffizienteste Speicherebene für Daten bestimmen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Bei der Datenklassifizierung wird identifiziert, welche Arten von Daten in einem Informationssystem verarbeitet und gespeichert werden, das einer Organisation gehört oder von ihr betrieben wird. Dazu gehört auch die Bestimmung der Kritikalität der Daten und der wahrscheinlichen Auswirkungen von Preisgaben, Verlusten oder Missbrauch von Daten.

Implementieren Sie Richtlinien zur Datenklassifizierung, indem Sie von der kontextuellen Verwendung der Daten ausgehen und ein Kategorisierungsschema erstellen, das den Grad der Kritikalität eines bestimmten Datensatzes für die Abläufe einer Organisation berücksichtigt.

Implementierungsschritte

- Bestandsaufnahme vornehmen: Führen Sie eine Bestandsaufnahme der verschiedenen Datentypen durch, die für Ihre Workload vorhanden sind.
- Daten gruppieren: Bestimmen Sie die Kritikalität, Vertraulichkeit, Integrität und Verfügbarkeit von Daten auf der Grundlage des Risikos für die Organisation. Verwenden Sie diese Anforderungen, um Daten in eine der von Ihnen gewählten Datenklassifizierungsebenen einzuteilen. Ein Beispiel finden Sie unter [Vier einfache Schritte zur Klassifizierung Ihrer Daten und zur Sicherung Ihres Startups](#).
- Datenklassifizierungsebenen und Richtlinien definieren: Definieren Sie für jede Datengruppe die Datenklassifizierungsebene (z. B. öffentlich oder vertraulich) und die Verarbeitungsrichtlinien. Kennzeichnen Sie Daten entsprechend. Einzelheiten zu den Kategorien für die Datenklassifizierung finden Sie im Whitepaper zur Datenklassifizierung.
- Regelmäßige Überprüfung: Überprüfen und kontrollieren Sie Ihre Umgebung regelmäßig auf nicht markierte und nicht klassifizierte Daten. Verwenden Sie die Automatisierung, um diese Daten zu identifizieren und die Daten entsprechend zu klassifizieren und zu markieren. Ein Beispiel finden Sie unter [Data Catalog and crawlers in AWS Glue](#).
- Datenkatalog einrichten: Richten Sie einen Datenkatalog mit Prüfungs- und Governance-Funktionen ein.
- Dokumentation: Dokumentieren Sie Datenklassifizierungsrichtlinien und Verarbeitungsverfahren für jede Datenklasse.

Ressourcen

Zugehörige Dokumente:

- [Nutzung von AWS Cloud zur Unterstützung der Datenklassifizierung](#)
- [Kennzeichnen Sie Richtlinien von AWS Organizations](#)

Zugehörige Videos:

- [AWS re:Invent 2022 — Agilität mit aktivierter Datenverwaltung ermöglichen AWS](#)
- [AWS re:Invent 2023 — Datenschutz und Resilienz mit Speicher AWS](#)

SUS04-BP02 Verwenden von Technologien, die Datenzugriff und Speichermuster unterstützen

Nutzen Sie Speichertechnologien, die den Zugriff auf Ihre Daten und ihre Speicherung jeweils optimal unterstützen, um die Zahl der bereitgestellten Ressourcen zu minimieren und gleichzeitig den Workload zu unterstützen.

Typische Anti-Muster:

- Sie gehen davon aus, dass für alle Workloads ähnliche Datenspeicher- und Zugriffsmuster gelten.
- Sie verwenden nur eine Speicherebene, vorausgesetzt, dass alle Workloads in diese Ebene passen.
- Sie gehen davon aus, dass Datenzugriffsmuster im Laufe der Zeit konsistent bleiben.

Vorteile der Nutzung dieser bewährten Methode: Die Auswahl und Optimierung Ihrer Speichertechnologien auf der Grundlage von Datenzugriffs- und Speichermustern hilft Ihnen, die erforderlichen Cloud-Ressourcen zu reduzieren, um Ihre Geschäftsanforderungen zu erfüllen und die Gesamteffizienz der Cloud-Workload zu verbessern.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Niedrig

Implementierungsleitfaden

Wählen Sie für maximale Leistungseffizienz die für Ihre Zugriffsmuster geeignete Speicherlösung, oder passen Sie Ihre Zugriffsmuster an die Speicherlösung an.

Implementierungsschritte

- **Daten- und Zugriffsmerkmale bewerten:** Bewerten Sie Ihre Datenmerkmale und Zugriffsmuster, um die wichtigsten Merkmale Ihres Speicherbedarfs zu erfassen. Zu den berücksichtigenden Schlüsselmerkmalen gehören:
 - Datentyp: strukturiert, semistrukturiert, unstrukturiert
 - Datenwachstum: begrenzt, unbegrenzt
 - Lebensdauer von Daten: anhaltend, flüchtig, vorübergehend
 - Zugriffsmuster: Lese- oder Schreibzugriff, Häufigkeit, schwankend oder konsistent
- **Die richtige Speichertechnologie auswählen:** Migrieren Sie Daten zur geeigneten Speichertechnologie, die Ihre Datenmerkmale und Zugriffsmuster unterstützt. Hier sind einige Beispiele für AWS-Speichertechnologien und ihre Schlüsselmerkmale:

Typ	Technologie	Schlüsselmerkmale
Objektspeicher	Amazon S3	Ein Objektspeicherservice mit unbegrenzter Skalierbarkeit, hoher Verfügbarkeit und mehreren Zugriffsoptionen. Für die Übertragung von Objekten in und aus Amazon S3 und den Zugriff auf diese Objekte können Sie einen Service wie z. B. Transfer Acceleration oder Zugangspunkte verwenden, um Ihren Standort, Ihre Sicherheitsanforderungen und Zugriffsmuster zu unterstützen.
Archivieren von Speichern	Amazon Glacier	Speicherklasse von Amazon S3 für die Datenarchivierung.
Gemeinsames Dateisystem	Amazon Elastic File System (Amazon EFS)	Bereitstellbares Dateisystem, auf das mehrere Arten

Typ	Technologie	Schlüsselmerkmale
		von Datenverarbeitungs- lösungen zugreifen können. Amazon EFS erweitert und verringert Speicher automati- sch und ist leistungsoptimiert , um durchgängig niedrige Latenzen zu bieten.
Gemeinsames Dateisystem	Amazon FSx	Basiert auf den aktuellen AWS-Datenverarbeit- ungslösungen und unterstüt- zt vier gängige Dateisyst- eme: NetApp ONTAP, OpenZFS, Windows File Server und Lustre. Die Amazon FSx- Latenz, der Durchsatz und die IOPS variieren je nach Dateisystem und sollten bei der Auswahl des richtigen Dateisystems für Ihre Workload-Anforderu- ngen berücksichtigt werden.
Blockspeicher	Amazon Elastic Block Store (Amazon EBS)	Skalierbarer, hochleist- ungsfähiger Blockspei- cherservice für Amazon Elastic Compute Cloud (Amazon EC2). Amazon EBS umfasst SSD-gestü- tzten Speicher für transakti- ons- und IOPS-intensive Workloads und HDD-gestü- tzten Speicher für durchsatz- intensive Workloads.

Typ	Technologie	Schlüsselmerkmale
Relationale Datenbank	Amazon Aurora , Amazon RDS , Amazon Redshift	Sie unterstützt AKID-Transaktionen (Atomarität, Konsistenz, Isolation und Dauerhaftigkeit) und gewährleistet die referentielle Integrität sowie eine starke Datenkonsistenz. Bei zahlreichen herkömmlichen Anwendungen, Enterprise Resource Planning (ERP), Customer Relationship Management (CRM) und E-Commerce-Systemen werden relationale Datenbanken zum Speichern der Daten verwendet.
Schlüssel-Werte-Datenbank	Amazon DynamoDB	Für gängige Zugriffsmuster optimiert, üblicherweise zum Speichern und Abrufen großer Datenmengen. Web-Apps mit hohem Datenverkehr, E-Commerce-Systeme und Gaming-Anwendungen sind typische Anwendungsfälle für Schlüssel-Werte-Datenbanken.

- Speicherzuweisung automatisieren: Überwachen Sie bei Speichersystemen mit einer festen Größe, z. B. Amazon EBS oder Amazon FSx, den verfügbaren Speicherplatz und automatisieren die Speicherzuweisung bei Erreichen eines Schwellenwertes. Sie können mithilfe von Amazon CloudWatch verschiedene Metriken für [Amazon EBS](#) und [Amazon FSx](#) erfassen und analysieren.
- Die richtige Speicherklasse wählen: Wählen Sie die passende Speicherklasse für Ihre Daten.
 - Amazon-S3-Speicherklassen können auf Objektebene konfiguriert werden. Ein einzelner Bucket kann Objekte enthalten, die in allen Speicherklassen gespeichert sind.

- Sie können [Amazon-S3-Lebenszyklusrichtlinien](#) verwenden, um Objekte automatisch zwischen Speicherklassen zu wechseln oder Daten zu entfernen, ohne dass die Anwendung geändert werden muss. Im Allgemeinen müssen Sie bei diesen Speichermechanismen einen Kompromiss zwischen Ressourceneffizienz, Zugriffslatenz und Zuverlässigkeit eingehen.

Ressourcen

Zugehörige Dokumente:

- [Amazon-EBS-Volume-Typen](#)
- [Amazon-EC2-Instance-Speicher](#)
- [Amazon S3 Intelligent-Tiering](#)
- [Amazon-EBS-E/A-Merkmale](#)
- [Verwenden von Amazon-S3-Speicherklassen](#)
- [Was ist Amazon Glacier?](#)

Zugehörige Videos:

- [AWS re:Invent 2.023 - Improve Amazon EBS efficiency and be more cost-efficient](#)
- [AWS re:Invent 2.023 - Optimizing storage price and performance with Amazon S3](#)
- [AWS re:Invent 2.023 - Building and optimizing a data lake on Amazon S3](#)
- [AWS re:Invent 2.022 - Building modern data architectures on AWS](#)
- [AWS re:Invent 2.022 - Modernize apps with purpose-built databases](#)
- [AWS re:Invent 2.022 - Building data mesh architectures on AWS](#)
- [AWS re:Invent 2.023 - Deep dive into Amazon Aurora and its innovations](#)
- [AWS re:Invent 2.023 - Advanced data modeling with Amazon DynamoDB](#)

Zugehörige Beispiele:

- [Amazon-S3-Beispiele](#)
- [AWS Purpose Built Databases Workshop](#)
- [Databases for Developers](#)
- [AWS Modern Data Architecture Immersion Day](#)
- [Build a Data Mesh on AWS](#)

SUS04-BP03 Verwalten des Lebenszyklus von Datensätzen mithilfe von Richtlinien

Verwalten Sie den Lebenszyklus aller Daten und setzen Sie automatisch Löschen durch, um den für Ihre Workload benötigten Speicher insgesamt zu minimieren.

Typische Anti-Muster:

- Sie löschen Daten manuell.
- Sie löschen keine Workload-Daten.
- Sie verschieben Daten nicht abhängig von den Aufbewahrungs- und Zugriffsanforderungen in energieeffizientere Speicherebenen.

Vorteile der Einführung dieser bewährten Methode: Durch Richtlinien für den Lebenszyklus wird die Effizienz des Datenzugriffs und der Datenaufbewahrung für eine Workload sichergestellt.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Datensätze verfügen während ihres Lebenszyklus normalerweise über unterschiedliche Aufbewahrungs- und Zugriffsanforderungen. So kann eine Anwendung z. B. für einen bestimmten Zeitraum häufig Zugriff auf einige Datensätze benötigen. Danach wird nur noch unregelmäßig darauf zugegriffen. Implementieren Sie zur Verbesserung der Effizienz von Datenspeicherung und Berechnung im Laufe der Zeit Lebenszyklusrichtlinien. Dies sind Regeln, die definieren, wie Daten im Laufe der Zeit behandelt werden.

Mit Lebenszyklus-Konfigurationsregeln können Sie einen bestimmten Speicherservice anweisen, einen Datensatz in energieeffizientere Speicherebenen zu verschieben, zu archivieren oder zu löschen. Dieses Verfahren minimiert den aktiven Datenspeicher und die Zahl der Abrufe, was zu einem geringeren Energieverbrauch führt. Darüber hinaus unterstützen Verfahren wie das Archivieren oder Löschen veralteter Daten die Einhaltung gesetzlicher Vorschriften und die Daten-Governance.

Implementierungsschritte

- Verwenden der Datenklassifizierung: [Klassifizieren Sie die Datensätze in Ihrem Workload](#).
- Definieren von Bearbeitungsregeln. Definieren Sie Bearbeitungsverfahren für jede Datenklasse.
- Aktivieren der Automatisierung: Legen Sie automatisierte Lebenszyklusrichtlinien fest, um Lebenszyklusregeln durchzusetzen. Hier finden Sie einige Beispiele zum Einrichten von automatisierten Lebenszyklusrichtlinien für unterschiedliche AWS-Speicherservices:

Storage Service	Festlegen von automatisierten Lebenszyklusrichtlinien
Amazon S3	Mit Amazon-S3-Lebenszyklen können Sie Ihre Objekte während ihres gesamten Lebenszyklus verwalten. Wenn die Zugriffsmuster unbekannt oder nicht prognostizierbar sind oder sich ändern, können Sie Amazon S3 Intelligent-Tiering verwenden. Hiermit werden Zugriffsmuster überwacht und Objekte, auf die nicht zugegriffen wurde, automatisch in kostengünstigere Zugriffsebenen verschoben. Anhand der Metriken von Amazon S3 Storage Lens können Sie Optimierungsmöglichkeiten und Lücken im Lebenszyklusmanagement ermitteln.
Amazon Elastic Block Store	Mit Amazon Data Lifecycle Manager lassen sich das Erstellen, Aufbewahren und Löschen von Amazon-EBS-Snapshots und Amazon-EB S-gestützten AMIs automatisieren.
Amazon Elastic File System	Das Amazon-EFS-Lebenszyklusmanagement verwaltet den Dateispeicher für Ihre Dateisysteme automatisch.
Amazon Elastic Container Registry	Amazon-ECR-Lebenszyklusrichtlinien automatisieren die Bereinigung von Container-Images, indem Images abhängig von Alter oder Anzahl ablaufen.
AWS Elemental MediaStore	Sie können eine Objektlebenszyklus-Richtlinie verwenden, die steuert, wie lange Objekte im MediaStore-Container gespeichert werden sollen.

- Löschen nicht genutzter Komponenten: Löschen Sie nicht genutzte Volumes, Snapshots und Daten, deren Aufbewahrungszeitraum abgelaufen ist. Verwenden Sie zum Löschen native Service-Features wie [Amazon DynamoDB Time To Live](#) oder die [Amazon-CloudWatch-Protokollaufbewahrung](#).
- Aggregieren und Komprimieren: Aggregieren und komprimieren Sie Daten auf der Basis von Lebenszyklusregeln, wenn möglich.

Ressourcen

Zugehörige Dokumente:

- [Optimieren Ihrer Amazon-S3-Lebenszyklusregeln mit der Amazon-S3-Speicherklassenanalyse](#)
- [Evaluating Resources with AWS-Config-Regeln](#)

Zugehörige Videos:

- [AWS re:Invent 2.021 - Amazon S3 Lifecycle best practices to optimize your storage spend](#)
- [AWS re:Invent 2.023 - Optimizing storage price and performance with Amazon S3](#)
- [Simplify Your Data Lifecycle and Optimize Storage Costs With Amazon S3 Lifecycle](#)
- [Reduce Your Storage Costs Using Amazon S3 Storage Lens](#)

SUS04-BP04 Nutzen Sie Elastizität und Automatisierung, um den Blockspeicher oder das Dateisystem zu erweitern

Verwenden Sie Elastizität und Automatisierung, um den Block-Speicher oder das Dateisystem zu erweitern, wenn das Datenvolumen zunimmt, um den bereitgestellten Gesamtspeicher zu minimieren.

Typische Anti-Muster:

- Sie unterhalten einen großen Block-Speicher oder ein großes Dateisystem für künftige Anforderungen.
- Sie stellen die Eingabe- und Ausgabeoperationen pro Sekunde (IOPS) Ihres Dateisystems zu viel bereit.
- Sie überwachen die Nutzung Ihrer Daten-Volumes nicht.

Vorteile der Nutzung dieser bewährten Methode: Die Minimierung der übermäßigen Bereitstellung für das Speichersystem reduziert ungenutzte Ressourcen und verbessert die Gesamteffizienz Ihrer Workload.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Erstellen Sie Block-Speicher und Dateisysteme mit Größenzuweisung, Durchsatz und Latenz, die den Anforderungen Ihrer Workloads entsprechen. Verwenden Sie Elastizität und Automatisierung, um den Block-Speicher oder das Dateisystem zu erweitern, wenn das Datenvolumen zunimmt, ohne dass diese Speicherservices übermäßig bereitgestellt werden.

Implementierungsschritte

- Stellen Sie bei Speichern mit fester Größe wie [Amazon](#) sicherEBS, dass Sie die Menge des verwendeten Speichers im Vergleich zur Gesamtspeichergröße überwachen, und erstellen Sie nach Möglichkeit eine Automatisierung, um die Speichergröße zu erhöhen, wenn ein Schwellenwert erreicht wird.
- Verwenden Sie elastische Volumes und verwaltete Blockdaten-Services, um automatisch zusätzlichen Speicher zuzuweisen, wenn die Menge der persistenten Daten wächst. Beispielsweise können Sie [Amazon EBS Elastic Volumes](#) verwenden, um die Volume-Größe und den Volume-Typ zu ändern oder die Leistung Ihrer EBS Amazon-Volumes anzupassen.
- Wählen Sie die korrekte Speicherklasse sowie den korrekten Leistungs- und Durchsatz-Modus für Ihr Dateisystem für Ihre geschäftlichen Anforderungen und überschreiten Sie diese nicht.
 - [EFSLeistung von Amazon](#)
 - [EBSAmazon-Volumenleistung auf Linux-Instances](#)
- Legen Sie Zielstufen für die Nutzung Ihrer Daten-Volumes fest und passen Sie die Größe von Volumes an, die außerhalb der erwarteten Bereiche liegen.
- Passen Sie die Größe schreibgeschützter Volumes an die Datenmenge an.
- Migrieren Sie Daten zu Objektspeichern, um zu vermeiden, dass die überschüssige Kapazität aus Volumes mit fester Größe im Blockspeicher bereitgestellt wird.
- Überprüfen Sie elastische Volumes und Dateisysteme, beenden Sie nicht genutzte und verkleinern Sie zu große Volumes, um sie an den aktuellen Datenumfang anzupassen.

Ressourcen

Zugehörige Dokumente:

- [Erweitern Sie das Dateisystem, nachdem Sie die Größe eines EBS Volumes geändert haben](#)
- [Ändern Sie ein Volume mit Amazon EBS Elastic Volumes](#)
- [Amazon FSx – Dokumentation](#)
- [What is Amazon Elastic File System?](#)

Zugehörige Videos:

- [Tiefer Einblick in Amazon EBS Elastic Volumes](#)
- [Optimierungsstrategien von Amazon EBS und Snapshot für bessere Leistung und Kosteneinsparungen](#)
- [Optimierung von Amazon im EFS Hinblick auf Kosten und Leistung mithilfe von Best Practices](#)

SUS04-BP05 Entfernen nicht benötigter oder redundanter Daten

Entfernen Sie nicht benötigte oder redundante Daten, um die zum Speichern Ihrer Datensätze benötigten Speicherressourcen zu minimieren.

Typische Anti-Muster:

- Sie duplizieren Daten, die leicht abgerufen oder erneut erstellt werden können.
- Sie sichern alle Daten, ohne ihre Kritikalität zu berücksichtigen.
- Sie löschen Daten nur unregelmäßig, nur bei bestimmten Ereignissen oder gar nicht.
- Sie speichern Daten redundant, unabhängig von der Stabilität des Speicherservices.
- Sie aktivieren die Amazon-S3-Versionsverwaltung, ohne dass dies geschäftlich gerechtfertigt ist.

Vorteile der Einführung dieser bewährten Methode: Durch das Entfernen nicht benötigter Daten werden die für Ihre Workload benötigte Speichergröße und die Umweltbelastungen durch die Workload reduziert.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Wenn Sie nicht benötigte und redundante Datensätze entfernen, können Sie die Speicherkosten und den ökologischen Fußabdruck reduzieren. Dieses Verfahren kann auch die Datenverarbeitung effizienter gestalten, da Datenverarbeitungsressourcen nur wichtige Daten und keine unnötigen Daten verarbeiten. Automatisieren Sie das Löschen von nicht benötigten Daten. Verwenden Sie Technologien, die Daten auf Datei- und Blockebene deduplizieren. Verwenden Sie Service-Features für native Datenreplikation und Redundanz.

Implementierungsschritte

- Öffentliche Datensätze bewerten: Untersuchen Sie, ob Sie das Speichern von Daten vermeiden können, indem Sie vorhandene, öffentlich verfügbare Datensätze in [AWS Data Exchange](#) und [Offene Daten in AWS](#) verwenden.
- Daten deduplizieren: Verwenden Sie Mechanismen, die Daten auf Block- und Objektebene deduplizieren können. Hier finden Sie einige Beispiele zum Deduplizieren von Daten in AWS:

Storage Service	Deduplizierungsmechanismus
Amazon S3	Verwenden Sie AWS Lake Formation FindMatches und das neue FindMatches ML Transform, um übereinstimmende Einträge in einem Datensatz zu finden (auch solche ohne Bezeichner).
Amazon FSx	Verwenden Sie die Datendeduplizierung in Amazon FSx für Windows.
Snapshots von Amazon Elastic Block Store	Snapshots sind inkrementelle Backups, d. h., es werden nur die Blöcke des Geräts gespeichert, die sich seit der letzten Snapshot-Speicherung geändert haben.

- Lebenszyklusrichtlinien verwenden: Verwenden Sie Lebenszyklusrichtlinien, damit ungenutzte Komponenten automatisch gelöscht werden. Nutzen Sie zum Löschen native Service-Features wie [Amazon DynamoDB Time to Live](#), [Amazon S3 Lifecycle](#) oder die [Protokollaufbewahrung von Amazon CloudWatch](#).

- Datenvirtualisierung verwenden: Verwenden Sie Virtualisierungsfunktionen in AWS, um Daten an der Quelle beizubehalten und eine Duplikation zu vermeiden.
 - [Cloudnative Datenvirtualisierung in AWS](#)
 - [Optimierung von Datenmustern mithilfe von Amazon Redshift Data Sharing](#)
- Inkrementelle Sicherungen verwenden: Verwenden Sie Sicherungstechnologien, die inkrementelle Sicherungen ermöglichen.
- Native Stabilität verwenden: Nutzen Sie die Stabilität von [Amazon S3](#) und die [Replikation von Amazon EBS](#), um Ihre Stabilitätsziele zu erreichen, statt selbst verwaltete Technologien zu verwenden (z. B. ein redundantes Array unabhängiger Datenträger (RAID)).
- Effiziente Protokollierung verwenden: Zentralisieren Sie Protokoll- und Nachverfolgungsdaten, deduplizieren Sie identische Protokolleinträge und richten Sie Mechanismen für die Anpassung der Ausführlichkeit ein, wenn notwendig.
- Effizientes Caching nutzen: Füllen Sie Caches nur vorab aus, wenn dies begründet werden kann.
- Richten Sie Überwachung und Automatisierung für den Cache ein, um seine Größe entsprechend anzupassen.
- Veraltete Versionsressourcen entfernen: Entfernen Sie veraltete Bereitstellungen und Komponenten aus Objektspeichern und Edge-Caches, wenn Sie neue Versionen Ihres Workloads veröffentlichen.

Ressourcen

Zugehörige Dokumente:

- [Ändern der Protokoll-Datenaufbewahrung in CloudWatch Logs](#)
- [Datendeduplizierung in Amazon FSx für Windows File Server](#)
- [Amazon FSxs von Amazon FSx für ONTAP einschließlich Datendeduplizierung](#)
- [Aufheben der Gültigkeit von Dateien in Amazon CloudFront](#)
- [Sichern und Wiederherstellen von Amazon-EFS-Dateisystemen mit AWS Backup](#)
- [What is Amazon CloudWatch Logs?](#)
- [Arbeiten mit Sicherungen in Amazon RDS](#)
- [Integrate and deduplicate datasets using AWS Lake Formation](#)

Zugehörige Videos:

- [Amazon Redshift Data Sharing Use Cases](#)

Zugehörige Beispiele:

- [Wie verwende ich Amazon Athena, um meine Amazon-S3-Serverzugriffsprotokolle zu analysieren?](#)

SUS04-BP06 Verwenden geteilter Dateisysteme oder Objektspeicher für den Zugriff auf allgemeine Daten

Verwenden Sie geteilte Dateisysteme oder Speicher, um Datenduplizierungen zu vermeiden und eine effizientere Infrastruktur für Ihre Workload zu ermöglichen.

Typische Anti-Muster:

- Sie stellen für jeden einzelnen Client Speicher bereit.
- Sie trennen Daten-Volumes von inaktiven Clients nicht ab.
- Sie ermöglichen keinen Zugriff auf Speicher über Plattformen und Systeme hinweg.

Vorteile der Nutzung dieser bewährten Methode: Die Verwendung geteilter Dateisysteme oder Speicher ermöglicht die gemeinsame Nutzung von Daten für mehrere Verbraucher, ohne dass diese dazu kopiert werden müssen. Dies reduziert den Umfang der erforderlichen Speicherressourcen für die Workload.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Wenn Sie mehrere Benutzer oder Anwendungen haben, die auf dieselben Datensätze zugreifen müssen, ist die Verwendung geteilter Speichertechnologien wichtig für eine effiziente Infrastruktur für Ihre Workload. Solche Technologien bieten einen zentralen Speicherort für die Speicherung und Verwaltung von Datensätzen und zur Vermeidung von Datenduplizierungen. Dazu wird die Konsistenz der Daten über verschiedene Systeme hinweg durchgesetzt. Hinzu kommt, dass geteilte Speicher die effizientere Nutzung der Datenverarbeitungsleistung ermöglichen, da mehr Datenverarbeitungsressourcen gleichzeitig auf Daten zugreifen und diese verarbeiten können.

Rufen Sie Daten von diesen geteilten Speicherservices nur bei Bedarf ab und trennen Sie nicht genutzte Volumes, um Ressourcen freizugeben.

Implementierungsschritte

- Verwenden geteilter Speicher: Migrieren Sie Daten zu einem geteilten Speicher, wenn die Daten mehrfach genutzt werden. Hier sind einige Beispiele für geteilte Speichertechnologien in AWS:

Speicheroption	Wann sollte dies verwendet werden?
Amazon EBS Multi-Attach	Amazon EBS Multi-Attach ermöglicht die Anfügung eines einzelnen bereitgestellten IOPS SSD (io1 oder io2)-Volumes an mehrere Instances in derselben Availability Zone.
Amazon EFS	Siehe When to Choose Amazon EFS .
Amazon FSx	Siehe Auswählen eines Amazon-FSx-Dateisystems .
Amazon S3	Anwendungen, die keine Dateisystemstruktur benötigen und zur Arbeit mit Objektspeichern gedacht sind, können Amazon S3 als massive, skalierbare, dauerhafte und kostengünstige Speicherlösung nutzen.

- Abrufen von Daten wie erforderlich: Kopieren Sie Daten nur dann zu geteilten Dateisystemen oder rufen Sie Daten nur aus ihnen ab, wenn Sie die Daten benötigen. Sie können beispielsweise ein [Dateisystem von Amazon FSx für Lustre mit Unterstützung durch Amazon S3](#) erstellen und nur die Teilmenge der Daten laden, die für die Verarbeitung von Aufgaben zu Amazon FSx benötigt werden.
- Löschen nicht benötigter Daten: Löschen Sie Daten entsprechend Ihren Nutzungsmustern, wie in [SUS04-BP03 Verwalten des Lebenszyklus von Datensätzen mithilfe von Richtlinien](#) beschrieben.
- Trennen inaktiver Clients: Trennen Sie Volumes von Clients, die diese nicht aktiv verwenden.

Ressourcen

Zugehörige Dokumente:

- [Linking your file system to an Amazon S3 bucket](#)
- [Using Amazon EFS for AWS Lambda in your serverless applications](#)

- [Amazon EFS Intelligent-Tiering Optimizes Costs for Workloads with Changing Access Patterns](#)
- [Using Amazon FSx with your on-premises data repository](#)

Zugehörige Videos:

- [Storage cost optimization with Amazon EFS](#)
- [AWS re:Invent 2.023 - What's new with AWS file storage](#)
- [AWS re:Invent 2.023 - File storage for builders and data scientists on Amazon Elastic File System](#)

SUS04-BP07 Minimieren von Datenübertragungen zwischen Netzwerken

Verwenden Sie gemeinsam genutzte Dateisysteme oder Objektspeicher zum Zugriff auf häufig genutzte Daten und minimieren Sie die zur Unterstützung von Datenverschiebungen für Ihre Workload benötigten Netzwerkressourcen.

Typische Anti-Muster:

- Sie speichern alle Daten in derselben AWS-Region, unabhängig davon, wo sich deren Benutzer befinden.
- Sie optimieren Datenumfang und -format nicht vor der Verschiebung über das Netzwerk.

Vorteile der Nutzung dieser bewährten Methode: Die Optimierung der Datenverschiebung über das Netzwerk reduziert den Umfang der für die Workload benötigten Netzwerkressourcen und verringert die Umweltauswirkungen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Das Verschieben von Daten in der gesamten Organisation erfordert Datenverarbeitungs-, Netzwerk- und Speicherressourcen. Verwenden Sie Techniken zur Minimierung von Datenverschiebungen und verbessern Sie die Gesamteffizienz Ihrer Workloads.

Implementierungsschritte

- Verwenden der Proximity: Berücksichtigen Sie bei der Entscheidung hinsichtlich der [Auswahl einer Region für Ihren Workload](#) die Nähe zu Daten oder Benutzern.

- **Partitionierungsservices:** Partitionieren Sie regional genutzte Services so, dass regionsspezifische Daten in der Region gespeichert werden, in der sie genutzt werden.
- **Verwenden effizienter Dateiformate:** Verwenden Sie effiziente Dateiformate (z. B. Parquet oder ORC) und komprimieren Sie Daten, bevor Sie diese durch das Netzwerk leiten.
- **Minimieren von Datenbewegungen:** Verschieben Sie keine ungenutzten Daten. Einige Beispiele, die Ihnen helfen können, das Verschieben ungenutzter Daten zu vermeiden:
 - Beschränken Sie API-Antworten nur auf relevante Daten.
 - Aggregieren Sie Daten, wenn keine detaillierten Informationen auf Datensatzebene benötigt werden.
 - Siehe [Well-Architected Lab - Optimize Data Pattern Using Amazon Redshift Data Sharing](#).
 - Erwägen Sie die [kontoübergreifende Datenfreigabe in AWS Lake Formation](#).
- **Verwenden von Edge-Services:** Verwenden Sie Services, die Ihnen helfen können, den Code näher an den Benutzern Ihres Workloads auszuführen.

Service	Wann sollte dies verwendet werden?
Lambda@Edge	Verwenden Sie dies für rechenintensive Anwendungen, die ausgeführt werden, wenn sich Objekte nicht im Zwischenspeicher befinden.
CloudFront-Funktionen	Verwenden Sie diese für einfache Anwendungsfälle wie HTTP(s)-Anfragen oder Antwortmanipulationen, die von kurzlebigen Funktionen initiiert werden können.
AWS IoT Greengrass	Führen Sie lokale Datenverarbeitungsvorgänge, Messaging sowie die Datenzwischenspeicherung für verbundene Geräte aus.

Ressourcen

Zugehörige Dokumente:

- [Optimizing your AWS Infrastructure for Sustainability, Part III: Networking](#)

- [Globale AWS-Infrastruktur](#)
- [Hauptfeatures von Amazon CloudFront einschließlich CloudFront Global Edge Network](#)
- [Compressing HTTP requests in Amazon OpenSearch Service](#)
- [Intermediäre Datenkompression mit Amazon EMR](#)
- [Loading compressed data files from Amazon S3 into Amazon Redshift](#)
- [Bereitstellen komprimierter Dateien mit Amazon CloudFront](#)

Zugehörige Videos:

- [Demystifying data transfer on AWS](#)

SUS04-BP08 Sichern von Daten nur in dem Fall, wenn ihre erneute Erstellung schwierig ist

Vermeiden Sie das Sichern von Daten ohne geschäftlichen Wert, um die Anforderungen an Speicherressourcen für Ihre Workload zu minimieren.

Typische Anti-Muster:

- Sie haben keine Sicherungsstrategie für Ihre Daten.
- Sie sichern Daten, die problemlos erneut erstellt werden können.

Vorteile der Nutzung dieser bewährten Methode: Das Vermeiden der Sicherung nichtkritischer Daten reduziert den Umfang der benötigten Speicherressourcen für die Workload und verringert die Umweltauswirkungen.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Die Vermeidung der Sicherung nicht benötigter Daten kann Kosten senken und die von der Workload verwendeten Speicherressourcen verringern. Sichern Sie nur Daten, die einen geschäftlichen Wert haben oder zur Erfüllung von Compliance-Anforderungen benötigt werden. Prüfen Sie Sicherungsrichtlinien und vermeiden Sie einen flüchtigen Speicher, der in einem Wiederherstellungsszenario keinen Wert bietet.

Implementierungsschritte

- Klassifizieren von Daten: Implementieren Sie eine Richtlinie für die Datenklassifizierung wie in [SUS04-BP01 Implementieren Sie eine Datenklassifizierungsrichtlinie](#) beschrieben.
- Entwickeln einer Sicherungsstrategie: Entwickeln Sie anhand der Kritikalität Ihrer Datenklassifizierung eine Sicherungsstrategie basierend auf Ihrem Recovery Time Objective (RTO) und Recovery Point Objective (RPO). Vermeiden Sie die Sicherung nichtkritischer Daten.
 - Schließen Sie Daten aus, die problemlos erneut erstellt werden können.
 - Schließen Sie flüchtige Daten von Sicherungen aus.
 - Schließen Sie lokale Kopien von Daten aus, es sei denn, die für die Wiederherstellung dieser Daten von einem gemeinsamen Standort benötigte Zeit überschreitet Ihre Service Level Agreements (SLAs).
- Verwenden einer automatisierten Sicherung: Verwenden Sie eine automatisierte Lösung oder einen verwalteten Service, um geschäftskritische Daten zu sichern.
 - [AWS Backup](#) ist ein vollständig verwalteter Service, der das Zentralisieren und Automatisieren des Datenschutzes in AWS-Services, in der Cloud sowie On-Premises erleichtert. Praktische Anleitungen zur Erstellung automatisierter Sicherungen mit AWS Backup finden Sie unter [Well-Architected Labs - Testing Backup and Restore of Data](#).
 - [Automate backups and optimize backup costs for Amazon EFS using AWS Backup](#).

Ressourcen

Zugehörige bewährte Methoden:

- [REL09-BP01 Ermitteln und Sichern aller zu sichernden Daten oder Reproduzieren der Daten aus Quellen](#)
- [REL09-BP03 Automatische Daten-Backups](#)
- [REL13-BP02: Verwenden von definierten Wiederherstellungsstrategien, um die Wiederherstellungsziele zu erreichen](#)

Zugehörige Dokumente:

- [Sichern und Wiederherstellen von Amazon-EFS-Dateisystemen mit AWS Backup](#)
- [Amazon EBS snapshots](#)
- [Working with backups on Amazon Relational Database Service](#)

- [APN-Partner: Partner, die Sie bei der Sicherung unterstützen können](#)
- [AWS Marketplace: Für die Sicherung geeignete Produkte](#)
- [Backing Up Amazon EFS](#)
- [Backing Up Amazon FSx for Windows File Server](#)
- [Backup and Restore for Amazon ElastiCache \(Redis OSS\)](#)

Zugehörige Videos:

- [AWS re:Invent 2.023 - Backup and disaster recovery strategies for increased resilience](#)
- [AWS re:Invent 2.023 - What's new with AWS Backup](#)
- [AWS re:Invent 2.021 - Backup, disaster recovery, and ransomware protection with AWS](#)

Hardware und Services

Frage

- [SUS 5 Wie wählen und nutzen Sie Cloud-Hardware und -Services in Ihrer Architektur so, dass Ihre Nachhaltigkeitsziele unterstützt werden?](#)

SUS 5 Wie wählen und nutzen Sie Cloud-Hardware und -Services in Ihrer Architektur so, dass Ihre Nachhaltigkeitsziele unterstützt werden?

Suchen Sie nach Möglichkeiten, die Auswirkungen auf die Nachhaltigkeit Ihrer Workloads durch Änderungen der Methoden für die Hardwareverwaltung zu reduzieren. Minimieren Sie den Umfang der für die Bereitstellung erforderlichen Hardware und wählen Sie die jeweils effizienteste Hardware und den effizientesten Service für die jeweilige Workload aus.

Best Practices

- [SUS05-BP01 Verwenden der geringstmöglichen Menge an Hardware zur Erfüllung Ihrer Anforderungen](#)
- [SUS05-BP02 Verwenden von Instance-Typen mit den geringsten Auswirkungen](#)
- [SUS05-BP03 Managed Services nutzen](#)
- [SUS05-BP04 Optimieren der Nutzung von hardwarebasierten Datenverarbeitungsbeschleunigern](#)

SUS05-BP01 Verwenden der geringstmöglichen Menge an Hardware zur Erfüllung Ihrer Anforderungen

Verwenden Sie die geringstmögliche Menge an Hardware für Ihre Workload, um Ihre geschäftlichen Anforderungen in effizienter Weise zu erfüllen.

Typische Anti-Muster:

- Sie überwachen die Ressourcenauslastung nicht.
- Sie haben Ressourcen mit geringer Auslastung in Ihrer Architektur.
- Sie prüfen die Nutzung statischer Hardware nicht, um festzustellen, ob sie neu dimensioniert werden muss.
- Sie formulieren keine Ziele für die Hardwarenutzung in Ihrer Datenverarbeitungsinfrastruktur auf der Grundlage geschäftlicher KPIs.

Vorteile der Nutzung dieser bewährten Methode: Die korrekte Dimensionierung Ihrer Cloud-Ressourcen hilft dabei, die Umweltauswirkungen von Workloads zu reduzieren, Geld zu sparen und Leistungsbenchmarks einzuhalten.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Wählen Sie die optimale Anzahl von Hardwaregeräten für Ihre Workload aus, um die allgemeine Effizienz zu verbessern. Die AWS Cloud bietet die Flexibilität, Ressourcen dynamisch durch verschiedene Mechanismen wie etwa [AWS Auto Scaling](#) zu erweitern oder zu reduzieren, um einem veränderten Bedarf gerecht zu werden. Dazu kommen [APIs und SDKs](#), mit denen Ressourcen mit minimalem Aufwand angepasst werden können. Verwenden Sie diese Möglichkeiten für häufige Änderungen an Ihren Workload-Implementierungen. Verwenden Sie dazu Dimensionierungsanleitungen von AWS-Tools für den effizienten Betrieb Ihrer Cloud-Ressourcen und die Erfüllung Ihrer geschäftlichen Anforderungen.

Implementierungsschritte

- **Auswahl des Instance-Typs:** Wählen Sie den Instance-Typ aus, der Ihren Anforderungen am besten entspricht. Weitere Informationen zur Auswahl von Instances von Amazon Elastic Compute Cloud und zur Verwendung von Mechanismen wie der attributbasierten Auswahl des Instance-Typs finden Sie im Folgenden:
 - [Wie wähle ich den geeigneten Amazon-EC2-Instance-Typ für meine Workload aus?](#)

- [Attributbasierte Auswahl von Instance-Typen für Amazon-EC2-Flotte](#)
- [Erstellen einer Auto-Scaling-Gruppe mit attributbasierter Auswahl des Instance-Typs](#)
- Skalierung: Skalieren Sie variable Workloads in kleinen Schritten.
- Verwendung mehrerer Einkaufsoptionen für Datenverarbeitung: Kombinieren Sie Instance-Flexibilität, Skalierbarkeit und Kosteneinsparungen mit mehreren Einkaufsoptionen für Datenverarbeitung.
- [On-Demand-Instances von Amazon EC2](#) eignen sich am besten für neue, zustandsbehaftete Workloads mit Spitzen, die hinsichtlich Instance-Typ, Standort oder Zeit nicht flexibel sein können.
- [Amazon EC2 Spot Instances](#) eignen sich hervorragend zur Ergänzung der anderen Optionen für Anwendungen, die fehlertolerant und flexibel sind.
- Nutzen Sie [Compute Savings Plans](#) für stabile Workloads, die Flexibilität ermöglichen, wenn sich Ihre Anforderungen (wie AZ, Region, Instance-Familien oder Instance-Typen) ändern.
- Nutzung der Vielfalt von Instances und Availability Zones: Maximieren Sie die Anwendungsverfügbarkeit und nutzen Sie überschüssige Kapazitäten, indem Sie Ihre Instances und Availability Zones diversifizieren.
- Korrekte Dimensionierung von Instances: Verwenden Sie die Empfehlungen zur Dimensionierung in AWS-Tools, um Anpassungen an Ihrer Workload vorzunehmen. Weitere Informationen finden Sie unter [Kostenoptimierung mit Empfehlungen zur richtigen Dimensionierung](#) und [Richtige Dimensionierung: Bereitstellen von Instances entsprechend den Workloads](#).
- Verwenden Sie die Empfehlungen zur Dimensionierung in AWS Cost Explorer oder [AWS Compute Optimizer](#) zur Identifizierung von Dimensionierungsmöglichkeiten.
- Verhandlung von Service Level Agreements (SLAs): Verhandeln Sie SLAs, die eine vorübergehende Reduzierung der Kapazität ermöglichen, während die Automatisierung Ersatzressourcen bereitstellt.

Ressourcen

Zugehörige Dokumente:

- [Optimizing your AWS Infrastructure for Sustainability, Part I: Compute](#)
- [Attribute based Instance Type Selection for Auto Scaling for Amazon EC2 Fleet](#)
- [AWS Compute Optimizer-Dokumentation](#)
- [Ausführen von Lambda: Leistungsoptimierung](#)

- [Dokumentation zu Auto Scaling](#)

Zugehörige Videos:

- [AWS re:Invent 2.023 - What's new with Amazon EC2](#)
- [AWS re:Invent 2.023 - Smart savings: Amazon Elastic Compute Cloud cost-optimization strategies](#)
- [AWS re:Invent 2.022 - Optimizing Amazon Elastic Kubernetes Service for performance and cost on AWS](#)
- [AWS re:Invent 2.023 - Sustainable compute: reducing costs and carbon emissions with AWS](#)

SUS05-BP02 Verwenden von Instance-Typen mit den geringsten Auswirkungen

Überwachen und nutzen Sie kontinuierlich neue Instance-Typen, um Verbesserungen bei der Energieeffizienz zu nutzen.

Typische Anti-Muster:

- Sie verwenden lediglich eine Familie von Instances.
- Sie verwenden nur x86-Instances.
- Sie geben einen Instance-Typ in Ihrer Konfiguration von Amazon EC2 Auto Scaling an.
- Sie verwenden AWS-Instances in einer Weise, für die sie nicht gedacht sind (beispielsweise für Datenverarbeitung optimierte Instances für speicherintensive Workloads).
- Sie evaluieren nicht regelmäßig neue Instance-Typen.
- Sie prüfen nicht die Empfehlungen von AWS-Dimensionierungstools wie etwa [AWS Compute Optimizer](#).

Vorteile der Nutzung dieser bewährten Methode: Durch die Verwendung energieeffizienter und korrekt dimensionierter Instances können Sie die Umweltauswirkungen und die Kosten Ihrer Workloads deutlich reduzieren.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Die Verwendung effizienter Instances für Cloud-Workloads ist von entscheidender Bedeutung für eine geringere Ressourcennutzung und die Kosteneffizienz. Überwachen Sie kontinuierlich die Einführung

neuer Instance-Typen und nutzen Sie Verbesserungen bei der Energieeffizienz, einschließlich Instance-Typen, die zur Unterstützung spezifischer Workloads bestimmt sind, wie z. B. Machine-Learning-Trainings und -Inferenzen und Videotranskodierung.

Implementierungsschritte

- Kennenlernen der Instance-Typen: Finden Sie Instance-Typen, mit denen Sie die Umweltbelastung Ihrer Workloads verringern können.
 - Abonnieren Sie [Neuerungen bei AWS](#), um sich über die aktuellen AWS-Technologien und -Instances auf dem Laufenden zu halten.
 - Informieren Sie sich über die verschiedenen AWS-Instance-Typen.
 - Informieren Sie sich über AWS-Graviton-basierte Instances, die die beste Leistung pro Watt in Amazon EC2 bieten. Sehen Sie sich [re:Invent 2.020 - Deep dive on AWS Graviton2 processor-powered Amazon EC2 instances](#) und [Deep dive into AWS Graviton3 and Amazon EC2 C7g instances](#) an.
- Verwendung von Instance-Typen mit den geringsten Auswirkungen: Planen Sie Ihre Workload und stellen Sie sie auf Instance-Typen mit den geringsten Auswirkungen um.
 - Definieren Sie einen Prozess zur Evaluierung neuer Features oder Instances für Ihre Workloads. Nutzen Sie die Agilität in der Cloud, um schnell zu testen, wie neue Instance-Typen die ökologische Nachhaltigkeit Ihrer Workloads verbessern können. Nutzen Sie Proxy-Metriken, um zu messen, wie viele Ressourcen Sie für eine Arbeitseinheit benötigen.
 - Modifizieren Sie Ihre Workload nach Möglichkeit so, dass sie mit unterschiedlichen Zahlen von vCPUs und Arbeitsspeichergrößen kompatibel ist, um die größtmögliche Auswahl an Instance-Typen zu erhalten.
 - Erwägen Sie die Übertragung Ihrer Workload zu auf Graviton basierenden Instances, um die Leistungseffizienz Ihrer Workload zu verbessern. Weitere Informationen zum Verschieben von Workloads zu AWS Graviton finden Sie unter [AWS Graviton Fast Start](#) und [Considerations when transitioning workloads to AWS Graviton-based Amazon Elastic Compute Cloud instances](#).
 - Erwägen Sie die Auswahl der Option AWS Graviton, wenn Sie [verwaltete AWS-Services](#) verwenden.
 - Migrieren Sie Ihre Workload zu Regionen mit Instances, die die geringsten nachhaltigkeitsbezogenen Auswirkungen bieten und dennoch Ihre geschäftlichen Anforderungen erfüllen.
 - Nutzen Sie für Machine-Learning-Workloads spezielle Hardware, die auf Ihre Workload abgestimmt ist, z. B. [AWS Trainium](#), [AWS Inferentia](#) und [Amazon EC2 DL1](#). AWS Inferentia-

Instances wie Inf2-Instances bieten eine um bis zu 50 % bessere Leistung pro Watt als vergleichbare Amazon-EC2-Instances.

- Verwenden Sie [Amazon SageMaker AI Inference Recommender](#) für die korrekte Dimensionierung des ML-Inferenz-Endpunkts.
- Verwenden Sie für Workloads, bei denen es gelegentlich zu zusätzlichen Kapazitätsanforderungen kommt, [Instances mit Spitzenlastleistung](#).
- Verwenden Sie für zustandslose und fehlertolerante Workloads [Amazon EC2 Spot Instances](#), um die allgemeine Auslastung der Cloud zu verbessern und die Nachhaltigkeitsauswirkungen ungenutzter Ressourcen zu reduzieren.
- Betrieb und Optimierung: Betreiben und optimieren Sie Ihre Workload-Instance.
 - Prüfen Sie für kurzzeitige Workloads die [Amazon-CloudWatch-Metriken der Instance](#) wie CPUUtilization, um festzustellen, ob die Instance gar nicht oder zu wenig genutzt wird.
 - Prüfen Sie für stabile Workloads in regelmäßigen Intervallen AWS-Dimensionierungstools wie etwa [AWS Compute Optimizer](#), um Möglichkeiten zur Optimierung und zur korrekten Dimensionierung der Instances zu erkennen. Weitere Beispiele und Empfehlungen finden Sie in den folgenden Labs:
 - [Well-Architected Lab - Rightsizing Recommendations](#)
 - [Well-Architected Lab - Rightsizing with Compute Optimizer](#)
 - [Well-Architected Lab - Optimize Hardware Patterns and Observe Sustainability KPIs](#)

Ressourcen

Zugehörige Dokumente:

- [Optimizing your AWS Infrastructure for Sustainability, Part I: Compute](#)
- [AWS Graviton](#)
- [Amazon EC2 DL1](#)
- [Amazon-EC2-Flotten zur Kapazitätsreservierung](#)
- [Amazon-EC2-Spot-Flotte](#)
- [Funktionen: Lambda-Funktionskonfiguration](#)
- [Attributbasierte Auswahl von Instance-Typen für Amazon-EC2-Flotte](#)
- [Building Sustainable, Efficient, and Cost-Optimized Applications on AWS](#)
- [How the Contino Sustainability Dashboard Helps Customers Optimize Their Carbon Footprint](#)

Zugehörige Videos:

- [AWS re:Invent 2.023 - AWS Graviton: The best price performance for your AWS workloads](#)
- [AWS re:Invent 2.023 - New Amazon Elastic Compute Cloud generative AI capabilities in AWS-Managementkonsole](#)
- [AWS re:Invent 2.023 = What's new with Amazon Elastic Compute Cloud](#)
- [AWS re:Invent 2.023 - Smart savings: Amazon Elastic Compute Cloud cost-optimization strategies](#)
- [AWS re:Invent 2.021 - Deep dive into AWS Graviton3 and Amazon EC2 C7g instances](#)
- [AWS re:Invent 2.022 - Build a cost-, energy-, and resource-efficient compute environment](#)

Zugehörige Beispiele:

- [Lösung: Guidance for Optimizing Deep Learning Workloads for Sustainability on AWS](#)

SUS05-BP03 Managed Services nutzen

Verwenden Sie verwaltete Services für effizientere Betriebsabläufe in der Cloud.

Typische Anti-Muster:

- Sie verwenden EC2 Amazon-Instances mit geringer Auslastung, um Ihre Anwendungen auszuführen.
- Ihr internes Team verwaltet nur die Workload, ohne Zeit zu haben, sich auf Innovation oder Vereinfachungen zu konzentrieren.
- Sie nutzen und verwalten Technologien für Aufgaben, die effizienter auf verwalteten Services ausgeführt werden können.

Vorteile der Nutzung dieser bewährten Methode:

- Durch den Einsatz von Managed Services verlagert sich die Verantwortung auf das Unternehmen AWS, das Einblicke in Millionen von Kunden hat, die dazu beitragen können, neue Innovationen und Effizienzsteigerungen voranzutreiben.
- Ein verwalteter Service verteilt die Umweltauswirkungen des Services durch Multi-Tenet-Steuerebenen auf mehrere Benutzer.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Mit Managed Services verlagert sich die AWS Verantwortung auf die Aufrechterhaltung einer hohen Auslastung und die Optimierung der Nachhaltigkeit der eingesetzten Hardware. Verwaltete Services eliminieren dazu den betrieblichen und administrativen Aufwand für die Wartung eines Service, sodass Ihr Team mehr Zeit hat und sich auf Innovationen konzentrieren kann.

Überprüfen Sie Ihren Workload, um die Komponenten zu identifizieren, die durch AWS Managed Services ersetzt werden können. [AmazonRDS](#), [Amazon Redshift](#) und [Amazon ElastiCache](#) bieten beispielsweise einen verwalteten Datenbankservice an. [Amazon Athena](#)EMR, [Amazon](#) und [Amazon OpenSearch Service](#) bieten einen verwalteten Analysedienst.

Implementierungsschritte

1. Inventarisieren Ihrer Workload: Inventarisieren Sie Ihre Workload für Services und Komponenten.
2. Identifizieren von Kandidaten: Bewerten und identifizieren Sie Komponenten, die durch verwaltete Services ersetzt werden können. Hier finden Sie einige Beispiele für Situationen, in denen Sie einen verwalteten Service in Erwägung ziehen sollten:

Aufgabe	Wofür soll ich es verwenden AWS
Hosten einer Datenbank	Verwenden Sie verwaltete Amazon Relational Database Service (AmazonRDS) -Instances, anstatt Ihre eigenen RDS Amazon-Instances auf Amazon Elastic Compute Cloud (AmazonEC2) zu verwalten.
Hosten einer Container-Workload	Verwenden Sie AWS Fargate , anstatt Ihre eigene Container-Infrastruktur zu implementieren.
Hosten von Web-Apps	Verwenden Sie AWS Amplify Hosting als vollständig verwalteten CI/CD- und Hosting-Service für statische Websites und serverseitig gerenderte Web-Apps.

3. Erstellen eines Migrationsplans: Identifizieren Sie Abhängigkeiten und erstellen Sie einen Migrationsplan. Aktualisieren Sie Runbooks und Playbooks entsprechend.

- Der [AWS Application Discovery Service](#) erfasst und präsentiert automatisch detaillierte Informationen über Anwendungsabhängigkeiten und die Nutzung von Anwendungen. Damit treffen Sie bei der Planung Ihrer Migration fundiertere Entscheidungen.
4. Tests: Testen Sie den Service vor der Migration zum verwalteten Service.
 5. Ersetzen selbst gehosteter Services: Verwenden Sie Ihren Migrationsplan, um selbst gehostete Services durch verwaltete Services zu ersetzen.
 6. Überwachen und anpassen: Überwachen Sie den Service nach der Migration kontinuierlich, um erforderliche Anpassungen vorzunehmen und den Service zu optimieren.

Ressourcen

Zugehörige Dokumente:

- [AWS Cloud Produkte](#)
- [AWS Rechner für die Gesamtbetriebskosten \(TCO\)](#)
- [Amazon DocumentDB](#)
- [Amazon Elastic Kubernetes Service \(\) EKS](#)
- [Von Amazon Managed Streaming for Apache Kafka \(AmazonMSK\)](#)

Zugehörige Videos:

- [AWS re:Invent 2021 — Cloud-Betrieb im großen Maßstab mit AWS Managed Services](#)
- [AWS re:Invent 2023 — Bewährte Methoden für den Betrieb auf AWS](#)

SUS05-BP04 Optimieren der Nutzung von hardwarebasierten Datenverarbeitungsbeschleunigern

Sie können die Nutzung von beschleunigten Computing-Instances optimieren, um die Anforderungen Ihrer Workload an die physische Infrastruktur zu reduzieren.

Typische Anti-Muster:

- Sie überwachen die GPU-Nutzung nicht.
- Sie verwenden eine Allzweck-Instance für die Workload, während eine speziell erstellte Instance eine höhere Leistung, geringere Kosten und eine bessere Leistung pro Watt bieten kann.
- Sie verwenden hardwarebasierte Datenverarbeitungsbeschleuniger für Aufgaben, bei denen CPU-basierte Alternativen effizienter sind.

Vorteile der Nutzung dieser bewährten Methode: Durch den optimalen Einsatz hardwarebasierter Beschleuniger können Sie die Anforderungen an die physische Infrastruktur Ihrer Workload reduzieren.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Wenn Sie eine hohe Verarbeitungsleistung benötigen, können Sie beschleunigte Computing-Instances verwenden. Diese bieten Zugriff auf hardwarebasierte Datenverarbeitungsbeschleuniger wie Grafikprozessoren (Graphics Processing Units, GPUs) und Field-Programmable Gate Arrays (FPGAs). Diese Hardwarebeschleuniger führen bestimmte Funktionen wie die Grafikverarbeitung oder Datenmusterzuordnung effizienter aus als CPU-basierte Alternativen. Viele beschleunigte Workloads, wie Rendering, Transkodierung und Machine Learning, sind sehr variabel im Bezug auf die Ressourcennutzung. Betreiben Sie diese Hardware nur so lange wie nötig und nehmen Sie sie automatisch außer Betrieb, wenn sie nicht mehr benötigt wird, um den Ressourcenverbrauch zu minimieren.

Implementierungsschritte

- Ermitteln von Computing-Beschleunigern: Ermitteln Sie die [beschleunigten Computing-Instances](#), die für Ihre Anforderungen geeignet sind.
- Verwenden spezieller Hardware: Verwenden Sie für Machine-Learning-Workloads spezielle Hardware, die auf Ihren Workload abgestimmt ist, z. B. [AWS Trainium](#), [AWS Inferentia](#) und [Amazon EC2 DL1](#). AWS Inferentia-Instances wie Inf2-Instances bieten eine um bis zu 50 % bessere Leistung pro Watt als vergleichbare Amazon-EC2-Instances.
- Überwachen von Nutzungsmetriken: Erfassen Sie Nutzungsmetriken für Ihre beschleunigten Computing-Instances. Sie können den CloudWatch-Agent beispielsweise verwenden, um Metriken wie `utilization_gpu` und `utilization_memory` für Ihre GPUs zu sammeln, wie im [Artikel zum Erfassen von NVIDIA-GPU-Metriken mit Amazon CloudWatch](#) beschrieben.
- Dimensionierung: Optimieren Sie Code, Netzwerkbetrieb und die Einstellungen von Hardwarebeschleunigern, um sicherzustellen, dass die zugrunde liegende Hardware optimal genutzt wird.
 - [Optimieren der GPU-Einstellungen](#)
 - [GPU-Überwachung und -Optimierung im Deep-Learning-AMI](#)
 - [Optimieren von E/A für die GPU-Leistungsoptimierung von Deep-Learning-Training in Amazon SageMaker AI](#)

- Auf dem Laufenden bleiben: Verwenden Sie die neuesten leistungsstarken Bibliotheken und GPU-Treiber.
- Freigeben nicht benötigter Instances: Automatisieren Sie die Freigabe nicht verwendeter GPU-Instances.

Ressourcen

Zugehörige Dokumente:

- [Beschleunigte Datenverarbeitung](#)
- [Let's Architect! Architecting with custom chips and accelerators](#)
- [Wie wähle ich den geeigneten Amazon-EC2-Instance-Typ für meine Workload aus?](#)
- [VT1-Instances von Amazon EC](#)
- [Choose the best AI accelerator and model compilation for computer vision inference with Amazon SageMaker AI](#)

Zugehörige Videos:

- [AWS re:Invent 2.021 - How to select Amazon EC2 GPU instances for deep learning](#)
- [AWS Online Tech Talks - Deploying Cost-Effective Deep Learning Inference](#)
- [AWS re:Invent 2.023 - Cutting-edge AI with AWS and NVIDIA](#)
- [AWS re:Invent 2.022 - \[NEW LAUNCH!\] Introducing AWS Inferentia2-based Amazon EC2 Inf2 instances](#)
- [AWS re:Invent 2.022 - Accelerate deep learning and innovate faster with AWS Trainium](#)
- [AWS re:Invent 2.022 - Deep learning on AWS with NVIDIA: From training to deployment](#)

Prozess und Kultur

Frage

- [SUS 6 Wie unterstützen Ihre betrieblichen Prozesse Ihre Nachhaltigkeitsziele?](#)

SUS 6 Wie unterstützen Ihre betrieblichen Prozesse Ihre Nachhaltigkeitsziele?

Reduzieren Sie nachhaltigkeitsbezogene Auswirkungen, indem Sie Ihre Entwicklungs-, Test- und Bereitstellungsmethoden ändern.

Best Practices

- [SUS06-BP01 Kommunizieren und Kaskadieren Ihrer Nachhaltigkeitsziele](#)
- [SUS06-BP02 Einführen von Methoden, die schnelle Verbesserungen für die Nachhaltigkeit ermöglichen](#)
- [SUS06-BP03 Konstantes Aktualisieren Ihrer Workload](#)
- [SUS06-BP03 Höhere Auslastung von Entwicklungsumgebungen](#)
- [SUS06-BP04 Verwenden von verwalteten Gerätefarmen für Tests](#)

SUS06-BP01 Kommunizieren und Kaskadieren Ihrer Nachhaltigkeitsziele

Technologie ist ein wichtiger Faktor, wenn es um die Erzielung von Nachhaltigkeit geht. IT-Teams spielen eine kritische Rolle bei der Einführung relevanter Veränderungen hinsichtlich der Nachhaltigkeitsziele Ihrer Organisation. Diese Teams sollten die Nachhaltigkeitsziele des Unternehmens genau verstehen und diese Prioritäten an alle Geschäftsbereiche kommunizieren und kaskadieren.

Typische Anti-Muster:

- Sie kennen die Nachhaltigkeitsziele Ihrer Organisation nicht und wissen nicht, inwiefern sie für Ihr Team relevant sind.
- Sie sind sich der ökologischen Auswirkungen von Cloud-Workloads nicht ausreichend bewusst und dahingehend nicht ausreichend geschult.
- Sie sind sich nicht sicher, welche spezifischen Bereiche Sie priorisieren sollten.
- Sie beziehen Ihre Mitarbeiter und Kunden nicht in Ihre Nachhaltigkeitsinitiativen ein.

Vorteile der Einführung dieser bewährten Methode: Von der Optimierung von Infrastruktur und Systemen bis zum Einsatz innovativer Technologien – IT-Teams können die CO2-Emissionen der Organisation reduzieren und den Ressourcenverbrauch minimieren. Die Kommunikation von Nachhaltigkeitszielen kann IT-Teams die Möglichkeit bieten, kontinuierliche Verbesserungen einzuführen und Anpassungen an neue Herausforderungen im Bereich der Nachhaltigkeit

vorzunehmen. Darüber hinaus führen diese nachhaltigen Optimierungen häufig auch zu Kosteneinsparungen, was ein weiterer Vorteil ist.

Risikostufe bei fehlender Befolgung dieser bewährten Methode: Mittel

Implementierungsleitfaden

Die primären Nachhaltigkeitsziele für IT-Teams sollten in der Optimierung von Systemen und Lösungen bestehen, um die Ressourceneffizienz zu steigern und gleichzeitig die CO2-Emissionen und allgemeinen ökologischen Auswirkungen der Organisation zu minimieren. Die Freigabe von Services und Initiativen wie Schulungsprogramme und operative Dashboards können Organisationen dabei unterstützen, den IT-Betrieb zu optimieren und Lösungen für eine deutliche Reduzierung der CO2-Emissionen zu entwickeln. Die Cloud ermöglicht nicht nur die Verlagerung der Verantwortung für physische Infrastruktur und Energiebeschaffung in den Verantwortungsbereich des Cloud-Anbieters, sondern auch die kontinuierliche Optimierung der Ressourceneffizienz cloudbasierter Services.

Wenn Teams die inhärente Effizienz der Cloud und das Modell der geteilten Verantwortung nutzen, können sie die ökologischen Auswirkungen der Organisation auf relevante Weise reduzieren. Dies kann in der Folge zu den allgemeinen Nachhaltigkeitszielen der Organisation beitragen und den Wert dieser Teams als strategische Partner auf dem Weg zu einer nachhaltigeren Zukunft belegen.

Implementierungsschritte

- Definieren von Zielen und Vorgaben: Legen Sie gut definierte Ziele für Ihr IT-Programm fest. Dazu gehört das Einholen von Feedback verantwortlicher Stakeholder aus verschiedenen Bereichen wie IT, Nachhaltigkeit und Finanzen. Diese Teams sollten messbare Ziele entsprechend den Nachhaltigkeitszielen Ihrer Organisation definieren, einschließlich Bereichen wie Reduzierung der CO2-Emission und Ressourcenoptimierung.
- Verstehen der CO2-Bilanzierung für Ihr Unternehmen: Informieren Sie sich über die Auswirkungen von Methoden für die CO2-Bilanzierung auf Ihre Workloads in der Cloud, z. B. des Greenhouse Gas (GHG)-Protokolls. (Weitere Informationen finden Sie unter [Cloud-Nachhaltigkeit](#)).
- Verwenden von Cloud-Lösungen für die CO2-Bilanzierung: Verwenden Sie Cloud-Lösungen wie [CO2-Bilanzierungslösungen in AWS](#), um die Treibhausgasemissionen in Scope 1, 2 und 3 in Operations, Portfolien und Wertschöpfungsketten zu erfassen. Mit diesen Lösungen können Organisationen die Erfassung von Treibhausgasemissionsdaten optimieren, Berichte vereinfachen und Erkenntnisse für ihre Klimaschutzstrategien ableiten.
- Überwachen der CO2-Emissionen Ihres IT-Portfolios: Verfolgen Sie die CO2-Emissionen Ihrer IT-Systeme und erstellen Sie entsprechende Berichte. Mit dem [AWS-Tool für die Erfassung der CO2-](#)

[Emissionen von Kunden](#) können Sie die CO2-Emissionen nachverfolgen, messen, überprüfen und prognostizieren, die durch Ihre Nutzung von AWS verursacht werden.

- Kommunizieren der Ressourcennutzung anhand von Proxymetriken für Ihre Teams: Verfolgen Sie Ihre [Ressourcennutzung anhand von Proxymetriken](#) und erstellen Sie entsprechende Berichte. Bei den On-Demand-Preismodellen der Cloud ist die Ressourcennutzung mit den Kosten verbunden, eine allgemein verständliche Metrik. Sie sollten die Kosten mindestens als Proxymetrik verwenden, um Ressourcennutzung und Verbesserungen durch die einzelnen Teams zu kommunizieren.
- Aktivieren der Granularität auf Stundenbasis im Cost Explorer und Erstellen eines [Kosten- und Nutzungsberichts \(CUR\)](#): Der CUR stellt Daten zur Nutzungsgranularität auf Tages- oder Stundenbasis, Tarifen, Kosten und Nutzung für alle AWS-Services bereit. Verwenden Sie die [Cloud Intelligence Dashboards](#) und das zugehörige Dashboard für Nachhaltigkeits-Proxymetriken als Ausgangspunkt für die Verarbeitung und Visualisierung kosten- und nutzungsbasierter Daten. Weitere Informationen finden Sie im Folgenden:
- [Messen und Verfolgen der Cloud-Effizienz mit Proxymetriken für Nachhaltigkeit, Teil I: Was sind Proxymetriken?](#)
- [Messen und Verfolgen der Cloud-Effizienz mit Proxymetriken für Nachhaltigkeit, Teil II: Einrichten einer Metrik-Pipeline](#)
- Kontinuierliches Optimieren und Bewerten: Verwenden Sie einen [Verbesserungsprozess](#), um Ihre IT-Systeme (einschließlich Cloud-Workloads) kontinuierlich im Hinblick auf Effizienz und Nachhaltigkeit zu optimieren. Überwachen Sie die CO2-Emissionen vor und nach der Implementierung der Optimierungsstrategie. Bewerten Sie die Effektivität anhand der Reduzierung der CO2-Emissionen.
- Fördern einer Nachhaltigkeitskultur: Nutzen Sie Schulungsprogramme (z. B. [AWS Skill Builder](#)), um Ihre Mitarbeiter über Nachhaltigkeit aufzuklären. Binden Sie Ihre Mitarbeiter in Nachhaltigkeitsinitiativen ein. Teilen und feiern Sie ihre Erfolgsgeschichten. Nutzen Sie Anreize, um sie auszuzeichnen, wenn sie Nachhaltigkeitsziele erreichen.

Ressourcen

Zugehörige Dokumente:

- [Grundlegendes zu CO2-Emissionsschätzungen](#)

Zugehörige Videos:

- [AWS re:Invent 2023 – Accelerate data-driven circular economy initiatives with AWS](#)

- [AWS re:Invent 2023 – Sustainability innovation in AWS Global Infrastructure](#)
- [AWS re:Invent 2.023 – Sustainable architecture: Past, present, and future](#)
- [AWS re:Invent 2.022 – Delivering sustainable, high-performing architectures](#)
- [AWS re:Invent 2022 – Architecting sustainably and reducing your AWS carbon footprint](#)
- [AWS re:Invent 2.022 – Sustainability in AWS global infrastructure](#)

Zugehörige Beispiele:

- [Well-Architected Lab – Kosten- und Nutzungsberichte in Effizienzberichte umwandeln](#)

Zugehörige Schulungen:

- [Nachhaltigkeits-Transformation in AWS](#)
- [SimuLearn – Nachhaltigkeitsberichte](#)
- [Dekarbonisierung mit AWS](#)

SUS06-BP02 Einführen von Methoden, die schnelle Verbesserungen für die Nachhaltigkeit ermöglichen

Nutzen Sie Methoden und Prozesse zur Validierung potenzieller Verbesserung, zur Minimierung von Testkosten und zur Bereitstellung kleinerer Verbesserungen.

Typische Anti-Muster:

- Die Prüfung Ihrer Anwendung auf Nachhaltigkeitsaspekte erfolgt nur einmal zu Beginn des Projekts.
- Ihre Workload stagniert, da der Freigabeprozess zu komplex ist, um kleinere Verbesserungen für die Ressourceneffizienz umzusetzen.
- Sie verfügen über keine Mechanismen zur Verbesserung Ihrer Workload unter Nachhaltigkeitsaspekten.

Vorteile der Nutzung dieser bewährten Methode: Durch die Einrichtung eines Prozesses für die Einführung und Nachverfolgung von Nachhaltigkeitsverbesserungen können Sie kontinuierlich neue Funktionen einführen, Probleme beseitigen und die Workload-Effizienz verbessern.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Mittel

Implementierungsleitfaden

Testen und validieren Sie potenzielle Verbesserungen in Bezug auf die Nachhaltigkeit, bevor Sie sie in der Produktion bereitstellen. Berücksichtigen Sie die Testkosten bei der Berechnung des potenziellen zukünftigen Nutzens einer Verbesserung. Entwickeln Sie kostengünstige Testmethoden, um kleinere Verbesserungen einzuführen.

Implementierungsschritte

- Kenntnis und Kommunikation der Nachhaltigkeitsziele Ihrer Organisation: Machen Sie sich mit den Nachhaltigkeitszielen Ihrer Organisation vertraut, z. B. zur Reduzierung der CO₂-Emissionen oder zum verantwortungsvollen Umgang mit Wasser. Übersetzen Sie diese Ziele in Nachhaltigkeitsanforderungen für Ihre Cloud-Workloads. Kommunizieren Sie diese Anforderungen an wichtige Stakeholder.
- Ergänzung des Backlogs mit Nachhaltigkeitsanforderungen: Fügen Sie Ihrem Entwicklungs-Backlog Anforderungen zur Verbesserung der Nachhaltigkeit hinzu.
- Iterieren und verbessern: Verwenden Sie einen [iterativen Verbesserungsprozess](#), um diese Verbesserungen zu identifizieren, zu bewerten, zu priorisieren, zu testen und bereitzustellen.
- Tests unter Verwendung des Minimum Viable Product (MVP): Entwickeln und testen Sie potenzielle Verbesserungen unter Verwendung der Minimum-Viable-Komponenten, um die Kosten und die Umweltauswirkungen der Tests zu reduzieren.
- Prozess optimieren: Verbessern und optimieren Sie Ihre Entwicklungsprozesse kontinuierlich. Sie können beispielsweise Ihren Softwarebereitstellungsprozess mit Pipelines für die Continuous Integration und Continuous Delivery (CI/CD) automatisieren, um potenzielle Verbesserungen zu testen und bereitzustellen und so den Aufwand zu reduzieren und Fehler durch manuelle Prozesse zu minimieren.
- Schulung und Sensibilisierung: Führen Sie Schulungsprogramme für Ihre Teammitglieder durch, um sie über Nachhaltigkeit und die Auswirkungen ihrer Aktivitäten auf die Nachhaltigkeitsziele Ihrer Organisation aufzuklären.
- Bewerten und anpassen: Bewerten Sie kontinuierlich die Auswirkungen von Verbesserungen und nehmen Sie bei Bedarf Anpassungen vor.

Ressourcen

Zugehörige Dokumente:

- [AWS ermöglicht nachhaltige Lösungen](#)

Zugehörige Videos:

- [AWS re:Invent 2.023 - Sustainable architecture: Past, present, and future](#)
- [AWS re:Invent 2.022 - Delivering sustainable, high-performing architectures](#)
- [AWS re:Invent 2.022 - Architecting sustainably and reducing your AWS carbon footprint](#)
- [AWS re:Invent 2.022 – Sustainability in AWS global infrastructure](#)
- [AWS re:Invent 2023 - What's new with AWS observability and operations](#)

SUS06-BP03 Konstantes Aktualisieren Ihrer Workload

Halten Sie Ihre Workload auf neustem Stand, um effiziente Features zu übernehmen, Probleme zu beseitigen und die allgemeine Effizienz der Workload zu wahren.

Typische Anti-Muster:

- Sie gehen davon aus, dass Ihre aktuelle Architektur statisch ist und im Laufe der Zeit nicht aktualisiert wird.
- Sie haben keine Systeme oder regelmäßigen Besprechungen zur Prüfung, ob aktualisierte Software und Pakete mit Ihrer Workload kompatibel sind.

Vorteile der Nutzung dieser bewährten Methode: Wenn Sie einen Prozess einrichten, um Ihre Workload auf neustem Stand zu halten, können Sie neue Features und Kapazitäten nutzen, Probleme lösen und die Workload-Effizienz verbessern.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Niedrig

Implementierungsleitfaden

Aktuelle Betriebssysteme, Laufzeiten, Middleware, Bibliotheken und Anwendungen können die Workload-Effizienz verbessern und die Nutzung effizienterer Technologien unterstützen. Aktuelle Software kann darüber hinaus Features für eine genauere Messung der Auswirkungen Ihrer Workload bereitstellen, da die Anbieter mit ihren Features ebenfalls Nachhaltigkeitsziele erfüllen müssen. Sorgen Sie für Regelmäßigkeit bei der Aktualisierung Ihrer Workload mit den neuesten Features und Versionen.

Implementierungsschritte

- Definieren eines Prozesses: Definieren Sie einen Prozess und einen Zeitplan, um neue Features oder Instances für Ihre Workloads zu evaluieren. Nutzen Sie die Agilität in der Cloud, um schnell zu testen, wie neue Features Ihre Workloads verbessern können:
 - Reduzierung von Auswirkungen auf die Nachhaltigkeit
 - Erzielen von Leistungseffizienzen
 - Beseitigen von Hindernissen für geplante Verbesserungen
 - Verbesserung Ihrer Fähigkeit für die Messung von und den Umgang mit Nachhaltigkeitsauswirkungen
- Inventarisierung: Inventarisieren Sie Ihre Workload-Software und -Architektur und identifizieren Sie Komponenten, die aktualisiert werden müssen.
 - Sie können [AWS Systems Manager Inventory](#) verwenden, um Betriebssystem-, Anwendungs- und Instance-Metadaten von Ihren Amazon-EC2-Instances zu erfassen und so schnell zu verstehen, welche Instances die Software und die Konfigurationen ausführen, die Ihre Softwarerichtlinie erfordert, und welche Instances aktualisiert werden müssen.
- Kennenlernen des Aktualisierungsverfahrens: Erfahren Sie, wie die Komponenten Ihrer Workload aktualisiert werden.

Workload-Komponente	Aktualisierung
Machine Images	Verwenden Sie EC2 Image Builder zur Verwaltung von Updates für Amazon Machine Images (AMIs) für Linux- oder Windows Server-Images.
Container-Images	Verwenden Sie Amazon Elastic Container Registry (Amazon ECR) mit Ihrer vorhandenen Pipeline zur Verwaltung von Amazon Elastic Container Service (Amazon ECS)-Images .
AWS Lambda	AWS Lambda enthält Features zur Versionsverwaltung .

- Verwendung von Automatisierung: Verwenden Sie Automatisierung für den Aktualisierungsvorgang, um den Aufwand für die Bereitstellung neuer Features zu reduzieren und Fehler zu begrenzen, die durch manuelle Prozesse verursacht werden.
- Sie können [CI/CD](#) verwenden, um AMIs, Container-Images und andere Artefakte im Zusammenhang mit Ihrer Cloud-Anwendung automatisch zu aktualisieren.
- Sie können Tools wie [AWS Systems Manager Patch Manager](#) verwenden, um den Systemaktualisierungsprozess zu automatisieren, und die Aktivität mit [AWS Systems Manager Maintenance Windows](#) planen.

Ressourcen

Zugehörige Dokumente:

- [AWS-Architekturzentrum](#)
- [Neuerungen bei AWS](#)
- [AWS-Entwicklungstools](#)

Zugehörige Videos:

- [AWS re:Invent 2.022 - Optimize your AWS workloads with best-practice guidance](#)
- [All Things Patch: AWS Systems Manager](#)

SUS06-BP03 Höhere Auslastung von Entwicklungsumgebungen

Erhöhen Sie die Ausnutzung von Ressourcen zum Entwickeln, Testen und Erstellen Ihrer Workloads.

Typische Anti-Muster:

- Sie stellen Ihre Build-Umgebungen manuell bereit oder beenden sie manuell.
- Sie lassen Ihre Build-Umgebungen unabhängig von Test-, Build- oder Freigabeaktivitäten laufen (dazu gehört etwa der Betrieb einer Umgebung außerhalb der Arbeitszeit der Mitglieder Ihres Entwicklungsteams).
- Sie stellen übermäßig viele Ressourcen für Ihre Build-Umgebung bereit.

Vorteile der Nutzung dieser bewährten Methode: Durch die Steigerung der Ausnutzung von Build-Umgebungen können Sie die allgemeine Effizienz Ihrer Cloud-Workload verbessern, da die Ressourcen in effizienter Weise Entwicklungs-, Test- und Build-Aktivitäten zugewiesen werden.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Niedrig

Implementierungsleitfaden

Verwenden Sie Automatisierung und Infrastructure as Code, um Build-Umgebungen in Betrieb zu nehmen, wenn sie gebraucht werden, und sie andernfalls zu deaktivieren. Eine typische Vorgehensweise besteht in der Planung von Verfügbarkeitszeiten, die mit den Arbeitszeiten der Entwicklungsteams übereinstimmen. Ihre Testumgebungen sollten der Produktionskonfiguration sehr stark ähneln. Suchen Sie aber nach Möglichkeiten, Instance-Typen mit Burst-Kapazität, Amazon EC2 Spot Instances, automatisch skalierenden Datenbankservices, Containern und Serverless-Technologien zu verwenden, um die Entwicklungs- und Testkapazität an der Nutzung auszurichten. Begrenzen Sie das Datenvolumen auf die Testanforderungen. Wenn Sie Produktionsdaten für einen Test verwenden, sollten Sie nach Möglichkeiten suchen, Daten aus der Produktion gemeinsam zu nutzen, anstatt Daten hin- und herzuschieben.

Implementierungsschritte

- Infrastructure as Code verwenden: Verwenden Sie Infrastructure as Code, um Ihre Entwicklungsumgebungen bereitzustellen.
- Automatisierung verwenden: Nutzen Sie Automatisierungen, um den Lebenszyklus Ihrer Entwicklungs- und Testumgebungen zu verwalten und die Effizienz Ihrer Entwicklungsressourcen zu maximieren.
- Nutzung maximieren: Verwenden Sie Strategien zur Maximierung der Nutzung von Entwicklungs- und Testumgebungen.
 - Verwenden Sie die geringstmögliche Zahl repräsentativer Umgebungen, um mögliche Verbesserungen zu entwickeln und zu testen.
 - Nutzen Sie nach Möglichkeit Serverless-Technologien.
 - Verwenden Sie On-Demand-Instances, um Entwicklergeräte zu ergänzen.
 - Verwenden Sie Instance-Typen mit Burst-Kapazität, Spot Instances und andere Technologien, um die Entwicklungskapazität an der Nutzung auszurichten.
 - Nutzen Sie native Cloud-Services für den sicheren Instance-Shell-Zugriff, statt Bastion-Host-Flotten bereitzustellen.
 - Skalieren Sie Ihre Build-Ressourcen automatisch je nach Build-Aktivität.

Ressourcen

Zugehörige Dokumente:

- [AWS Systems Manager Session Manager](#)
- [Amazon EC2 Burstable performance instances](#)
- [Was ist AWS CloudFormation?](#)
- [Was ist AWS CodeBuild?](#)
- [Instance Scheduler in AWS](#)

Zugehörige Videos:

- [AWS re:Invent 2.023 - Continuous integration and delivery for AWS](#)

SUS06-BP04 Verwenden von verwalteten Gerätefarmen für Tests

Verwenden Sie verwaltete Gerätefarmen zum effektiven Testen neuer Features auf einer repräsentativen Auswahl von Hardwaregeräten.

Typische Anti-Muster:

- Sie testen Ihre Anwendung manuell und stellen sie auf einzelnen physischen Geräten bereit.
- Sie verwenden keinen App-Testservice zum Testen und zum Interagieren mit Ihren Apps (beispielsweise Android, iOS und Web-Apps) auf realen physischen Geräten.

Vorteile der Nutzung dieser bewährten Methode: Die Verwendung verwalteter Gerätefarmen zum Testen cloud-fähiger Anwendungen bringt eine Reihe von Vorteilen mit sich:

- Dazu gehören effizientere Features zum Testen von Anwendungen auf einer breiten Palette von Geräten.
- Sie machen hausinterne Infrastruktur zum Testen überflüssig.
- Sie bieten unterschiedliche Gerätetypen, darunter ältere und weniger verbreitete Hardware, was unnötige Geräte-Upgrades eliminiert.

Risikostufe, wenn diese bewährte Methode nicht eingeführt wird: Niedrig

Implementierungsleitfaden

Die Verwendung verwalteter Gerätefarmen kann Ihnen dabei helfen, Ihre Testprozesse für neue Features auf einer repräsentativen Auswahl von Hardwaregeräten zu optimieren. Verwaltete Gerätefarmen stellen verschiedene Gerätetypen bereit, unterstützen auch ältere und weniger verbreitete Hardware und vermeiden nachhaltigkeitsbezogene Auswirkungen auf Kunden durch unnötige Geräte-Upgrades.

Implementierungsschritte

- Testanforderungen definieren: Definieren Sie Ihre Testanforderungen und Ihren Testplan (z. B. Testtyp, Betriebssysteme und Testzeitplan).
 - Sie können [Amazon CloudWatch RUM](#) verwenden, um clientseitige Daten zu erfassen und zu analysieren und Ihren Testplan zu entwerfen.
- Verwaltete Gerätefarm auswählen: Wählen Sie eine verwaltete Gerätefarm, die Ihre Testanforderungen unterstützen kann. Sie können beispielsweise [AWS Device Farm](#) verwenden, um die Auswirkungen Ihrer Änderungen auf eine repräsentative Auswahl von Hardwaregeräten zu testen und zu verstehen.
- Automatisierung verwenden: Verwenden Sie Automatisierung und kontinuierliche Integration/ Bereitstellung (CI/CD) für die Planung und Durchführung Ihrer Tests.
 - [AWS Device Farm mit Ihrer CI/CD-Pipeline zur Durchführung browserübergreifender Selenium-Tests integrieren](#)
 - [Erstellen und Testen von iOS- und iPadOS-Apps mit AWS DevOps und mobilen Services](#)
- Prüfen und Anpassen: Prüfen Sie kontinuierlich Ihre Testergebnisse und nehmen Sie die erforderlichen Verbesserungen vor.

Ressourcen

Zugehörige Dokumente:

- [AWS Device Farm-Geräteliste](#)
- [Das CloudWatch-RUM-Dashboard anzeigen](#)

Zugehörige Videos:

- [AWS re:Invent 2.023 - Improve your mobile and web app quality using AWS Device Farm](#)

- [AWS re:Invent 2.021 - Optimize applications through end user insights with Amazon CloudWatch RUM](#)

Zugehörige Beispiele:

- [AWS Device Farm-Beispiel-App für Android](#)
- [AWS Device Farm-Beispiel-App für iOS](#)
- [Appium-Web-Tests für AWS Device Farm](#)

Hinweise

Kunden sind dafür verantwortlich, Ihre eigene unabhängige Bewertung der Informationen in diesem Dokument vorzunehmen. Dieses Dokument: (a) dient nur zu Informationszwecken, (b) stellt die aktuellen AWS-Produktangebote und -praktiken dar, die ohne Vorankündigung geändert werden können, und (c) begründet keine Verpflichtungen oder Zusicherungen seitens AWS und der mit ihr verbundenen Unternehmen, Lieferanten oder Lizenzgeber. AWS-Produkte oder -Services werden ohne Mängelgewähr bereitgestellt, ohne ausdrückliche oder stillschweigende Garantien, Zusicherungen oder Bedingungen jeglicher Art. Die Verantwortung und Haftung von AWS gegenüber seinen Kunden werden durch AWS-Vereinbarungen geregelt. Dieses Dokument gehört, weder ganz noch teilweise, nicht zu den Vereinbarung von AWS mit seinen Kunden und ändert diese Vereinbarungen auch nicht.

Copyright © 2024 Amazon Web Services, Inc. bzw. Tochtergesellschaften des Unternehmens.

AWS Glossar

Die neueste AWS Terminologie finden Sie im [AWS Glossar](#) in der AWS-Glossar Referenz.