AWS Whitepaper

Establishing Your Cloud Foundation on AWS

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

Establishing Your Cloud Foundation on AWS: AWS Whitepaper

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

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

Table of Contents

..... vi
Abstract and introduction 1
Abstract 1
Introduction 1
Capabilities 3
Capabilities definitions 4
Governance, Risk Management, and Compliance 5
Operations 7
Security 8
Business Continuity 10
Finance 11
Infrastructure 12
Working with the capabilities 14
Identity Management and Access Control capability 16
Overview 17
Structure your identities to access your environment 17
Define functions and responsibilities to manage your environment 18
Federated access 20
Delegate the administration of different services 21
Establish preventive controls across your environment 21
Log Storage capability 21
Overview 22
Benefits of Centralized logs 23
Log strategy 24
Tagging capability 25
Overview 26
Choosing tags for your environment 28
Tagging standards 30
Governance capability 32
Overview 33
Establish the relationship with your cloud services provider 33
Define how cloud services are adopted 34
Get started with your cloud services provider 34
Build cloud capability across your organization 35

Respond to growth or change ... 37
Industry-specific governance ... 38
Security Assurance on AWS ... 38
Workload Isolation capability ... 41
Overview ... 42
Design isolated resource environments ... 42
Provision processes for isolated environments ... 43
Implement controls on isolated resource environments ... 44
Provision baseline standards to isolated resource environments ... 44
Provision pre-approved deployable architectures ... 45
Decommission process for isolated resource environments ... 45
Network Connectivity capability ... 46
Overview ... 47
Connectivity within the cloud environment ... 48
Network designing and planning ... 49
Centralized or distributed network configuration and management ... 51
Hybrid environment set up ... 52
Establish logging and monitoring set up ... 53
Construct DNS set up for applications ... 54
Change Management capability ... 55
Overview ... 56
Establish a change management process ... 57
Change Management categories and priorities ... 59
Define change management fulfillment process ... 62
Recover from failed changes ... 63
Establish mechanisms to access, review, and monitor changes ... 63
Cloud Financial Management capability ... 63
Overview ... 65
Cost allocation ... 66
Planning and forecasting ... 66
Cost monitoring and reporting ... 67
Cost optimization ... 68
Cloud financial operations ... 68
Next steps ... 70
Conclusion ... 71
Contributors ... 72

Further reading 73
Document revisions 74
Appendix A: Capability structure and example 75
Capability Structure 75
Capability example - Log Storage 75
Definition 75
Scenarios 76
Guidance 76
Appendix B: Sample timeline 78
Notices 79
AWS Glossary 80

This whitepaper is for historical reference only. Some content might be outdated and some links might not be available.

Establishing Your Cloud Foundation on AWS

Publication date: **September 13, 2023** (*Document revisions*)

Abstract

The increasing breadth and depth of cloud services makes the cloud a powerful enabler of efficiency, agility, and rapid innovation. However, building a foundational AWS Cloud environment requires decisions across multiple AWS and partner products, services, and solutions. Customers are looking for guidance to help them set up and operate an environment that is compatible with their IT practices, and enables their builders and operators while adhering to governance requirements.

This whitepaper introduces a guided path approach to help customers build and evolve their AWS Cloud environment based on a consolidated set of definitions, scenarios, guidance, and automations. The approach includes people, process, and technology considerations of establishing an AWS Cloud environment.

Introduction

The primary business drivers behind moving to the cloud include greater agility, innovation, and scale. When planning a cloud adoption strategy, the number of decisions that you need to make to stand up a production-ready cloud environment is significant. Decisions that are made early on can affect your ability to enhance and/or scale your environment in the future. This complexity has led customers to look for prescriptive guidance across the range of AWS services that can be used to create a foundational environment.

Establishing a cloud foundation on AWS requires guidance tailored to your business needs. Using a capability-based approach, you can create an environment to deploy, operate, and govern your workloads. You can also enhance the capabilities to extend your environment as your requirements evolve and you deploy additional workloads to the cloud.

Building a foundational environment on AWS can be done with a standard, prescriptive set of capabilities across different functional areas. These capabilities can be used as a structured way to quickly build or expand your AWS Cloud environment, and include scenarios and corresponding guidance.

You can adopt and implement capabilities according to your operational and governance needs. As your business requirements mature, the capability-based approach can be used as a mechanism to

verify that your cloud environment is ready to support your workloads and scale as needed. This approach enables you to confidently establish your cloud environment for your builders and your business.

Capabilities

To support cloud adoption, AWS recommends that you have a foundational set of capabilities that enable you to deploy, operate, and govern your workloads.

A *capability* includes a definition, scenarios, opinionated guidance, and supporting automation to establish and operate a specific part of a cloud environment. Capabilities are components that can help you plan, implement, and operate your cloud environment, and include *people*, *process*, and *technology* considerations. Capabilities are designed to integrate into your overall technology environment.

In addition to technology implementation guidance, capabilities include operational guidance (for instance, notifications, event handling, and remediation, as well as team resource skills and processes) needed to stand up and operate each capability. For an example of what a capability should offer, refer to Appendix A.

AWS has defined a set of 29 capabilities that span six categories to help you establish a cloud foundation.

Table 1 - Cloud Foundations capabilities by categories

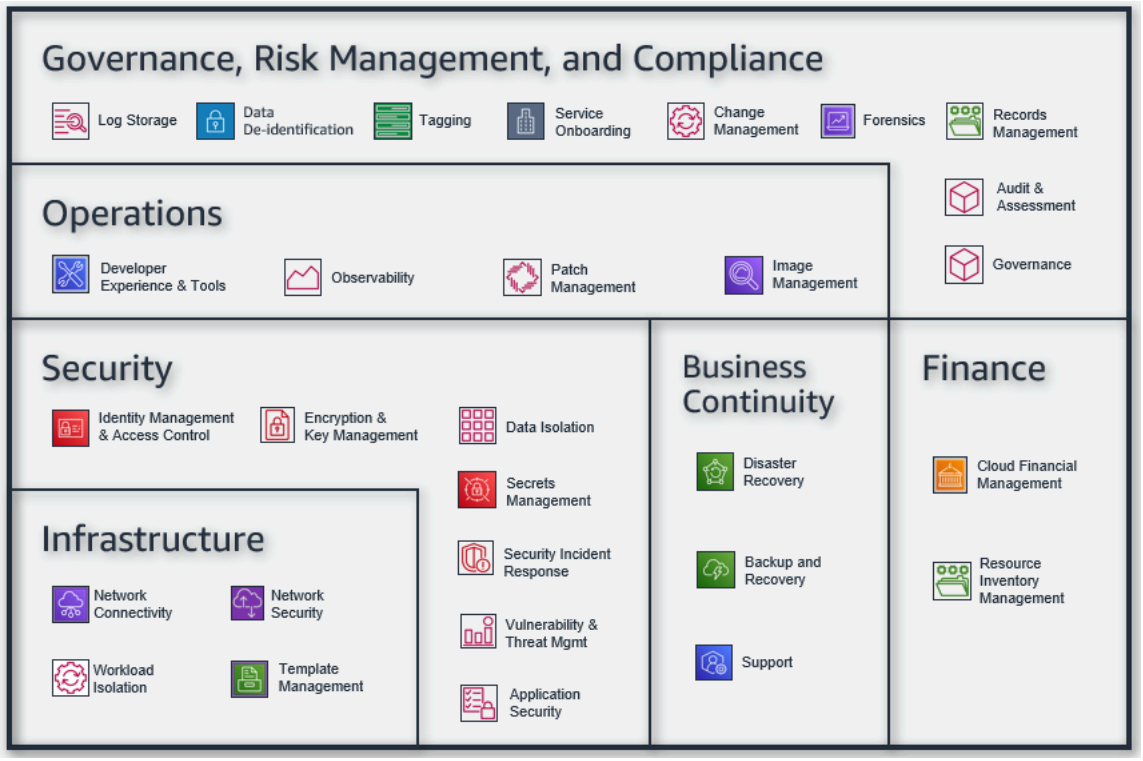
Governance, Risk, and Compliance	Security	Operations	Infrastructure	Finance	Business Continuity
Log Storage	Identity Management & Access Control	Developer Experience & Tools	Network Connectivity	Cloud Financial Management	Backup & Recovery
Governance	Secrets Management	Image Management	Network Security	Resource Inventory Management	Disaster Recovery
Audit & Assessment	Security Incident Response	Observability	Workload Isolation	Support	

Governance, Risk, and Compliance	Security	Operations	Infrastructure	Finance	Business Continuity
Tagging	Encryption & Key Management	Patch Management	Template Management		
Service Onboarding	Vulnerability & Threat Management				
Change Management	Application Security				
Forensics	Data Isolation				
Data De-identification					
Records Management					

Each capability includes stages of maturity that enable you to implement based on where you are in your cloud journey, including your governance and operational requirements. As your cloud environment grows and matures, the *capabilities* can be enhanced to meet your new requirements.

Capabilities definitions

This section includes high-level definitions for each foundational capability organized by their category. For a deeper dive into a specific capability and what it includes, refer to Appendix A.



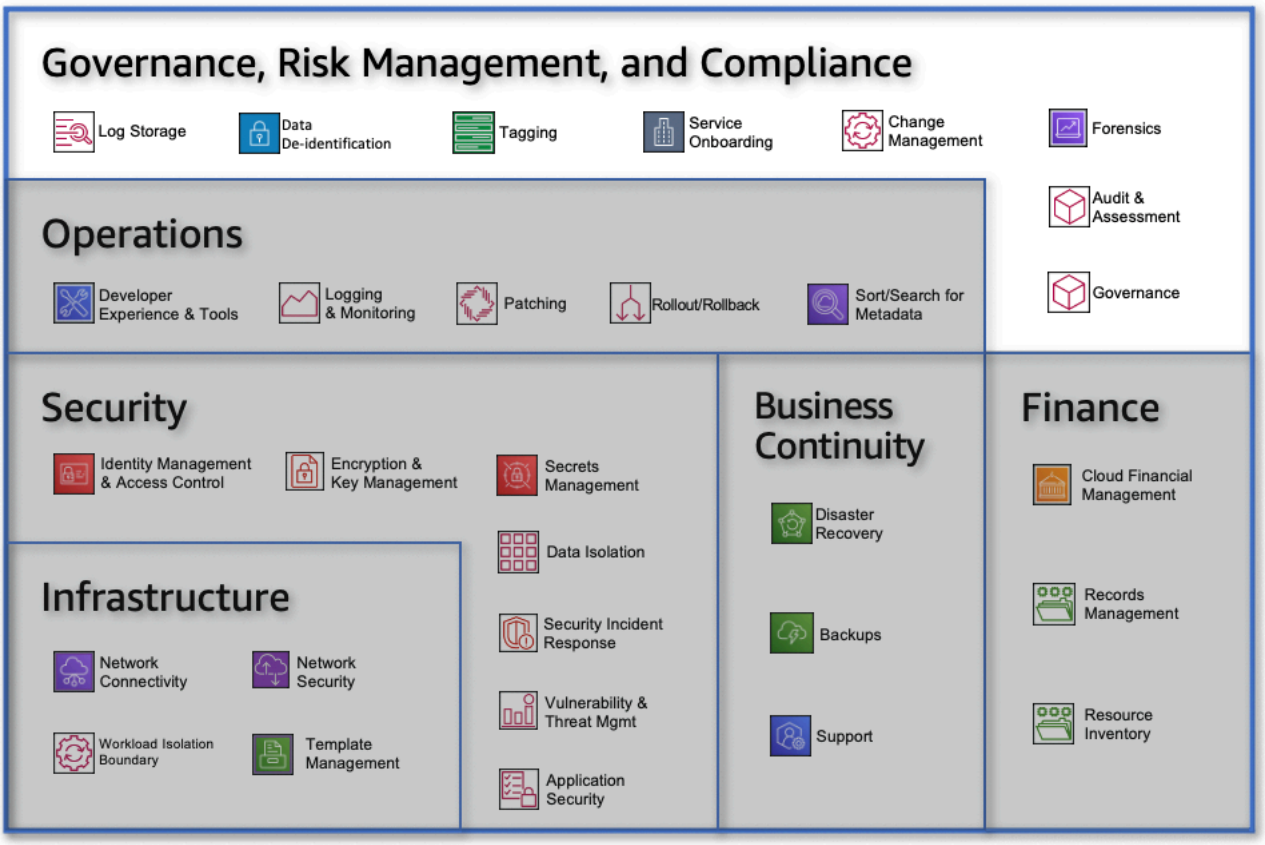
29 capabilities across six categories

Topics

- Governance, Risk Management, and Compliance
- Operations
- Security
- Business Continuity
- Finance
- Infrastructure

Governance, Risk Management, and Compliance

Governance, Risk Management, and Compliance (GRC) helps organizations set the foundation for meeting security and compliance requirements and define the overall policies your cloud environment should adhere to. The capabilities within this area help you define what needs to happen, defines your risk appetite, and informs alignment of internal policies.



Governance, Risk Management, and Compliance Category

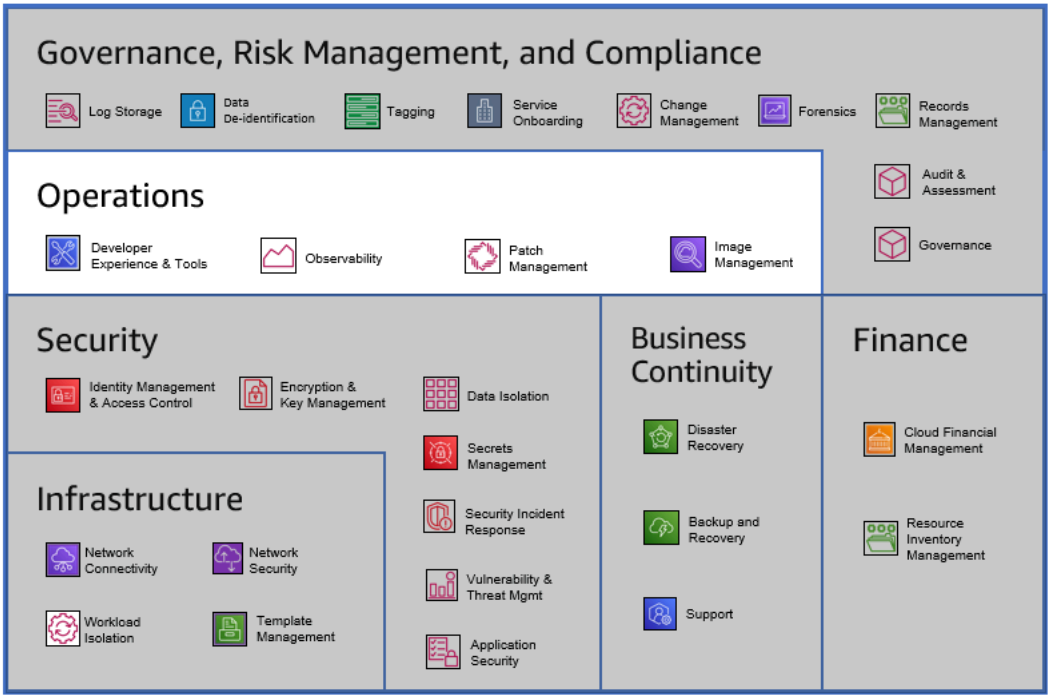
Governance, Risk Management, and Compliance capabilities include:

- **Tagging** enables you to group sets of resources by assigning metadata to cloud resources for a variety of purposes. These purposes include access control (such as ABAC), cost reporting, and automation (such as patching for select tagged instances). Tagging can also be used to create new resource constructs for visibility or control (such as grouping together resources that make up a microservice, application, or workload). Tagging is fundamental to providing enterprise-level visibility and control.
- **Log storage** enables you to collect and store your environment logs centrally and securely. This will enable you to evaluate, monitor, alert, and audit access and actions performed on your cloud resources and objects.
- **Forensics** involves the analysis of log data and evidentially-captured images of potentially compromised resources, to determine whether a compromise occurred (and if so, how). Outcomes of root cause analysis resulting from forensic investigations are typically used to produce and motivate the application of preventative measures.

- **Service Onboarding** provides the ability to review and approve AWS services for use based on consideration of internal, compliance, and regulatory requirements. This capability includes risk assessment, documentation, implementation patterns, and the change communication aspects of service consumption.
- **Data De-Identification** enables you to discover and protect sensitive data as it is stored and processed (for example, national ID numbers, trade data, healthcare information).
- **Governance** enables you to define and enforce business and regulatory policies for your cloud environment. Policies can include rules for your environment or risk definitions. A portion of your governance policies is embedded in other capabilities across your environment to ensure that you meet your requirements.
- **Audit & Assessment** provides the ability to gather and organize documentary evidence to enable internal or independent assessment of your cloud environment. This capability allows you to validate assertions that all changes were performed in accordance with policy.
- **Change Management** enables you to deploy planned alterations to all configurable items that are in an environment within the defined scope, such as production and test. An approved change is an action which alters resource configuration that is implemented with a minimized and accepted risk to existing IT infrastructure.
- **Records Management** enables you to store, retain, and secure your data according to your internal policies and regulatory requirements. Some examples may include financial records, transactional data, audit logs, business records, and personally identifiable information (PII).

Operations

Enable your developers and operations teams to innovate faster, while ensuring the quality of application and infrastructure updates. The capabilities within this area enable you to build, deploy, and operate, workloads with ease in the cloud with developer experience and tools capabilities.



Operations Category

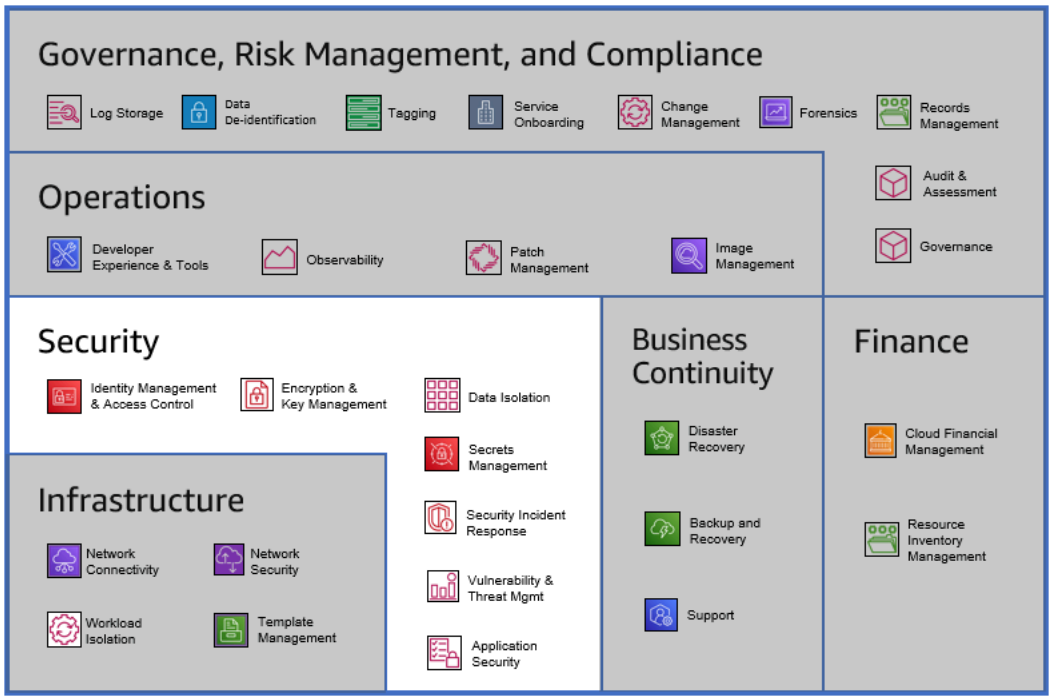
Operations capabilities include:

- **Observability** enables you to gather and analyze operational data about system and application activities. This includes the analysis of data to identify anomalies, indicators of compromise, performance, and configuration changes.
- **Image Management** enables you to manage compute images throughout their entire lifecycle. This can involve the creation, acquisition, distribution, and storage of the images.
- **Patch Management** is the ability to deploy sets of changes to update, fix, and/or enhance the operation and security properties of infrastructure and workloads. This includes addressing security vulnerabilities, bug fixes, and other related work. The scope of patch management includes operating systems, applications, and any relevant software systems.
- **Developer Experience and Tools** enables you to provide the tools and processes required for developers to build and deploy workloads. This capability includes managing code, building workflows, and promoting workloads into production environments.

Security

Create a secure, high-performing, and resilient foundation for your cloud environment.
The capabilities within this area enable you to design and implement security policies and

controls across different levels of the stack to protect your resources from external or internal vulnerabilities and threats. They ensure confidentiality, availability, integrity, and usability, while providing priorities and advice to assist with remediation.



Security Category

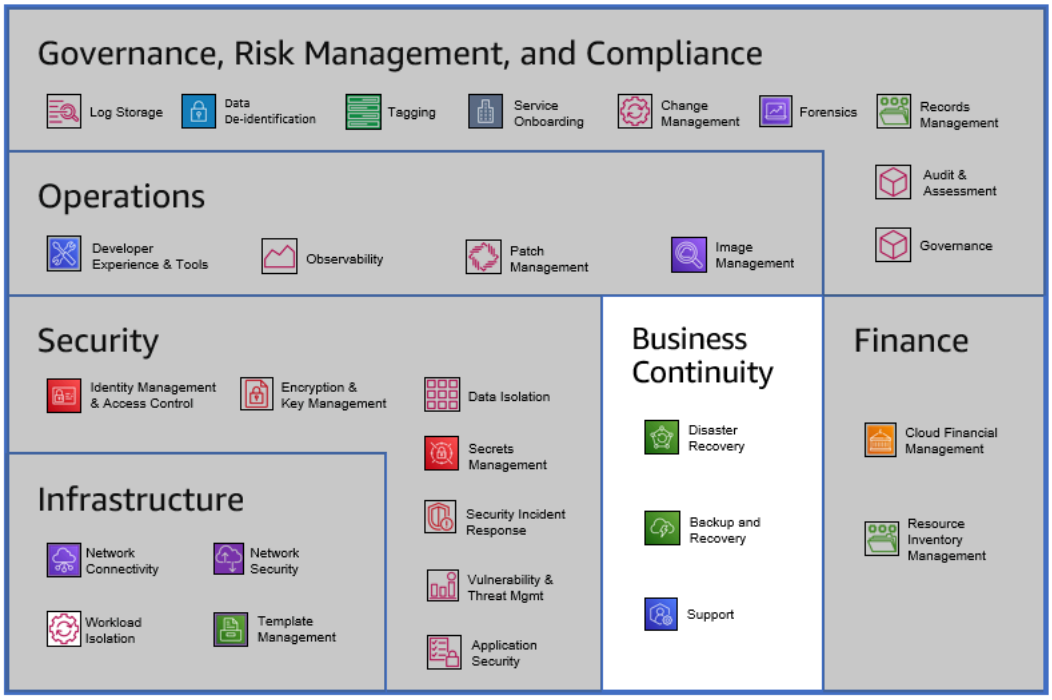
Security capabilities include:

- **Identity Management & Access Control** helps you build and monitor permissions in your environment. Use this capability to structure access to your resources within defined isolated groups following the principal of least privilege (PoLP). This capability will help your team develop a framework to manage your environment and provide access to your services.
- **Data Isolation** enables you to limit access to data at rest and in transit so that data is only accessible to appropriate and authorized entities. This capability also includes the ability to detect misuse and/or unauthorized access, leak, and theft of data.
- **Application Security** enables the protection of application software, and the detection of anomalous behavior in the context of the applications' interactions with customers. Threats to be addressed include unauthorized access, privilege escalation, and other application-level threats typically characterized in threat frameworks.
- **Encryption and Key Management** enables you to implement a key management strategy. This includes the ability to encrypt data at rest and in transit, provide least privileged access to keys, report on anomalies, and rotate keys based on requirements.

- **Secrets Management** enables you to manage secrets such as passwords, access keys, other API keys, X.509, or SSH private keys. This capability includes storage, access control, access logging, revocation, and rotation aspects for managing secrets.
- **Security Incident Response** enables you to effectively respond to a security incident based on decisions specified in policy. The response involves characterizing the nature of the incident and making changes (which may involve activities including restoration of operational status, identification and remediation of root cause, and gathering evidence pursuant to civil or criminal prosecution).
- **Vulnerability & Threat Management** is the ability to identify vulnerabilities that can affect the environment (availability, performance, or security). This capability enables you to assess the impact and scope (such as blast radius) of vulnerabilities and threats, and address/remediate them.

Business Continuity

Resilience is critical, it affects the quality of service your users experience. The capabilities within this area enable you to have a strategy in place to continue operations during a time of inefficiency or crisis, including Disaster Recovery, Backups and Support. Having this in place can help avoid downtime during outages or unprecedented situations.



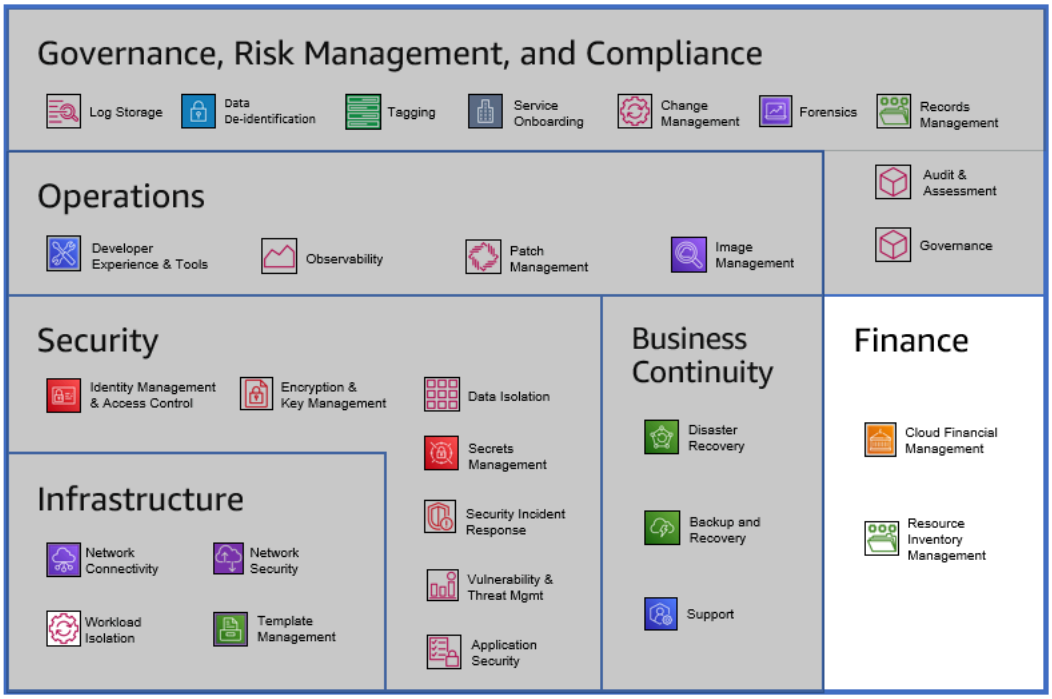
Business Continuity Category

Business Continuity capabilities include:

- **Backups** is the ability to make reliable copy of data in a reliable way for retrieval as needed to meet business and security goals, Recovery Point Objective (RPO), and Recovery Time Objective (RTO). Content to be backed up includes: orchestration framework data and configuration, application data, logs, and customer data.
- **Disaster Recovery** enables you to plan for and respond to a disaster scenario to ensure continuity of systems and to minimize the impact to the business. This includes the backup or replication of data and systems, failing over, testing, and executing against a DR plan.
- **Support** enables you to troubleshoot your cloud environment, submit tickets, integrate into existing ticketing systems, and escalate issues to an appropriate entity for a timely response depending on criticality and support level.

Finance

The capabilities within this area enable you to establish and enhance your existing finance processes to be cloud ready in order to establish and operate with cost transparency, control, planning, and optimization. Additionally, manage your records and resource inventory while meeting compliance and regulatory needs.



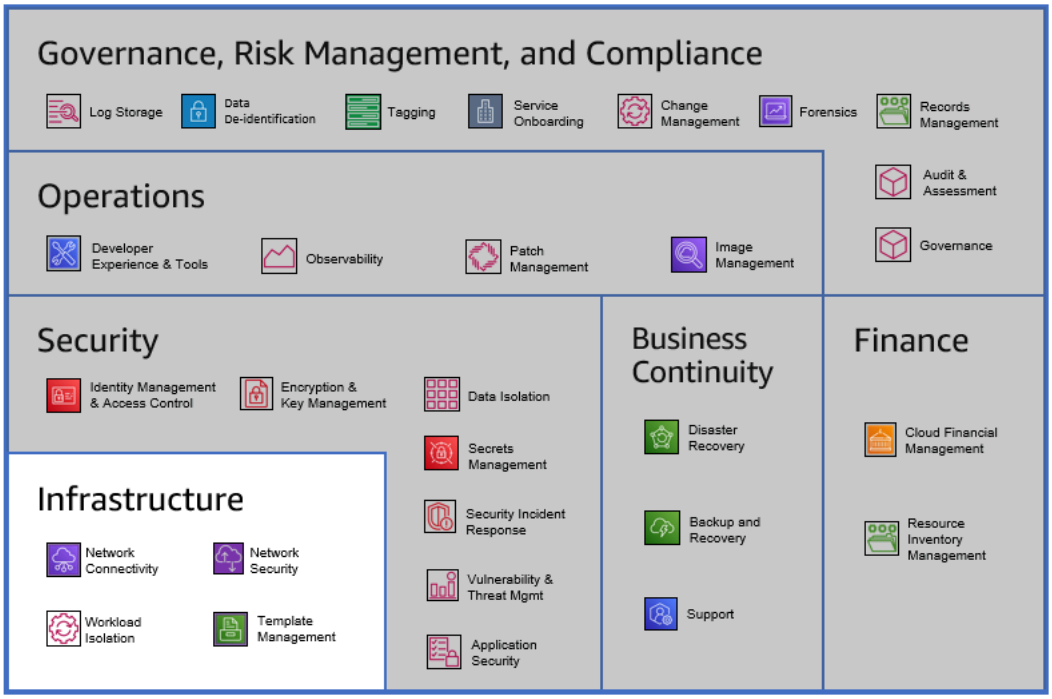
Finance Category

Finance capabilities include:

- **Cloud Financial Management** provides the ability to manage and optimize your expense for cloud services. This capability enables you to track, notify, and apply cost optimization techniques across your environment and resources. Expense information is centrally managed and consumed, and access to critical stakeholders can be provided for targeted visibility and decision making.
- **Resource Inventory Management** enables the collection, visibility, tracking, configuration validation, and service mapping of cloud resources.

Infrastructure

The capabilities within this area enable you to design, build, and manage a secure and highly available cloud infrastructure. Use practices such as Network Security to design and implement security policies and controls across different levels of the networking stack, and Workload Isolation to isolate environments that contain your newly migrated workloads. If you are migrating apps from on premises or building them natively in the cloud, the infrastructure that you build on should be both secure and reliable.



Infrastructure Category

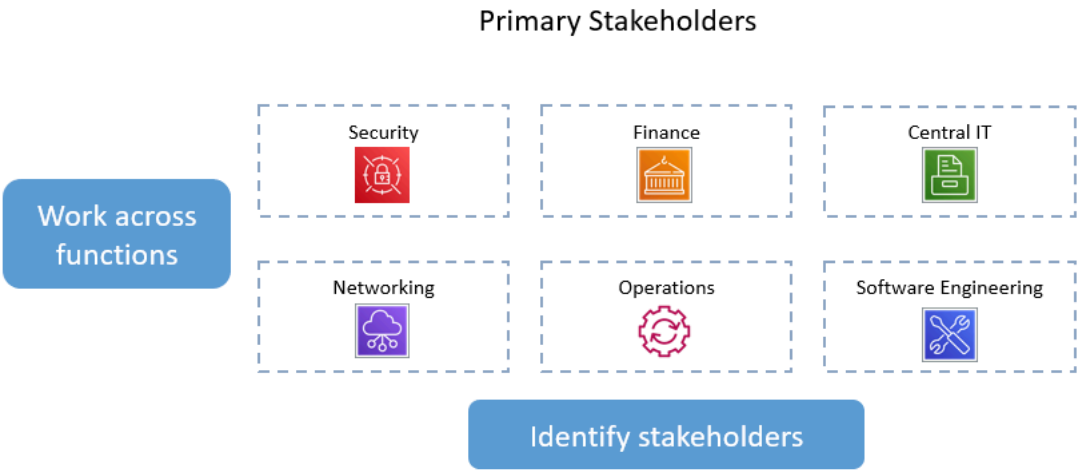
Infrastructure capabilities include:

- **Network Security** enables you to design and implement security policies and controls across different levels of the networking stack to protect your resources from external or internal threats to ensure confidentiality, availability, integrity, and usability. This capability includes the prevention, detection, and blocking of anomalous network traffic based on monitoring of ingress/egress and lateral data movement.
- **Network Connectivity** enables you to create, manage, and monitor secure, scalable, and highly available networks for your applications and workloads. This includes connectivity within the cloud, Hybrid connectivity, IP address management, network logging and monitoring, and DNS management.
- **Template Management** enables you to create and group reusable templates in a central repository to quickly deploy, manage, and update infrastructure, schemas, and resources across the environment. This capability includes the necessary processes to create, test, update, and validate the templates when required. These templates are pre-approved implementation patterns using approved cloud services, and are ready to be used by different teams based on requirements.
- **Workload Isolation** enables you to create and manage isolated environments for your workloads. This approach reduces the impact of vulnerabilities and threats, and eases the complexity of compliance by providing mechanisms to isolate access to resources.

Working with the capabilities

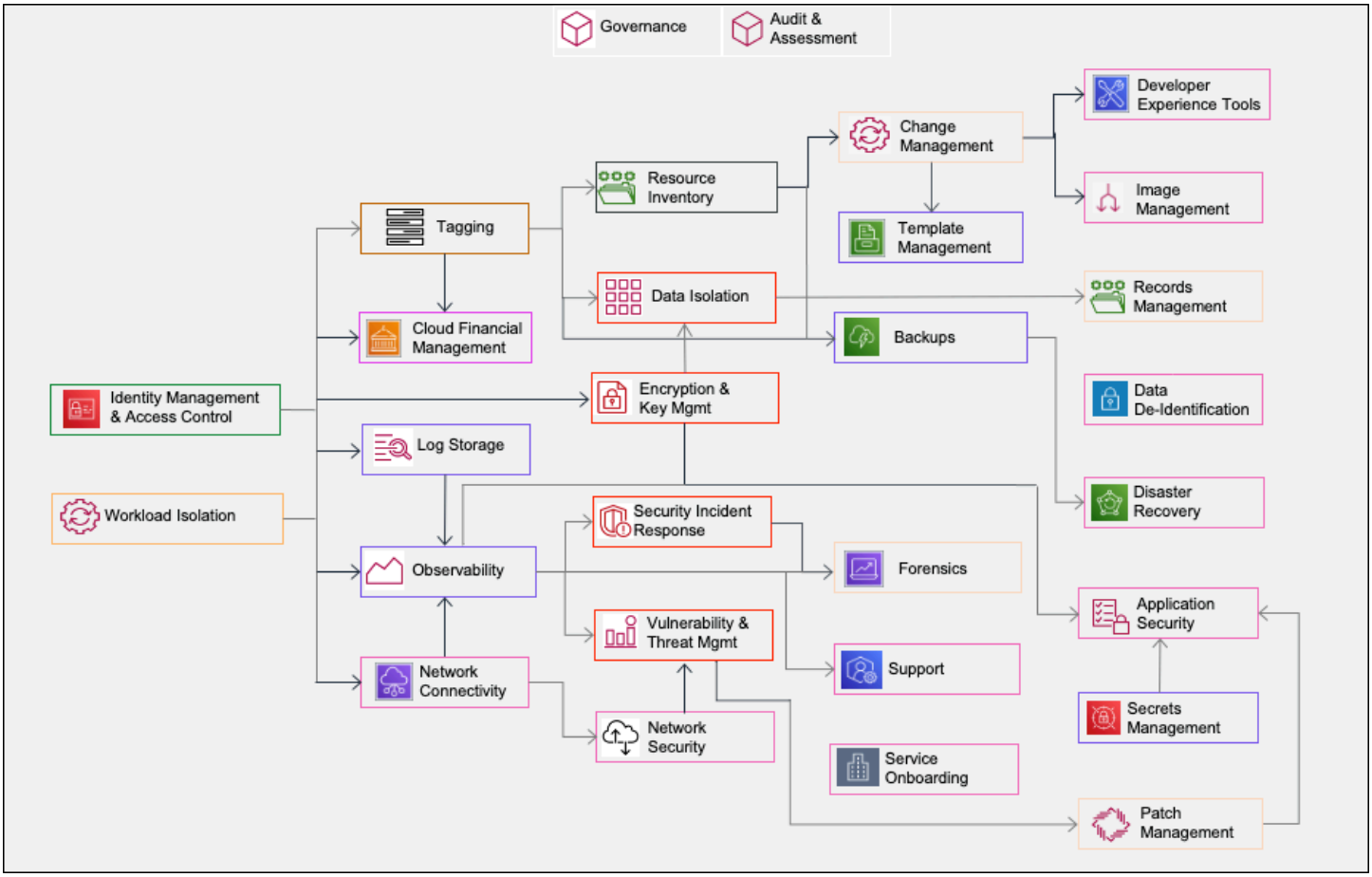
Each organization's cloud adoption journey is unique. To successfully build your cloud environment, you need to understand your organization's current state, the target state, and the transition required to achieve the target state. As you work on your plan to establish your environment, these capabilities can help you drive the conversation and decisions across relevant stakeholders (identified by the functional areas for each capability).

There are functional areas within each capability to help identify owners and stakeholders. Each capability has one primary functional area, which indicates the owner accountable for the capability. However, most capabilities are also relevant to other functional areas, which indicate the stakeholders responsible for providing input, and help make decisions for that capability.



Primary stakeholders across six categories diagram

The following graph shows a path that you can follow when planning your environment. It's based on dependencies between capabilities, and can be used to create a project plan for the implementation of capabilities in your environment. In addition to the dependencies shown (via the arrows), some capabilities apply to the overall environment (for example, Governance and Audit and Assessment).



Capability dependency guided path

The following foundational capabilities based on AWS best practices and guidance, can help you get started with building your environment.

Topics

- Identity Management and Access Control capability
- Log Storage capability
- Tagging capability
- Governance capability
- Workload Isolation capability
- Network Connectivity capability
- Change Management capability
- Cloud Financial Management capability

Identity Management and Access Control capability

The Identity Management and Access Control (IMAC) capability will help you build and monitor IAM permissions in your environment. This capabilities will enable you to structure your organization, organize your resources within defined isolated groups following the principal of least privilege (PoLP). The following guidance will help your team develop a framework to manage your environment and provide access to your services.

Category: Security

Stakeholders:

- Security (Primary)
- Operations
- Central IT
- Software Engineering

Personas:

- **Cloud Team** - the team(s) who make AWS available to customers.
- **Identity Management Team** – the members of the cloud subject matter expert (SME) team responsible for Identity Management and Access control in the cloud.
- **Information Security Team** - the team responsible for security in the cloud.
- **Consumer** - everyone who needs to access the cloud platform.

Supporting capabilities: Governance Capability

Scenarios:

- **CF2 - S1: Identity management**
- **CF2 - S4: Identity Operations**
- **CF2 - S7: Permissions management**

Topics

- Overview
- Structure the environment
- Define functions and responsibilities to manage your environment
- Federated access
- Delegate the administration of different services
- Establish preventive controls across your environment

Overview

When you are building your environment, access to your platform, your resources, and your applications needs to be established. First to build the environment, and second, to operate the environment through the established capabilities and services you build on it.

As you structure your environment, you want to delegate administrative tasks to different teams, and separate responsibilities across different functions. For example, implementing security tooling, managing the network, or creating central repositories. Different teams may be responsible, and granting access to administer and consume this resources and services you are building within the environment.

Access to your environment should be secured to all users, regardless of the function they will be responsible for. Enabling a form of multi-factor authentication (MFA) for every user is a requirement in order to meet a minimum-security standard.

Structure your identities to access your environment

Once your roles have been defined and you have decided what services you will start using, you need to structure your environment in a way that allows you to assign and separate the responsibilities previously described. We recommend that you start small where you can, and separate the security functions, workload environments (separating production from the rest of the environments), and sandbox environments. You can achieve this by using a mechanism to create isolated group of resources from each other.

You can group workloads with a common business purpose in a distinct isolated group of resources. This enables you to align the ownership and decision making with the isolated group of resources and avoid dependencies and conflicts with how workloads in other isolated group of resources are secured and managed.

Workloads often have distinct security profiles that require separate control policies and mechanisms to support them, you can **apply distinct security controls by environment.**

When you limit sensitive data stores to an account that is built to manage it, you can more easily **constrain the number of people and processes that can access and manage the data store**. This approach simplifies the process of achieving least privilege access.

In the early stages of a workload's lifecycle, you can help **promote innovation** by providing your builders with separate isolated group of resources in support of experimentation, development, and early testing.

Organizations often have **multiple IT operating models** or ways in which they divide responsibilities among parts of the organization to deliver their application workloads and platform capabilities.

Additionally, creating isolated group will help you organized your resources **based on their function**, and **share them across these the** different isolated groups when needed. Restrictions can also be applied across isolated group of resources that perform a similar action **applying common policies.**

Define functions and responsibilities to manage your environment

Federated access grants you the ability to efficiently manage the access to the environment, and should be established and operated centrally. The benefits of managing your identities and controlling access to your environment centrally, allows you to quickly create, update, and delete the permissions and policies you need to meet your business requirements. From granting or revoking permissions to specific users or roles, or by establishing preventative controls on your overall environment, your security teams can manage access to the environment from one place.

The responsibilities to perform certain actions of the environment need to be separated. Granting permissions to perform only the necessary actions to specific roles, and users, depending on the purpose of the service being established, achieving a least privileged access model. You also need to ensure that you group different users by their job family to access different tools with a different set of permissions, and that regardless of the user, there may be some preventative controls needed to set centrally to prevent access or modification to certain resources and areas of your environment.

We recommend that you establish Multi-Factor Authentication (MFA) for every role that has access to your environment, a minimum requirement is to establish MFA for administrator roles. This adds

an extra layer of protection on top of your sign-in credentials. Additionally, it is very important that every ***root user user*** has MFA enabled. For access to your organizational account, the root user user MFA needs to be enabled, and the access key and password should be stored separately.

When establishing your environment, you need to set the following functions in your environment, these functions will enable you to manage your overall infrastructure:

- The **Environment administrator role** manages the overall environment, creates isolated group of resources, sets guardrails, and delegates the administration of different services to the appropriate isolated group of resources and teams.
- The **Network administrator** manages the network. This function will have access to create and configure network topologies, DNS, VPNs, and build network security across your environment. Network administrators are responsible for securing the network and distributing resources workloads.
- The **Directory Services administrator** manages access to the environment. This function creates, updates, deletes, assigns, and removes access from different users to the environment.
- The **Security administrator** manages security services and tools across your environment, ensuring all your services and workloads are running on a secured infrastructure. This function has access to the environment to remediate any possible security threat.
- The **Billing administrator** manages the spend of your environment and creates budgets and alerts based on the forecasting of your expenses. This function is also responsible to pay the invoice for your environment.
- The **Read Only Security** function is used to monitor the environment. Security administrators can use this function to oversee the environment in real time and interact with the different security tools, without having full access to the environment
- The **Security Audit** is an exclusively read only function intended to grant access to external or internal auditors that need to examine the environment.
- The **Log Storage administrator** manages the log storage in your environment. This function creates or updates the resources needed to security and immutably store your logs, and manages the environment when changes need to be applied.
- The **Shared Services administrator** manages all the shared services across the environment. For example, this function is used to set up a central DNS or a central Template Management function.
- The **Support** function will have access to read only permissions for the infrastructure and not the data within each of the isolated group of resources in the environment. This will allow the cloud

team to help troubleshooting the environment, and the workload infrastructure when deployed, helping the team solve any issues with the environment or internal workflows. When necessary, it can be used to escalate to find resolution to your cloud service provider.

These functions with the appropriate permissions and boundaries should be assigned to the user groups which job family needs to perform tasks related to the roles above. This will allow them to access, monitor, operate, update, and secure the environment as needed to meet your business requirements. These functions represent responsibilities within your cloud environment, and multiple functions can be assumed by the same team or person.

To achieve this, you will need to build isolation boundaries to limit the access from each group to the services and tools needed to build, deploy, and operate. In some cases, these groups will need to work together to establish a connection to consume services between the boundaries isolating the resources, to create a cohesive environment that is secure and scalable, and provide access to these services to workloads or other foundational services within the environment.

Federated access

Federation is a common approach to building access control systems which manage users centrally within a central Identity Providers (IdP) and govern their access to multiple applications and services acting as Service Providers (SP).

An identity provider enables you to manage your user identities outside of the application, service, or solution; and give these external user identities permissions to use AWS resources in your account. This is useful if your organization already has its own identity system, such as a corporate user directory or managed identity service. The process of federation allows you to centrally manage user credentials which grant access to applications. The centralization of user credentials in one data store allows you to effectively manage the lifecycle of the user and impose security requirements such as password policies, MFA requirements, and service principals. If a user leaves the company, they can simply delete the user's corporate identity, which then also revokes access to all your federated environments.

When you use an identity provider, you don't have to create custom sign-in code or manage your own user identities. The IdP provides that for you. Your external users sign in through a well-known IdP, such as a log in with Amazon, Facebook, or Google. You can give those external identities permissions to use AWS resources in your account. Identity providers help keep your AWS account secure because you don't have to distribute or embed long-term security credentials in your application.

Delegate the administration of different services

As your environment grows, the complexity of managing your environment will increase with the amount of isolated group of resources you have. In order to effectively manage your environment, you will need to have a level of automation that matches the complexity of your environment.

Each team in your organization will assume different roles to manage different aspects of the environment. Delegating the administration of the services that the team will be managing in your environment will allow them to establish boundaries for data and security separately, they can build or deploy the necessary automation they need to operate and oversee the environment as they require, without affecting other teams or environments. For example, delegate the management of security services to your security team, delegate the infrastructure deployments, and template management to your Operations and Central IT teams, and delegate the management of your identity to your Identity or Security team.

Establish preventive controls across your environment

Only granting permissions does not guarantee that our environment is fully secured. To ensure that only the services that are intended to be used by the assigned roles, you need to limit what actions can be performed in your overall environment. For example, to limit the modification of the logs that are being stored in your log storage.

To prevent anyone from deleting or modifying these logs by mistake, you need to enable preventive controls to restrict the deletion on the logs in your log storage. On AWS this can be accomplished by applying service control policies to your accounts. These policies allow you to limit certain actions within a specific account, but you can also use them to prevent access to services completely, or limit the actions in your environment in specific regions that are not approved for use in your environment.

Log Storage capability

The Log Storage capability enables you to collect and store your environment logs centrally and securely. This will enable you to evaluate, monitor, alert, and audit access and actions performed on your cloud resources and objects.

Stakeholders:

- Security (Primary)
- Operations

- Central IT

Personas:

- **Cloud Team** - the team(s) who make AWS available to your customers.
- **Security Team** - the members of the cloud team responsible for security in the cloud.

Supporting capabilities: Identity Management and Access Control capability

Scenarios:

- **CF1 - S1: Central reliable log storage storage**
- **CF1 - S3: Log protection and integrity**
- **CF1 - S4: Log lifecycle management**
- **CF1 - S6: Log access management**

Topics

- Overview
- Benefits of centralized logs
- Log storage

Overview

The Log Storage capability primary mapping is to Security. The **Security team** should be responsible for implementing this capability according to your governance requirements.

Having a separated ***log storage*** allows you to establish a secure location where the logs become the source of truth for the actions and events happening in your environment relevant to security and operations. For example access to different accounts, or infrastructure updates.

Log storage must be tamper resistant and encrypted, and only accessed by controlled, automated, and monitored mechanisms, based on least privilege access by role. Controls need to be

implemented around the log storage to protect the integrity and availability of the logs and their management process.

Benefits of centralized logs

As your environment grows and scales with your business needs, creating a single location to aggregate all the logs across your environment helps simplify the analysis and monitoring of the logs. Additionally, it makes it easier to access the environment logs, and controlling who is able to consume the logs. This allows you to create different dashboards and tools for your logging capabilities.

When your environment scales and distributes across multiple resources and isolated environments, there are some benefits that centralizing your logs in one place may bring to your environment.

Create a single location for your logs

Logs should be aggregated into a central location for long-term storage and centralized analysis. This enables you to monitor your environment centrally and simplifies your operations. It also creates a single source of truth across your resources, security, and operations logs. Additionally, it reduces the chance of logs being lost and ensures your environment is appropriately tracked continuously.

Secure your logs

When the logs in your environment are stored in a central location it is easier for you to establish comprehensive controls to protect your environment. Human access to your logs should be limited. Instead, read actions should be performed by different automated mechanism standardize access controls. We recommend you to have a monitoring mechanism in place that triggers an alarm when the log storage is access with write or admin permissions.

Protect your logs with centralized controls

All of your logs should be stored in the same isolated environment protected by centralized controls. Controls should be configured to protect your log storage environment using both preventative and detective controls.

- **Preventive controls** enable you to prevent actions in the environment. Preventative controls can be used to restrict access and actions to log storage based on role, action type, service, or region.

- **Detective controls** are implemented to actively monitor the environment. This allows you to create alerts based on unwanted or unexpected actions taken within the environment. Optionally, remediation actions can be invoked automatically mitigate the risks within the log storage environment.

Log strategy

Each type of log that is being collected may require a different log storage strategy. The strategy will vary depending on the type of log, frequency, retention, size, quantity, compliance, and access that may be required. Some examples of common log types are: network logs, access logs, financial logs, DNS logs, inventory records, and change management records. A common lifecycle pattern for logs is keeping them for a period of time in: standard storage, cold storage option, archival storage, and then deleted.

Audit logs

We recommend that you protect your organization with a wide array of preventative controls to help you inhibit non-compliant changes. However, given the degree of self-service and agility often required by modern business, you need to ensure full transparency of changes made to at least production aspects of your environment, workloads, and data so that detective and corrective controls can be employed.

A secure, centralized repository of logs should represent the single source of truth and be tamper resistant because centralizing your audit logs provides you a clear view of what has occurred in your environment and when it happened. For example, this would be help you facilitate access to a trail during forensic investigations.

Auditors use of audit logs

If you work in a regulated industry, you will be engaging the services of an external auditing company in order to periodically assert your compliance with relevant standards. Your auditor will most likely have their own accounts as part of their own organization. They will need to analyze your log data as part of their audit process to determine whether you have remained in compliance since their last audit. It's a benefit to both you and the auditor to grant an account they nominate in their organization read-only access to your log archive bucket(s). This will enable your auditor to proactively access and analyze your logs in their environment before they need to engage in other audit activities, such as reviewing documentation and interviewing operations staff.

As part of the audit process, your internal security team might need to have a security assurance function. This would involve conducting internal dry-runs of external audits to minimize the risk of the external audit not proceeding smoothly. This process can be conducted by your security team, though they may wish to separate security assurance-specific activities into their own account for isolation from day-to-day security operations. If you have a security assurance team separate from your security team, their function should be separated into its own account in order to enforce separation of duty.

Configuration logs

Configuration logs contain detailed information about changes in your infrastructure or applications. Configuration logs also provide a current and historical view of infrastructure or application configurations. The length of time to keep configuration logs in each lifecycle phase will heavily depend on requirements, business policies, and applicable regulations.

Networking logs

Networking logs give you an overview of what is happening on your network. They can help you monitor traffic in your environment and diagnose network related issues. Due to the amount and frequency that networking logs are generated, it is common to keep them in accessible storage for a much shorter time compared to other logs. A best practice is to define the lifecycle strategy to keep your networking logs based on technical requirements, cost considerations, and the criticality of the infrastructure.

Tagging capability

Tagging is the act of assigning metadata to the different resources in your AWS environment for a variety of purposes, such as Attribute Based Access Control (ABAC), Cloud Financial Management, and automation (such as patching for select *tagged* instances). Tagging can also be used to create new resource constructs for visibility or control (such as grouping together resources that make up a micro-service, application, or workload). Tagging is fundamental to providing enterprise-level visibility and control.

Stakeholders:

- Central IT (Primary)
- Finance
- Security

- Software Engineering

Personas:

- **Cloud Team** - the team(s) who make cloud available to customers.
- **Security Team** - the members of the cloud team responsible for security in AWS.
- **Finance Team** - the members of the finance team responsible for reporting, allocating, and forecasting cloud costs.
- **Customer** - entity within the company that consumes the logs stored within the log storage.

Supporting capabilities: Identity Management and Access Control capability

Scenarios:

- **CF23 - S1: Tag definition and assignment**
- **CF23 - S2: Tag compliance**
- **CF23 - S3: Tag usage**

Topics

- Overview
- Choosing tags for your environment
- Tagging standards

Overview

Tagging enables you to assign metadata to the different resources in your environment. As metadata, tags allow you to assign additional labels to these resources for you to identify them according you your business needs. We recommend you define a tagging strategy for your environment, this will allow you to confidently and efficiently identify resources across your environment and teams.

It is important to define a strategy to tag your resources as soon as possible when establishing your Cloud Foundation on AWS, this will enable you to find resources and environments quickly,

as your overall environment expands and matures. When defining your tagging strategy, you need to determine the right tags that will help you gather all of the information you will need in your environment for the following scenarios:

Tags for workload and ownership

You can use tags to help you organize and display the resources that are owned by the same team or developer, as well as the resources that belong to the same workload across your environment. These tags can also help you identify what resources within a workload belong to a specific software development lifecycle (SDLC).

Tags for cloud financial management

Being able to control how much you are spending on the cloud and what resources are incurring the costs in your environment can help you reduce your costs in the long term. Being able to create reports and view the resources associated with a specific tag will also enable you to create budgets and forecast your spend based on specific tags.

Tags for regulatory scope definition and security risk management

When your resources are identifiable through tags, you can filter resources during your automated infrastructure activities. For example, when deploying, updating, or deleting resources within your infrastructure. Additionally, you can use tags to stop or start an entire fleet of resources according to your business needs.

Tags for operations and automation

When your resources are identifiable through tags, you can filter resources during your automated infrastructure activities. For example, when deploying, updating, or deleting resources within your infrastructure. Additionally, you can use tags to stop or start an entire fleet of resources according to your business needs.

Tags for operational support and disaster recovery

You can use tags to identify the kind of support a group of resources may need, and as part of your incident management process. Tags can be assigned to resources when they are isolated, or when they are on standby before deleting them or archiving them. This can help your support teams to identify the resources within a workload that need to be worked on. Tags can also be used to identify the frequency your resources need to be backed up, and where the backup copies need to go or where to restore the backups.

Tags for Attribute-based access control

In addition to role-based access control (RBAC), tagging your resources enables you to define and enhance the security of your resources in the environment. You can limit access to certain resources for roles in different environments, and you can also use tags to grant a temporary elevated access to certain resources. For more information, refer to the What is ABAC for AWS? documentation.

Authorization-based access control (ABAC) is not supported for all services. For information on what services support tags refer to, the service table. In the table, locate the service and check the **Authorization based on tags** column. You can also select the service name for additional documentation on authorization and access control for the service.

Choosing tags for your environment

Across the different kind of tags, you have to define for your environment, you need to decide what tags will be the **mandatory tags** or **discretionary tags**. Additionally, you need to define what resources need to tagged, and define detection and enforcement mechanisms to ensure all the required resources have the mandatory tags. When building an environment with multiple accounts, every account in the environment should have the mandatory tags that allow you to identify what the purpose of the account is for and who is responsible for the resources in that account.

> **Note**
>
> When building your tag strategy personally identifiable information (PII) should not be used to label your resources, as tags are not encrypted and are visible across your environment. Codify these values, so you can identify owners internally.

Mandatory tags are the set of tags that every resource should have, regardless of purpose. These tags will enable you to identify the necessary metadata to identify the resource.

The list of recommended **mandatory tags** includes:

- **Owner**- This tag indicates who is the owner and main user of the resource, this can be a team or an individual.

> **Note**
>
> The Owner is not always the user who created the resource.

- **Business Unit** - This tag identifies the business unit the resource belongs to.
- **SDLC Stage** - This tag indicates if the resources are being used for Production or for non-Production. (For example, development, test, or sandbox.)
- **Cost Center** - This tag specifies the budget or account that will be used to pay for the spend associated with the tag.
- **Financial Owner** - This tag identifies who is responsible for the costs associated with the resource tagged with a specific tag.

Discretionary tags are the set of tags that must be defined as part of your tagging strategy, so they are available to be assigned to resources that need them (for example, temporary elevation of permissions, or data sensitivity).

The list of recommended **discretionary tags** includes:

- **Workload ID/Name** - This tag indicates if the resource belongs to a specific workload. The value can be the workload ID or name.
- **Compliance Requirement** - This tag identifies the resources that are subject to a specific compliance framework (for example, PCI-DSS or HIPAA).
- **Environment version** - This tag indicates the version of the environment, in case the same workload has more than one environment associated.
- **Workload tier** - This tag indicates the type of workload the resources belong to. Some workload types examples are confidential, internal, or critical.

- **Backup** - This tag indicates if the resource needs to be backed up based on the type of workload and the data that it manages.

- **SLA level** - This tag indicates SLA requirements.

- **Lifespan** - This tag indicates the lifetime of the resources of the workload. If exceeded, these resources should be reviewed, replaced, or isolated.

Tagging standards

As you define your tagging strategy, a naming convention needs to be established for the different tags across your environment ensuring a standard, and making it easier for the tagged resources to be identified. Tags enable you to identify resources, and having no more than 50 tags per resource will allow you to keep your tag strategy manageable in your environment. The following are examples for tag key names and values:

```
example-key:owner = SecOps
```

```
example-key:cost-center = 5432
```

```
example-key:financial-owner = Security
```

Resources that need to be tagged

There are resources that always need to be tagged in your environment, because it is critical to have the identifiable information about the resources at all times. The resources in this category are meant to be persistent, and in some occasions, they act as resource containers for other resources. These resources include, but are not limited to, accounts, critical workloads, and shared infrastructure. For these resources, you should aim to tag a hundred percent of the resources which will allow you to identify the spend, access, ownership, and permissions for the tagged resources.

However, as operational complexity increases, and the level of automation to manage tags becomes more demanding, you may choose to not tag certain types of resources that are ephemeral. These resources should run within a resource container that is properly tagged to allow you to identify and trace what happened within that environment, but enforcing the tags on these types of resources may not be necessary if they do not belong to critical workloads or applications.

Enforcing tagging

Because of the importance of tagging and the level of complexity, it's recommended to automate the tagging process when possible. This will reduce the human error that can be introduced when tagging critical resources, and will minimize the number of resources that are not identifiable due to the lack of tags. When possible, creating tag policies in your organization can help you ensure that the tags assigned to resources have the correct value assigned.

Additionally, automation needs to be established in the environment to discover resources with missing tags or resources that are not compliant with the established tagging strategy. Once the resources have been identified, a report including these resources on the environment needs to be sent to the relevant stakeholders, to evaluate and make a decision to remediate the situation, if needed.

Based on the results of this report, if a situation where persistent resources that are identified as non-compliant or have missing tags is given, it should be remediated immediately, by assigning a default pre-defined tag value defined as part of your tagging strategy, or if pre-defined tag doesn't exist deleting the non-compliant resources.

As part of your tagging strategy, it is important to implement preventative controls that ensure that disallow resource to be created without the appropriated tags on critical resources. For more information, refer to Establish preventive controls across your environment.

Build a tagging dictionary

As part of your tagging dictionary, you should define certain tags that can be used to access specific environments resources based on the tags attached to a role at a certain time. These tags can be used for a temporary escalation of privileges or for deploying changes through infrastructure as code that other identities may not have access otherwise.

We recommend that you define and build a tagging dictionary where all these values are available for developers, cloud architects, and environment operators. In order to add, update, or remove values for each of the tags included in the tagging dictionary, you need to establishes processes where all the relevant stakeholders can provide their inputs, when a tag becomes standard. This will ensure that all relevant stakeholders involved in the definition of the tags in your environment are aware of any changes they need to provision and deploy across their resources.

This tagging dictionary needs to be made available to builders and stakeholders, so tags can be applied consistently across the environment, and everyone is aware of requirements or errors that

can be caused due to an incorrect used of tags in the environment. Including missing tags and wrong or misspelled tag keys in your resources.

Defining tags for Attribute-based access control

As part of your tagging dictionary, you should define certain tags that can be used to access specific environments resources based on the tags attached to a role at a certain time. These tags can be used for a temporary escalation of privileges or for deploying changes through infrastructure as code that other identities may not have access otherwise.

Governance capability

The Governance capability enables you to define and enforce business and regulatory policies for your cloud environment. Policies can include rules for your environment or risk definitions. A portion of your governance policies is embedded in other capabilities across your environment to ensure that you meet your requirements.

Stakeholders:

- Governance, Risk Management, and Compliance (Primary)
- Finance
- Security
- Central IT

Personas:

- **Cloud Team** - the team(s) who make cloud available to customers.
- **Information Security Team** - the members of the cloud team responsible for security in AWS.
- **Finance Team** - the team responsible for reporting, allocating, and forecasting cloud costs.
- **Compliance Team** - the team responsible for compliance.
- **Procurement Team** - the team responsible for procurement of services from the cloud service provider.

Scenarios:

- **CF26 - S1: Cloud service provider relationship**

- **CF26 - S2: Operational standards**
- **CF26 - S3: Organizational cloud awareness**
- **CF26 - S4: Policy communication**
- **CF26 - S5: Governance at scale**
- **CF26 - S6: Compliance management**

Topics

- Overview
- Establish the relationship with your cloud services provider
- Define how cloud services are adopted
- Get started with your cloud services provider
- Build cloud capability across your organization
- Respond to growth or change
- Industry-specific governance
- Security Assurance on AWS

Overview

Governance of your environment is important to address questions on why and how cloud services are consumed. You cloud environment will need to align with your organization's strategy on cloud service provider usage. All organizations, regardless of size and industry will need to establish a capability to successfully consume cloud services, define policies and standards, understand and mitigate risks and confirm necessary legal, commercial, and regulatory requirements.

Establish the relationship with your cloud services provider

When starting your cloud journey, you need to establish a commercial relationship with your cloud services provider. You will complete relevant customer agreements and set up preferences for

communication and how you will pay for cloud services consumed. For larger organizations, you will need to confirm which parts of your organization are responsible for these functions.

When selecting your cloud provider, ensure that you decide on your cloud strategy. A cloud first strategy will allow you to bring new workloads, projects, and experiments to your cloud environment. Freeing up the load from your on-premises resources, if you have any. When new workloads are designed for the cloud, this allows you to realize the cloud benefits faster.

When you select your cloud provider, you can conduct risk and compliance assessments. Each cloud provider has different tools you can leverage to obtain those reports. AWS Artifact is a self-service portal at no cost where you can get AWS compliance reports.

When establishing a relationship with a cloud provider, you can benefit from procurement agreements. Ensure that you review and accept the terms included in these agreements, and if needed, consult with your legal team.

Define how cloud services are adopted

Before you start building your cloud environment, you need to define policies on cloud consumption. Having your governance policies well-defined, will ensure that the foundational environment you build will support your workloads, and will enable you to define processes to follow to deploy, operate, and govern the different workloads across your environment.

As a part of defining how cloud services are consumed, you will need to confirm which risk and compliance frameworks apply and how your environment will meet those requirements on an ongoing basis.

Another key component of managing and governing your cloud environment will be the operating model that you put in place. You will need to define roles and responsibilities for how you will address the customer components of the Shared Responsibility Model. Many customers decide to set up a Cloud Center of Excellence (CCoE), Cloud Business Office (CBO), or a cloud team which is charged with developing the approach to implementing cloud technology at scale for your organization.

Get started with your cloud services provider

When you start using cloud services you will need to decide on the region in which you will be mainly operating your cloud services. An AWS Region is a physical location around the world where data centers. AWS calls each group of logical data centers an Availability Zone (AZ). Each region consists of multiple, isolated, and physically separate AZs within a geographic area. Customers

choose a region based on where it makes sense for their workloads and the customer's security, risk and compliance posture. In order to choose appropriately, you can consider:

- Proximity to your location (headquarters)
- Proximity to your customers
- Services available in your region
- Regulatory and data residency considerations
- Compliance frameworks that are relevant, such as GDPR and HIPAA
- Legal and/or tax requirements

Generally, customers select a single main region where they set up administrative services to govern and control all of their cloud resources across all regions where they have workloads. In some specific cases, you may use more than one region in order to best serve your customers or to provide for additional scalability, reliability or low latency for certain workloads, or to satisfy workload-specific requirements, but generally one region is sufficient.

Once you decide which one will be your main region, the next policy you need to establish is what regions you will be operating in, considering your customer base, your disaster recovery strategy, and (other) policies you may have already established in your current IT environment. The policy should include not only what regions you will use, but what you need/want to do with the regions in which you will not be operating any of your workloads, and allowing or restricting access to these regions. This will be part of your **Data residency and retention requirements**; basically, where does your data need to live to comply with your legal requirements, and how long do you need to keep this data stored for your customers. As you define this, you can build data lifecycle polices, to help you archive data at a specified frequency, and delete data that is older than the maximum required archive date.

Build cloud capability across your organization

As you prepare to offer cloud as a service for your organization, you should consider identifying an owner who will sponsor the cloud adoption and they can build a team with the appropriate skill sets to deploy, operate, and govern the environment. As part of your foundation journey, we will be providing an estimated level of effort and the skill sets needed for building and operating each of the capabilities. To maximize the gains/outputs from your cloud initiatives, the Cloud Adoption Framework Governance perspective includes details to help you identify what needs to be done in these area.

As your cloud environment grows, responsibilities within your cloud environment will grow, and you need to ensure that you identify the appropriate owners to support the different workloads you will be deploying. Designating appropriate stakeholders to be aware of what is being built in your environment, to unblock your cloud team and your developer teams when they need to establish certain capabilities or deploy their workloads to the platform. When the appropriate stakeholders are identified early on, you will be enabled to make the right decisions for your environment faster. You can use the primary functional areas for the Cloud Foundations capabilities to identify stakeholders in your organization.

Once the different stakeholders are identified, we recommend you align them to the Shared Responsibility model. This will enable you to define a cloud operating model for your environment, where all the different teams involved in creating or enhancing your environment are aware of what needs to be done to move forward. Processes and standards are easier to manage, and they are visible across your organization.

Finally, as your organization grows, different teams will benefit from a training and certification program. This will allow the teams and stakeholders within your organization to stay up to date with the newest technologies, methodologies, and recommendations when managing your environment and the workloads running on it.

When establishing standards for your cloud environment, you need to define a home Region, where your data will be kept, and if there are any applicable region restrictions that need to be considered. You will also need to assign different stakeholders to each of the capabilities that need to be established in your environment according to your policy. This will enable you to establish a standard approve/deny process for new projects and workloads for your cloud environment.

Each team can create isolated environment for their workloads, in order to enable them to innovate and experiment. Different policies can match to different use cases in your environment, such as:

- Sandbox usage
- Training time
- How/when to request a new isolated workload environment
- What does the isolated workload environment look like?

As you prepare to establish your environment, the cloud foundations capabilities will provide a guided path to establish an environment based on AWS Best Practices and Recommendations,

that will enable you to implement each of these capabilities in your environment adhering to your Governance policies and requirements.

As you get started with your cloud provider, certain standards will allow you to simplify the management of your cloud environment, such as setting up standards and roles that the teams will use to interact with the cloud provider, defining different namespaces and email addresses for each team to use when accessing the environment and what is the level of internal support within your organization and from the cloud provider.

Other standards that we walk you through within other capabilities will allow you to define and develop mechanisms such as:

- How to create, test, and create cloud policies
- How to define a strategy to source and distribute software and Infrastructure as Code
- Determine what type of risk you can assign to your workloads, from those that need minimal governance, to those that are high risk, and will need board or CCoE approval to be deployed, updated, or removed

Establishing capabilities and standardizing process across your organization following an operating model you define, enables your teams to start realizing the benefit and power of the cloud, and will allow them to innovate faster and focus on key business differentiators, freeing them for complex and repetitive administrative task to manage their environment(s).

Respond to growth or change

A Cloud Center of Excellence (CCoE) or a Cloud Team can help to express and manage the cloud strategy you are following based on your governance policies and will assist to coordinate across different teams to set up governance, and assist in architecting the cloud environment and new workloads that will be deployed on the cloud. A CCoE or a Central Cloud Team, will drive the established standards across your organization helping to drive cloud adoption within your organization. Additionally, the CCoE can perform the function of a training and certification enabler for the teams across your organization.

Thinking about your home region is not always something that is done once at the start of your cloud journey. Situations can arise such as a merger or acquisition of another company, or an expansion of your company into other geographic regions that may cause you to revisit the regions where you operate and run cloud workloads. To respond to these kinds of events, there are

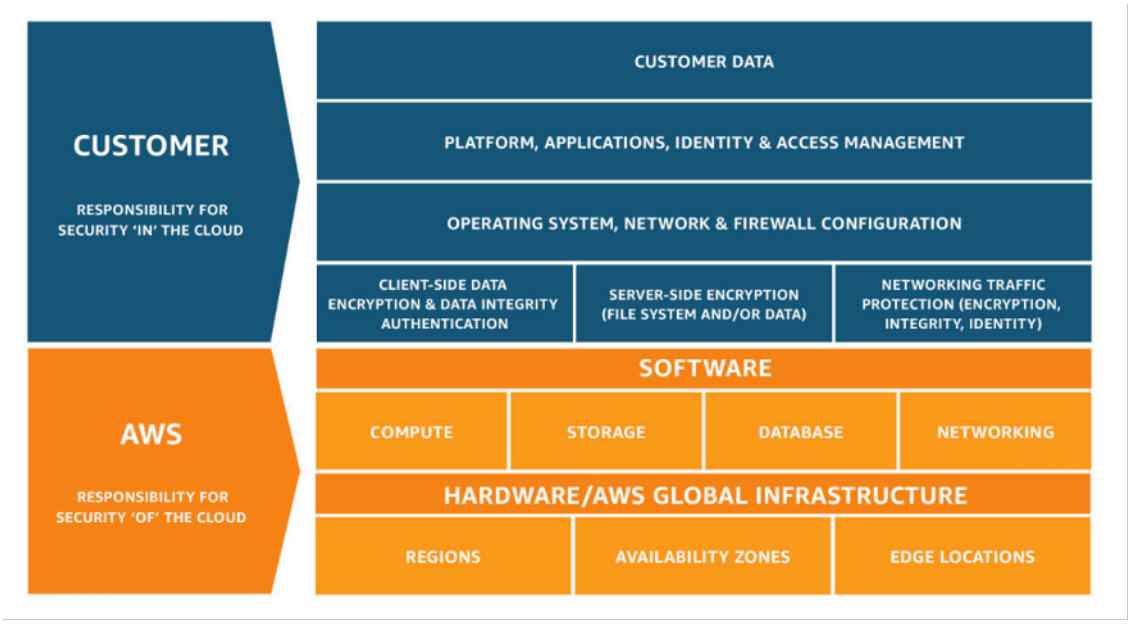
external vendors, partners, and products that can be used to help unlock your journey. However, processes to procure products through marketplaces or establishing relationships with preferred Partner(s) or Professional Services will allow you to quickly use industry standard and backfill the skill sets you need to deploy and operate your environment. A CCoE can help coordinate these relationships for your organization, and help make the right products available to the necessary teams.

Industry-specific governance

For customers operating in certain industries such as financial services, healthcare, or government; specific governance requirements may apply. As part of setting up governance for your foundational cloud environment, we recommend that you build in this capability from the start to address these specific industry governance requirements. We also recommend that you assign specific roles and responsibilities and work with your cloud services provider to leverage available guidance, solutions, or Partner support to assist you with meeting your industry-specific governance requirements with automation and optimization. For more information, refer to the AWS Compliance Center.

Security Assurance on AWS

Security and Compliance is a shared responsibility between AWS and the customer. This shared model can help relieve the customer's operational burden as AWS operates, manages and controls the components from the host operating system and virtualization layer down to the physical security of the facilities in which the service operates. The customer assumes responsibility and management of the guest operating system (including updates and security patches), other associated application software as well as the configuration of the AWS provided security group firewall. Customers should carefully consider the services they choose as their responsibilities vary depending on the services used, the integration of those services into their IT environment, and applicable laws and regulations. The nature of this shared responsibility also provides the flexibility and customer control that permits the deployment. As shown in the chart below, this differentiation of responsibility is commonly referred to as Security "of" the Cloud versus Security "in" the Cloud.



Shared responsibility model

Security of the Cloud

The following resources can be used to help you ensure security of your cloud environment:

AWS Global Infrastructure

The AWS Global Infrastructure is built around AWS Regions and Availability Zones. AWS Regions provide multiple physically separated and isolated Availability Zones, which are connected with low-latency, high-throughput, and highly redundant networking. With Availability Zones, you can design and operate applications and databases that automatically fail over between Availability Zones without interruption. Availability Zones are more highly available, fault tolerant, and scalable than traditional single or multiple data center infrastructures.

AWS Compliance Programs

The AWS Compliance Program is used by customers to understand the robust controls in place at AWS that maintain security and compliance in the cloud. IT standards that AWS comply with are broken out by Certifications and Attestations; Laws/Regulations; Privacy; and Alignments/Frameworks. You can use this information in the compliance programs as inputs and guides to build your own compliance program for how your organization can use AWS.

AWS Artifact

AWS Artifact provides a central resource for AWS security and compliance reports including Service Organization Control (SOC) reports, Payment Card Industry (PCI) reports, and certifications from accreditation bodies that validate the implementation and operating effectiveness of AWS security controls. You can use the reports available in AWS Artifact as inputs to questions that are a part of your internal supplier due diligence processes, as part of overall governance of the use of cloud services.

AWS Artifact Agreements enable you to use the AWS Management Console to review, accept, and manage agreements for your AWS account or AWS Organizations. An example of such an agreement is the Business Associate Addendum (BAA). A BAA typically is required for companies that are subject to the Health Insurance Portability and Accountability Act (HIPAA).

AWS services to help govern your AWS environment

The following resources can be used to help govern your AWS environment:

AWS Organizations

AWS Organizations allows you to centrally govern your AWS accounts. You can perform account management activities at scale by consolidating multiple AWS accounts into a single organization. You can leverage the multi-account management services available in AWS Organizations with many AWS services to perform tasks on all accounts that are members of your organization. AWS Organizations includes service control policies (SCPs) that you can use to provide centralized control over all accounts in your organization.

AWS Control Tower

AWS Control Tower is a managed service that orchestrates the set up and deployment of guardrails across the AWS accounts in AWS Organizations. If you are building a new AWS environment, starting out on your journey to AWS, or starting a new cloud initiative, AWS Control Tower can help you get started quickly with built-in governance and best practices.

AWS Solutions

AWS Solutions can help you implement the capability automatically where services are not available at the moment, please reach out to your account team for additional information on what types of solutions are available for your business needs.

- AWS Compliance Solutions Guide

- AWS Partner Solutions for Governance, Risk, and Compliance
- AWS Public Sector Partner Program

Workload Isolation capability

The Workload Isolation capability enables you to create and manage isolated environments for your workloads. This approach reduces the impact of vulnerabilities and threats, and eases the complexity of compliance by providing mechanisms to isolate access to resources.

Stakeholders:

- Central IT (Primary)
- Security
- Operations

Personas:

- **Cloud Team** - the team(s) who make cloud available to customers (such as App DevOps).
- **DevSecOps** - The team defining, building, analyzing, and mitigating access and authorization to application workloads and data assets.
- **Audit and Compliance / Governance, Risk Management, and Control** - The team(s) performing review and approval of control adherence or exception.

Supporting capabilities: Governance capability and Network Connectivity capability

Scenarios:

- **CF7 - S1: Designing isolated resource environments**
- **CF7 - S2: Isolated environment lifecycle management**
- **CF7 - S3: Baselining isolated environments**
- **CF7 - S4: Repeatable patterns for isolated environments**

Topics

- Overview
- Design isolated resource environments
- Provision processes for isolated environments
- Implement controls on isolated resource environments
- Provision baseline standards to isolated resource environments
- Provision pre-approved deployable architectures
- Decommission process for isolated resource environments

Overview

As you scale your cloud environment usage beyond a workload, it is important to develop repeatable processes for groups of isolated resources and workload provisioning within a set of guardrails. This maintains consistent controls and boundaries to support application development teams and other consumers. These controls will span cost, regulatory, and compliance domains with implementation often starting as runbook-based mechanisms and evolving into automated processes as business requirements drive the need to scale. By enabling automation, you can remove human error through use of CI/CD and Infrastructure as Code (IaC) principles.

Design isolated resource environments

Creating an identity and access management isolation boundary for workloads reduces the risk of a workload infrastructure update impacting a different workload, simplifies cost management, and allows application teams to operate within a bounded environment. Being able to implement preventative or detective controls on isolated resource environments will allow different controls for different workload types or Software Development Life Cycle (SDLC) environments (such as dev, test, prod).

Workloads often have distinct security profiles that require separate control policies and mechanisms to support them. For example, it's common to have different security and operational requirements for the non-production and production environments of a given workload. The resources and data that make up a workload are separated from other environments and workloads with defined isolation boundaries.

You can group workloads with a common business purpose in distinct environments. This enables you to align the ownership and decision making with those environments, avoiding dependencies and conflicts with how workloads in other environments are secured and managed.

Different business units or product teams might have different processes. Depending on your overall business model, you might choose to isolate distinct business units or subsidiaries in different environments. Isolation of business units can help them operate with greater decentralized control, but still provides the ability for you to implement overarching guardrails. This approach might also ease divestment of those units over time. For information about best practices for organizing your environment on AWS, refer to Organizing your AWS environment using multiple accounts.

Provision processes for isolated environments

At a smaller scale, request and review processes for creating isolated resource environments often start with manual based inputs to a cloud management team through a ticketing system or other request process. The cloud team will then analyze the request and ensure a set of information is included such as: owner, email address, cost center, project number, and SDLC environment. Additional customer specific meta data may also be collected based on requirements. If an application review board or enterprise architecture board are present within the customer environment they may also review the request. For certain types of environments, it may be decided that approvals are not required or an auto-approval mechanism may be put in place.

Once approved, the isolated environment is created with a baseline configuration and controls that apply to all environments. Configuration and controls may also be applied based on the metadata within the request. Non-production or production environments, for example, may have different requirements and controls applied. Baseline configurations might include the removal of default password configurations, authentication methods, integration with an existing identity provider, IP allocation, network segmentation and connectivity, logging, and security tooling. Additional customer required controls will be implemented at this point, often driven by industry specific regulatory requirements or cloud control best practice guidance through common frameworks such as Cloud Security Alliance - Cloud Controls Matrix or NIST Cybersecurity Framework (CSF) Reference Tool.

As your environment matures, request, review, and approval processes can be automated using integration with existing IT service management or human workflow tools. Deployment processes can leverage automation through GitOps deployment pipelines and infrastructure as code.

Implement controls on isolated resource environments

Implementing controls on isolated resource environments provide cloud consumers the autonomy to build workloads to meet their business objectives while keeping the organization compliant to standards and regulations. Controls can take the form of either preventative or detective.

Preventative controls prevent events from occurring based on a variety of criteria, including the event type, action, resource, caller identity, and more. For example, a preventative control might deny all users from deleting a specific resource.

Detective controls are designed to detect, log, and alert after an event has occurred. These controls will not block actions from occurring, instead they can be used to provide notifications, generate incidents, or initiate automation to address the violation. Detective controls can be helpful in validating that preventative controls are working, auditing an environment, building automated responses to address violations, or measuring the success of process changes.

The sooner a guardrail can be evaluated in the deployment process the better. Implementing preventative guardrails in the deployment process allows compliance risks to be caught earlier and before an attempt to deploy resources even occurs. Identifying risk earlier also saves time during deployments. Building controls into the deployment workflows allows for the inspection of code against standards which can enforce guardrail compliance as resources are being provisioned.

Provision baseline standards to isolated resource environments

In addition to provisioning guardrails to isolated resource environments, it is also necessary to provision other baseline requirements within the environment. This can include roles, network components and connectivity, and security applications or services.

In order to maintain consistency in configuration across different resource environments, the baseline configuration should be deployed in an automated fashion in existing or newly created isolated resource environments. This can include roles, network connectivity, or operational or security services. It is often helpful to be able to apply baseline configurations globally (to all isolated resource environments) or to logical groupings of the isolated resource environments. Global baselines will apply to all isolated resource environments and therefore should be considered carefully. Isolated resource environments should be grouped and organized in a way that allows baseline configurations to be applied to the logical grouping of environments. For example, it is common to apply stricter controls on production environments than in development environments. Therefore, production environments should be grouped in a way that allows for

different baseline configurations to be applied to production environments and development environments.

Once the different stakeholders are identified, we recommend you align them to the Shared Responsibility model. This will enable you to define a cloud operating model for your environment, where all the different teams involved in creating or enhancing your environment are aware of what needs to be done to move forward. Processes and standards are easier to manage, and they are visible across your organization.

Provision pre-approved deployable architectures

As you adopt the cloud you will see common workload patterns emerge, which may include standard 3-tier web applications, serverless frameworks, or many other architecture patterns. At the same time, certain skill sets may vary as teams learn to leverage the cloud. Common patterns may be implemented with any number of variations, along with potential for limited people knowledge has a risk to deploy non-compliant workloads. Pre-approving and building a pattern library will provide a common and repeatable mechanism to maintain controls and boundaries around workloads. Instead of application teams relying on run-book, process documentation, or lengthy review processes, cloud platform teams can publish ready to deploy patterns which include common resources along with included control guardrails. For example, a 3-tier web app will include a typical set of compute instances placed within the appropriate public and private network segments to prevent accidental exposure to internal systems.

Decommission process for isolated resource environments

There are various situations when your organization may need to decommission one or more isolated resource environments. For example, you may have a resource environment exclusively designed for and used by a single application that is being decommissioned, you may be using sandbox (disposable) environments, or you might have misconfigured a resource environment during your testing or development phases and determine that the best path is to delete the entire environment. For any situation that requires the decommissioning of isolated resource environments you will want to ensure a consistent termination workflow is in place to prevent unintended charges for resources no longer needed. Additionally, you will want to disable access to the resource environment during any interim waiting period while your resource environment is being terminated.

Similar to how you would approach provisioning isolated resource environments, you will want to have a request process in place for decommissioning isolated resource environments. You should

ensure that any dependencies with other isolated resource environments are no longer needed before decommissioning. Internal policies and compliance needs should guide how persistent data within the environment is either deleted, or safely transitioned into a separate environment. We recommend that you carefully consider what assets from the environment should be retained, and how they should be retained. For example, consider retaining assets that could add value to your business in the future or that could augment future projects by transitioning them out of the resource environment prior to termination of the environment.

The decommissioning process should be documented and include guidance on the required steps in the workflow. This might include:

- A method for determining if any assets or data within the environment should be retained.
- Applying restrictive controls on the isolated resource environment for a period of time to prevent the use of the resources but allow a recovery process, if necessary.
- A process or automation for disabling resources or deleting data.

Network Connectivity capability

The Network Connectivity capability enables you to create, manage, and monitor secure, scalable, and highly available networks for your applications and workloads. This includes connectivity within the cloud, Hybrid connectivity, IP address management, network logging and monitoring, and DNS management.

Stakeholders:

- Networking (Primary)
- Central IT
- Software Engineering

Personas:

- **Cloud Team** - the team(s) who make cloud available to customers.
- **Networking Team** - the members of the Cloud team responsible for security in AWS.
- **Developer experience** - Development teams that will be deploying workloads onto the network.

Supporting capabilities: Identity Management and Access Control capability

Scenarios:

- **CF11 - S1: Connectivity within the cloud**
- **CF11 - S2: IP address management**
  - Design IP address scheme
- **CF11 - S4: Hybrid connectivity**
- **CF11 - S5: Network logging and monitoring**
- **CF11 - S6: DNS management**
- **CF11 - S8: Network orchestration**

Topics

- Overview
- Connectivity within the cloud environment
- Network designing and planning
- Centralized or distributed network configuration and management
- Hybrid environment set up
- Establish logging and monitoring set up
- Construct DNS set up for applications

Overview

Connecting the underlying infrastructure is an essential piece of every workload running on your environment. This makes a redundant, highly available, and fault tolerant network architecture one of the top priorities for every organization. Every workload you run in the cloud or within an on-premises environment requires connectivity between different resources. For example, servers, firewalls, edge routers, gateways and more.

Having a strong networking foundation and planning ahead on the underlying network helps organizations to deploy, manage, and establish controls on the network.

Networking plays a critical role in the way organizations address their growing infrastructure needs, regional expansions, and redundancy plans. You can design, configure, and manage a cloud network to establish connectivity between your cloud, on-premises, and hybrid workloads.

Connectivity within the cloud environment

As you create virtual networks in your cloud environment you need to establish connectivity between the different workloads and their components. These connections within the cloud can be consumed in a service model, where you grant access between the workloads to connect and retrieve or send data. On the cloud, your workloads can have private or public IP addresses, allowing connectivity between them.

As you start building your cloud environment, different workloads will be hosted in different networks, and they may communicate through the Internet. However, as your environment grows and matures, you can configure your network so the traffic does not leave your cloud environment unless necessary. If you are working with Partners, or SaaS providers, you can work with them to establish a private connection, so the traffic never leaves your cloud environment.

There are multiple ways to establish network connections to ensure the traffic within your environment is secure. You can establish VPN connections between different networks or services, you can connect the different networks and access points through the route tables of your network benefiting from your cloud provider backbone network, or you can establish a physical connection between two locations.

Keeping the traffic within your cloud environment can help you reduce unnecessary data transfer costs between networks when you communicate with a partner, a SaaS provider, or to your data centers, since the traffic does not leave your network, and no internet connectivity is required. Additionally, it can help you to enhance your network security, reliability, and reduce latency when using private connection, since you will not be sharing bandwidth through the internet.

To improve discoverability of your workloads, services, and any possible partner product you are consuming, you need to plan and build your internal and external DNS configurations for private and public access.

Network designing and planning

Your workloads use IP addresses to communicate with each other, within your cloud environment, or in hybrid environments with resources outside your cloud environment. IP address management is an essential component for you to manage your network when you plan the different stages to deploy and operate your workloads. Without considering how your IP address space will be allocated, you run the risk of overlapping CIDRs and IP address exhaustion which can lead to network or service outages. Having a plan on where CIDR space will be allocated to will help you build your foundational environment to include support routing rules and route optimization.

CIDRs

In traditional on-premises networking you assign Classless Inter Domain Routing (CIDR) ranges to your LAN for private communication. Similarly, when creating networks in your cloud environment you also need to assign non-overlapping CIDR ranges to your network. Non-overlapping IP spaces allow you to build optimal routing that helps enhance your network performance and avoids intermittent connectivity issues and communication failures. There are cases were overlapping IP ranges cannot be avoided and it is important to design network routes or translation devices to ensure the communication between these networks performs as expected. When planning your network CIDR range, we recommend that you leverage a contiguous CIDR range for geographic regions, locations, or even services. By grouping your CIDRs you can classify or categorize them to help administrators easily identify and appropriately create routes.

IP address utilization

An important aspect of planning and designing your network is to ensure the risk of IP address exhaustion is mitigated. IP exhaustion happens when the number of IP addresses you need in a given network is greater than the IP addresses available. This is a big risk in the cloud environments due to the elasticity the cloud provides. As your workloads scale up and down, they consume more IP addresses in the given network. If there are no IPs available in your network, scaling will fail and you are at risk of having service impairment. When planning, you need to ensure that the IP range is large enough to accommodate networking and non-networking resources in your cloud environment. Addressing the size of the CIDR for current and future uses should be considered in planning.

Virtual networks

Virtual networks allow you to isolate resources from one another based on your operational requirements. Leveraging virtual networks enables your team to define a strict network boundary

between resources, which can ensure network isolation and mitigate risk of resources in one virtual network from access resources in another virtual network. For example, developers can have separate networks to create and develop resources isolated from production workloads. Issues or risky activity developers are taking within one virtual network will not impact network resources within another virtual network. At this stage, you should gauge short- and long-term projects that might affect network topologies such as merging of organizations, large data center migrations, and adoption of new vendors or technologies. Network admins and leadership should be ready to define virtual networks and restructure sub-networks and routing, and switching and physical layer networking modifications to establish communication across their network.

IPv4 and IPv6

Ever decreasing IPv4 space has posed challenges to IT and networking departments worldwide. The development of IPv6 is one of the most important advancement in networking technology. It not only solves the impending problem of IPv4 exhaustion, it simplifies routing, and provides an almost infinite pool of addresses which makes mobile networks and Internet of Things (IoT) devices easy to deploy and configure. IPv4 networks come with the challenges and complexities associated with planning a private network's IP schema. With the limited IPv4 space, a key design consideration is to decide how much space to allocate to a given application based on its requirements. Using IPv4 spaces often leads to the following design constraints:

1. Network architectures are designed too small, which requires you to expand the size of the network by adding new CIDR blocks to the network.
2. Network architectures are designed too large, which requires customers to accept overlapping IP, causing connectivity issues, and impacting the performance of the network.

IPv6 uses 128 bits instead of the IPv4 32 bits, which essentially eliminates any size considerations, allowing you to create unique IP addresses, to almost eliminate the overlapping concerns.

Network configuration and management

The network is the backbone of any IT infrastructure. Whether its post deployment operational issues or troubleshooting, you often need to make changes to individual components in the network, resulting in changes to the overall topology. These changes include modifications of route tables, allocation of new private and public IP addresses, and troubleshooting (connectivity, packet loss, throughput, bandwidth consumption, or latency issues) within and outside each team’s resources and your organization. A robust network architecture reduces the needs to make major changes to your network topology when any of these scenarios are encountered.

Centralized or distributed network configuration and management

As you build your network, you need to decide between distributed and centralized networking components for your organization. When starting distributed networks or Full Mesh routing solutions with smaller networks, on-premises sites, and cloud providers network may help reduce complicated point to point communication. However, as the network grows, and the number of nodes increases, the number of connections between the nodes increases the complexity for the network management introducing scalability challenges. Hub and spoke network topologies offer a different set of advantages over point to point communication links. Centralized networks (hub and spoke networks) allow you to inspect all your traffic in one point, controlling ingress and egress traffic from your environment, and simplify the management of the connections. Hub and spoke networks offer larger scalability over new links, and they offer centralized control over spoke sites. With these kinds of tools, you can centrally control access to public and private networks, restrict users access to resources over LAN and WIFI networks, control the addition and deletion of routing components, and simplify the modification of the logical divisions within your network (VLAN, VRFs, and so on).

Automating network infrastructure

In cloud environments, network engineers not only configure the networks, but they should also accustom themselves to orchestration tools using Infrastructure as Code (IaC) to deploy network infrastructure at scale. This helps in reducing deployment time, minimizes human error when configuring repeatable patterns, and provides ease in lift and shift for your current network infrastructure if it needs to be replicated in a different environment.

IP address management can be controlled and automated using IP Address Management (IPAM) solutions. IPAM solutions help enhancing dynamic allocations of IP addresses, adding/deleting of new and existing CIDRs, automated delivery and record maintenance. Automation to manage your network helps remove delays in on-boarding new applications or growing existing applications, by enabling you to assign IP addresses to your network resources. IPAM solutions can also automatically track critical IP address information, including its assignment attributes, location in your network, and routing and security domain. Reducing the need to manually track or do bookkeeping for your IP addresses and reducing the possibility of errors during IP assignments.

Traffic inspection

In hybrid environments, you should consult with your networking and security teams on traffic inspection requirement and identify application traffic which needs to be inspected. Typically, any traffic leaving your or cloud infrastructure and egressing to public internet can be prone to

attack. To mitigate this risk, any traffic within your network generated by a critical workload should be routed through an inspection device. While designing your network topology, your security team and your engineering team should collaborate with your networking team to build the level of granularity of the inspection given the traffic to be inspected. Some examples include: the direction of the traffic (east-west, north-south), the protocols, and origin and destination. These inspections need to meet the compliance and forensic requirements within your policy. Additionally, we recommend that you assemble high availability (HA) inspection devices to avoid a single point of failure.

Hybrid environment set up

As customers build their cloud environment it is common to have workloads on-premises and across different cloud providers. You might have connectivity requirements that you need to satisfy, so your workloads can communicate across different networks and environments.

The easiest way to connect different environments is over the public Internet. However, the public internet is a shared network, this can impact your performance, and the traffic is not encrypted at the network layer. We recommend that communications between critical and production workloads do not communicate through the internet, and limit over the internet communications to isolated test or development environments.

If you need to access resources in your cloud environment by using the internet, we recommend you create a bastion box within your environment. Using a bastion box allows you to access all other cloud resources privately which increases the security of the resources you don't need to expose on the internet. You can monitor access to your cloud environment and resources through this host.

The needs of your Business Units (BUs) often drive long and short-term connectivity initiatives. We recommend you work with your central IT team, security team, and the different workload owners to understand bandwidth, throughput, and compliance requirements. As you build your network SSL VPNs are great options for service connectivity and user to endpoint models. SSL or similar VPN clients can be installed on individual users' systems/servers/laptops to secure connectivity between your cloud environment and your on-premises and individual user devices. Virtual Private Networks (VPN) or a direct dedicated connectivity links between on-premises and cloud environment are some of the popular options chosen by customers. Both of these options typically require layer 2 and layer 3 connectivity between your data center/office and cloud provider's network. With a Virtual Private Network (VPN), you are encrypting traffic between two endpoints and traffic flows over private IP inside VPN tunnels. With physical links, traffic traverse

over dedicated lease lines so you get dedicated bandwidth and predictable network performance compared to VPN.

To configure routing for your networks, you can select between static or dynamic routing. Static routing requires to the manual configuration of routes for source and destination on both ends. Dynamic (BGP or OSPF) routing allows you to propagate routes across your network automatically. For most network option configurations, we recommend a dynamic routing solution for scalability and adaptability.

Edge connectivity is gradually becoming preferred option for applications with global user base. Content Delivery Networks (CDNs) and similar edge networking solutions employ edge locations distributed worldwide. This enables application traffic access through one of the nodes that is closer to the final user, reducing significantly the usage of the Internet Service Provider (ISP) network. These solutions are used to improve the end customer experience by enhancing performance of the network and workloads.

Regardless of the type of connectivity options you choose between your on-premises environment and your cloud environments, ensure **high availability** and **redundancy** for your communication channel. Routing maintenance, upgrades, other unexpected network, electrical incidences, and natural events can disrupt your primary connection. In these scenarios, a secondary (or fail over) connection might be required to avoid outages.

Establish logging and monitoring set up

Observability in your network is critical to maintaining optimal performance and mitigating risks. Network logs contain information like IP addresses, ports, protocols, and the kind of traffic being sent through your infrastructure that allows you to understand how your network is operating. Network logs, including application traffic for payload inspection, can be used to identify and perform corrective actions when unauthorized or malicious traffic is discovered. Network logs can also be used to troubleshoot network issues including connectivity and performance. Centralizing the network logs for analysis will help you reduce the complexity of a solution that works across all the devices generating traffic and logs in your network. Another benefit of centralizing these logs is the *ease of use* of a solution that can analyze traffic to identify patterns monitoring your network, and to perform proactive and reactive remediation actions.

During peak traffic and load for your workloads, your underlying infrastructure needs to be resilient and deliver the expected performance for your users. We recommend that you collect performance metrics of your infrastructure, that can help you identify if network is optimized correctly and adjust as needed to enhance its performance. In cloud environments, all these

metrics can be pushed to same monitoring dashboard and from single location basically you are monitoring your entire network and performance. These dashboards can also implement alerts and execute different automations based on the data collected when unusual activity is detected. This helps IT teams to reduce the time to remediate events identified in your network, and avoiding issues with your workloads.

Construct DNS set up for applications

A Domain Name Service (DNS) is used by every application and organization to connect names to IP addresses. In cloud environments, DNS management becomes a critical element, since it allows for the discovery of each workload and service among themselves. Organizations usually manage their own DNS servers or leverage public DNS systems to make DNS queries. Within a cloud environment you can create DNS zones to publish your records, without having to configure and maintain your own DNS servers and software.

We recommend you manage your internal workloads and server domain names in a private DNS zone, limited to your network. For your public facing workloads and services, we recommend that you set up a different DNS, the DNS should be publicly accessible for clients over internet to connect.

Typical application architecture involves monolithic, virtual machines (VMs), or containerized version of jobs and services, Relational or key-value databases and other data sets, and front-end consisting API routing or Load Balancer. Each of these constructs need unique networking planning and designing. Domain Name Service (DNS) plays a key role in forwarding traffic to your applications irrespective of their structure. You should work with your application team to evaluate their needs for user experience and application monitoring. You can leverage DNS health checks provided by a cloud provider to test your applications or databases to ensure they are healthy in production environment. Similarly, DNS policies can be applied for your application traffic in cloud to influence routing. For example, you can use DNS routing policies to control which endpoints serve content to users based on the specific geographical location of your applications, users, or latency. You can also configure DNS records to achieve active/passive or load balancing between two endpoints.

For customers with hybrid workloads, which include on-premises and cloud-based resources, extra steps are necessary to configure DNS to work seamlessly across both environments. You can use different endpoints on AWS for DNS traffic to be routed in and out of your cloud and on-premises network. Further, centralizing DNS management for all your cloud network and on-premises domains should be considered for ease of management. Additionally, when you have

hybrid connectivity between on-premises and cloud architecture of the DNS is one of the main players in the room as resources on-premises will need to resolve DNS names for resources in cloud and vice-versa.

Finally, to get insights into DNS traffic for auditing; DNS logging should be enabled and pushed to a monitoring system to get more insights into DNS data and perform necessary actions from that data.

Change Management capability

The Change Management capability enables you to manage risk and minimize negative impact when making changes to your cloud environment. This includes the ability to request, plan, track, deploy, and roll-back changes to your environment.

Stakeholders:

- Central IT (Primary
- Operations
- Security

Personas:

- **Cloud Team** - the team(s) who make cloud available to customers (such as App DevOps).
- **Information Security Team** - the team responsible for security in the cloud.
- **Central IT Operations** - The team(s) that oversee all the IT operations within the environment.

Scenarios:

- **CF30 - S1: Change management process**
- **CF30 - S2: Change management fulfillment**
- **CF30 - S3: Change rollback**
- **CF30 - S4: Change monitoring and assessment**

Topics

- Overview
- Establish a change management process
- Change Management categories and priorities
- Define change management fulfillment process
- Recover from failed changes
- Establish mechanisms to access, review, and monitor changes

Overview

The purpose of change management is to control the lifecycles of all changes, allowing changes to be made with minimal interruption to IT services. Changes to an environment introduces risk. Authorized changes should be prioritized, planned, tested, implemented, documented, and reviewed to mitigate any risks before deployment.

The Change Management capability ensures that changes are understood, recorded, evaluated prior to, during, and after implementation. It also complements the process and safety controls of your continuous integration (CI) practices and Continuous Delivery/Deployment (CD) methodology. The Change Management capability is ***NOT*** intended for changes made as part of an automated release process, such as a CI/CD pipeline, unless there is an exception or approval required for major or broad changes.

ITIL defines change management as “the process responsible for controlling the Lifecycle of all Changes. The primary objective of change management is to enable beneficial changes to be made, with minimum disruption to IT Services.” Every change should deliver business value and the change management processes should be geared towards enabling that delivery. ITIL states a number of benefits for effective change management, including “reducing failed changes and therefore service disruption, defects and re-work” and “delivering change promptly to meet business timescales". (ITIL Service Transition, AXELOS, 2011, page 44)

The key concepts of change management remain the same in the cloud as on-premises. Change delivers business value and it should be delivered efficiently. Agile methodologies and the automation capabilities of the cloud go hand in hand with the core principles of change management as they are also designed to deliver business value quickly and efficiently. There are some key areas, such as the management of infrastructure with software defined solutions,

that may require existing change processes to be modified to adapt to new methods of delivering change.

Establish a change management process

Change management practices are designed to reduce incidents and meet regulatory standards. These practices ensure efficient and prompt handling of changes to IT infrastructure and code. Modern change management methodologies can include rolling out new services, managing current ones, resolving problems in code, breaking down silos, providing context and transparency, eliminating bottlenecks, and minimizing risk.

The change control practice ensures that *risks are properly assessed, authorizing changes to proceed and managing a change schedule in order to maximize the number of successful service and product changes*.

There are multiple ways to establish network connections to ensure the traffic within your environment is secure. You can establish VPN connections between different networks or services, you can connect the different networks and access points through the route tables of your network benefiting from your cloud provider backbone network, or you can establish a physical connection between two locations.

Change management process

Change management scope

Change management is the practice of monitoring resource configurations to establish and manage a baseline. A baseline is a snapshot at a given point in time of a set of configurations. This snapshot enables you to identify different configuration states of a given configuration item (a snapshot of a particular configuration at a point in time). The purpose of your change management should be to control the changes made to the baseline in a safe and controlled method. The scope of your environment baseline needs to be defined so it doesn't conflict with any DevSecOps practices that deliver their own change management through a release management process. Any configurations that are not managed by a DevSecOps release management process should be tracked as a change management process. You will also need to determine which services and configurations necessitate change control which are outside of the management of templates or CI/CD operations. Common configurations item for change control can be:

1. Enterprise-wide configurations completed one time via manual effort
2. Temporary suspension of enterprise-wide policies

3. User access or membership changes to groups
4. Centralized changes that impact multiple groups

> **Note**
>
> DevSecOps is the preferred method to implement change and should follow the general principals of change control. Change management as defined in this capability is inclusive of DevSecOps changes in regard to progressing change fulfillment to an automated process leveraging infrastructure as code.

Change Management categories and priorities

Categories

Changes can be grouped and categorized based on perceived level of impact and urgency. Changes come in three categories: normal, standard, and emergency.

- **Standard Change** – This change is an established change that is low-risk and well understood; therefore, it does not need a formal review and approval process. These changes utilize a condensed version of the normal change procedures that simplifies the process in order to quickly satisfy the change requester's need. For example, when a user requests for an internal storage resource to be created within your environment. This change request has low risk and can frequently be automated with a self-service change management fulfillment service. Standard changes are submitted via a request for change process and reviewed by approving body sometimes know and a Change Advisory Board (CAB).
- **Normal Change** – This change is unique and has a higher risk of uncertainty of the anticipated outcome. This is typically defined as the default change and warrants a formal review and approval process to initiate the change implementation. For example, when a new service requires a firewall rule to be updated to allow incoming from a non-standard port.
- **Emergency Change** – This change addresses unforeseen operational issues, such as failures, security threats, and vulnerabilities. This type of change is a rapid change that is required to continue or restore the business operations, to address significant risk, or to solve emergency business needs. Emergency changes should still follow documented procedures and use the organizational emergency change management process; however, the approval process is streamlined by defining a smaller approval body commonly known as the Emergency Change

Advisory Board (ECAB). The ECAB is a special group convened to advise on approval/rejection and planning for emergency changes. The membership to the ECAB includes people with experience and authority to assume risk to make rapid decisions.

Depending on the categories you choose, you will need to provide a process for approval. Standard changes require a formal review process and approval from a change approving authority such as a CAB you will setup. Emergencies require a fast reaction; however, they still warrant an approving body. Depending on the risk your organization wants to take, change approval should be documented and have some type of approving authority for anything outside of the day-to-day changes. However, not all changes require a formal review and approval; they merely require some form of request submission and automated approval. As customers move toward automation, change management can be done in a highly automated and scripted manner with minimal manual input.

> **Note**
>
> Recurring changes that are tasks with low risk should be categorized as standard and auto-approved. Change management should focus on enablement and not become bureaucratic roadblock to delivering value.

Priorities

Changes should also be defined with a priority level to assist in identifying the impact and urgency of the change. Impact is a measure of the effect of an incident, issue, or change on a business process. Impact is often based on how service levels will be affected. *Urgency* is a measure of how long it will be until an incident, problem, or change has a significant business impact. For example, a high impact incident may have low urgency if the impact will not affect the business until the end of the financial year. Creating a matrix to assist in identifying the priority of change request helps facilitate a standardize process to identify the priority of the change and schedule the execution of the change appropriately.

Initiating request for change

An essential part of change management is the request for change process, documents, and end-users request for change and triggers the review and approval process. Changes that fall within scope for your change management process will require a *Request for Change* (RFC) to be

submitted for review and approval. A change request is a formal communication looking for a modification to one or more configuration items within scope of your change management system. A request for change can include the following:

- RFC Document
- Service Desk call
- Project initiation document
- Tickets generated from IT Support service

We recommend that you determine the applicable information to support your change control process. Depending on the category of change, a request for change can capture different levels of detail. Basic information that needs to be captured is the service or configuration item that will be change, who is submitting the change, and justification for the change. There needs to be enough information that the approving authority can approve or deny the change. The following table gives an example of basic information which should be captured on all RFCs:

RFC data point	Description
Submitted By	Point of contact for change initiation
Date Requested	Date the request for change was submitted
Title of Change	Title of the change
Description of Change	Brief description of the change
Justification of Change	Reason change is necessary
Identified Risks	A list of identified risks and their risk mitigatio n efforts
Impact Scope	The impact the change will have within the environment
Schedule	Schedule of the maintenance window for the change to be completed in
Suggested Implementation	Plan to implement the change

RFC data point	Description
Rollback	Plan to rollback change if necessary

Change communication

The change management process fulfills the review, approval, and orchestration of changes. A cohesive communications plan is a critical component for success to help further mitigate risks when making IT changes. When all the right people are involved - the technical manager, the business manager, the people making the change, and the business users – all the right tools are provided, and all the right fallback actions are tested, a communications plan helps ensure a comprehensive and successful change.

An important part of the process is to communicate the intended change and, if necessary, coordinate a maintenance window to implement the change. Change management requires that all stakeholders for the change are notified if the change will impact their service or operations.

Change management tools

To facilitate configuration and change management, manual or automated tools can be used. Tool selection should be based on the needs of the project stakeholders including organization and environmental considerations and/or requirements.

Change management tools can integrate with the operating environment facilitating workflows that deliver communication, change orchestration, approval processes, and configuration item baseline tracking and monitoring. This type of change management tool more often requires an advanced environment setup that support workflow templates for automation and take considerable amount of planning and setup. They require an understanding of the scope of change you desire to manage and how well-defined scripted steps for change implementation.

Define change management fulfillment process

Approved RFC's will need to go through a process of fulfillment which includes scheduling and executing the approved change. A key consideration is understanding the risk of deploying the change to the cloud. The goal of the change management fulfillment process is to understand the risk and provide risk migrating efforts. A proven strategy to mitigating change risk is to standardize the method to execute types of change. Organizations can create their own runbooks to execute a standardized or coded series of steps that include change validation and potential rollback

plans. This ensures repeatability and consistency across multiple environments, as well as enabling automation of software testing, compliance testing, security testing, and functional testing.

The change management fulfillment process should always be working to improve itself and add to its runbooks to facilitate faster change with mitigated risks. As you develop your own runbooks, you should review your categories of change attempting to move any normal change to an automated standard change that can be self-service using an integrated change management service or solution.

Recover from failed changes

Changes should not be approved without considering the consequences of a failure or degraded performance outcome. There should be a back-out or rollback plan which will restore the resource to its initial baseline configuration prior to the executed change. The cloud enables *rollback* plans to be fully automated using repeatable processes. Not all changes are reversible. However, the process to recover from failed changes should be documented and the risk accepted when approving the change. If failed changes have a much lower impact due to the speed and consistency of roll back, activating roll-backs should be considered to be part of the normal process. This is particularly true if it is possible to quickly remediate the issue and push it through the same automated pipelines to quickly deliver the original intended business value of the change.

Establish mechanisms to access, review, and monitor changes

All changes within the scope of your change management should be monitored and tracked for approved and out-of-band changes within the environment. Every change that is approved and implemented should have a positive impact to the overall service such as, a security gain or performance gain within the environment. Changes that are tracked and have adverse effects to the environment should be rolled back immediately which will restore the baseline to its previous state. The goal of continuously monitoring your baseline and alerting on configuration items changes is that all change introduced into the environment is controlled. Changes that happen outside of the management process need to be tracked down and reverted back to its previous baseline or documented in your change management system.

Cloud Financial Management capability

The Cloud Financial Management capability provides the ability to manage and optimize your expense for cloud services. This capability enables you to track, notify, and apply cost optimization

techniques across your environment and resources. Expense information is centrally managed and consumed, and access to critical stakeholders can be provided for targeted visibility and decision making.

Stakeholders:

- Finance (Primary)
- Central IT
- Operations
- Software Engineering

Personas:

- **Finance Team** - the team involved in defining and approving cloud budgeting and forecasts, business case development for new workload migrations, defining a cost allocation mechanism (including accounting treatment for cloud spend), analyzing cloud spend for insights, performing variance analysis, participating in developing a tagging dictionary, and integrating existing finance processes (accounts payable and procurement) with cloud billing and procurement.
- **CFM/FinOps Team** - single-threaded owner (individual, or team) that defines and oversees the programmatic implementation of the Cloud Financial Management uses case.
- **Software Development Team** - the team that builds cloud-based software that is cost-aware, applies changes to software that reduces cloud waste, provides inputs to an organization's commitment-purchase process to improve the accuracy of commitments being purchased, participates in the cost forecasting process for existing and new cloud products, and participates in variance analysis and root cause identification for unexpected/anomalous cloud costs.

Scenarios:

- **CF16 - S1: Cost allocation**
- **CF16 - S2: Cost forecasting**
- **CF16 - S3: Cost monitoring**
- **CF16 - S4: Cost optimization**

- **CF16 - S5: Financial operations**

Topics

- Overview
- Cost allocation
- Planning and forecasting
- Cost monitoring and reporting
- Cost optimization
- Cloud financial operations

Overview

Having visibility and understanding of the spend in your cloud environment is critical to your business. Setting up the right measures to monitor your resources will allow you to create reports, dashboards, and anomaly detection of the cloud spend of your budget and plan for cost optimization to avoid or reduce unnecessary spend.

Implementing these mechanisms and tools will help you support business decisions and establish cloud financial operations in your environment to socialize cost awareness across different business units, application teams, and other stakeholders without affecting the innovation of your teams.

In order to implement a cloud financial management function, you need to implement a tagging strategy for your environment. Refer to the tagging capability, to find recommended tags for your environment. Some of these tags within the tagging capability can be used to track spend in your cloud environment, and allow you to create dashboards, reporting for individual business units, workloads, and types of environments

You need to incorporate the tags and the values for these tags defined within your cloud financial management capability into the tagging dictionary defined as part of your tagging capability. We recommend that you make these tags widely available across your different stakeholders and teams in order to enable them to track spend back. This will allow your cloud team or your FinOps team to analyze the usage and work towards building a cost allocation strategy.

Cost allocation

Leveraging metadata across your environment, helps you to accurately allocate cost for workloads and applications. Using a showback approach you can identify the costs incurred by a business unit, product, or team. However, material spends, may not be accounted for due to the lack of enforcing tagging mechanisms. Grouping your different resource and establishing boundaries between them can help you identify how those groups of resources are using the budget assigned to a specific business unit or across different stages software development lifecycle of a workload.

Creating categories to consolidate groups of resources based on your business needs (some examples of categories would include: Business Unit, Product Line, Environment). Once these categories are established and set up, you can use them to monitor cost and usage information within these categories. Additionally, you can retrieve meaningful information from these groups of resources leveraging multiple dimensions (such as, who created a resource). For example, the Line of Business where a resource belongs.

Leveraging infrastructure as code to deploy infrastructure and resources for your workloads will allow you to consistently apply tags from your tagging dictionary, avoid untagged resources, and enforce your tagging policy to ensure the cloud financial management capability allows you to allocate costs for your infrastructure when needed.

With the appropriate mechanisms in place, the Cost Allocation methods allow you to carve out material cloud charges, including commitment purchases, standalone, and shared resources. For example, networking, log retention and archival, security tools, and operational tools charges. Establishing chargeback mechanisms will allow you to report costs incurred by different business units, products, and teams.

Planning and forecasting

If the workload you bring to your cloud environment is a new cloud native based workload, or a migration from your on-premises data center you need to model and plan costs. Using cloud native tools, you can extract data that will allow you to determine a total cost of ownership (TCO) to quantify the expected cloud costs. Additionally, for your overall environment, there are other non-cost cloud values that you need to consider, including staff productivity, operational resiliency, and business agility, which will showcase the business value of moving to the cloud.

We recommend regularly reviewing cloud budgets, to understand different variances, so you can plan ahead for spending. We also recommend performing forecasting exercises to define future IT and workload-based budgets. Cloud forecasting can be performed by using a combination

of trend-based, and driver-based methods to closely align to future cloud usage (such as, new products, new launches, or changes in cloud deployments) as well as future cloud demand (such as, forecasted customer demand for cloud-hosted products). Cloud spend planning should be a part of the organization's overall IT financial planning process, which may include on-premises or other hybrid spend planning.

Cost monitoring and reporting

Visibility across your cloud environment is critical. Your cloud team and your finance teams will require visibility into the cloud spend, and stakeholders owning business units and workloads need the ability to generate and save custom reports. Doing so, will create a cost-aware culture within your organization.

Specific reports generated for the different workloads in your environment will allow different stakeholders to create different views of the environment, based on groups of resources, environments, or stages of development, and will enable them to forecast the spend or detect anomalies in costs during a specific time frame.

The level of granularity and visibility in reports can be enabled at different levels for different dashboards, and is granted based on roles and groups for each of the workloads via federation. A few examples are:

1. In order to analyze and monitor your entire environment, your cloud team and/or your FinOps team will need access to the environment level billing. From here, they can set up budgets, analyze spend trends, and detect anomalies across the entire environment.
2. A business owner needs visibility across the different workloads and applications, access can be granted for the specific group of resources or custom dashboards to visualize the spend for each of the workloads.
3. A builder, a developer, or an individual user needs visibility, access can be granted to visualize the cost associated with the resources they are using, or within their sandbox environment.

Setting up the right monitoring tools for spend in your environment will allow you to evaluate different spend patterns. Evaluating your patterns weekly or monthly with these reports, can help you determine if any anomalies in spend patterns exist, and quickly identify and remediate the root cause.

Defining metrics that allow you to identify if your cost strategy is successful, for example `unit-cost`. This will feed into your overall finance strategy, allowing your technology and business

stakeholders identify cost-saving opportunities and repurpose some of the savings to new initiatives or projects to enhance your cloud environment.

Cost optimization

Cost optimization is treated as an essential component when performing tradeoffs between various designs and architectures during the early stages of product ideation. Cost optimization for your environment can happen across the entire life of your environment and workloads, from the design and architectural stage, all the way until the resources have been launched on the cloud. This can include proof-of-concept designs to help you estimate the cost before moving any resources to non-production or production environments.

Using a centralized model to acquire and manage billing and costs can benefit your environment as you can reserve resources or purchase saving plans that can be shared across all the teams. Grouping resources and managing them centrally throughout your environment can also offer the benefit of usage volume discounts, that you will receive when all the billing is consolidated. Each team will get recommendations and information from the central team and tools to optimize their usage and workloads, and should implement resource modifications before these commitments are in place to benefit from the discounted prices. As the environment and the workloads are designed, managing network and licensing centrally will reduce the overall cost and overhead for individual teams.

Each team should analyze the resources they own once they are deployed, to identify cloud waste associated with each workload. These recommendations include the size of the compute that is recommended, different tiers of storage available, and any committed pricing models that can be leveraged. Additionally, each team should analyze and identify opportunities to modernize their environments to use the newest tools and technologies that often include better performance ratios, leading to a reduction of the cost for the workloads.

Cloud financial operations

Cloud financial operations focuses on capabilities that allow customers to evolve organizations, processes, automation and tools, and establish a self-sustaining cost-aware culture of innovation. This ensures that stakeholders across organizations have a common understanding of cloud costs, and can be done through establishing new working partnerships between finance and technology teams to allow more accurate budgeting and cloud spend monitoring. Reporting, education, gamification, and celebrating efficiency wins can drive organizational cost awareness and help foster a cost-aware culture.

We recommend establishing a centralized function (individual or team) that owns cloud financial management activities across your cloud environment. This central function provides and controls access to the billing and costs tools, co-owning a cross-organizationally approved tagging dictionary defining cost categories, managing commitment purchases, and being a primary business partner for the finance organization. This central function is responsible for driving awareness about how the cloud environment is being used within the organization, performing budget reviews, and helping with forecasting exercises.

This function is also responsible for defining and implementing a cost management tooling strategy, which may require the procurement and curation of partner tools for internal consumption, or identify internal resources to build in-house cost management tools. Tools should be built to automate as many cost management activities as possible to reduce undifferentiated work, and to enable scale. In addition to a centralized approach at commitment purchases, the cloud financial management function augments existing business and technical processes to instill cost-awareness (such as, introducing cost into change management, incident management, and service operationalization/readiness processes), maintain direct relationships with technical, finance and business stakeholders to raise cross-organizational cost awareness (such as, through hackathons, all hands meetings, frugality awards), ensure cost transparency as it pertains to the business being supported by cloud (such as, KPI development, cost reporting), and drive cloud spend concerns to closure (such as budget variances, spend anomalies, root cause identification, and remediation).

Next steps

If you are still exploring the cloud, AWS recommends that you deploy a few proof-of-concepts (POCs) to demonstrate business value to your stakeholders. **If you are ready to start building a cloud environment** to host your workloads on the cloud, this set of defined capabilities can help you build your foundational cloud environment. Before getting started with your cloud adoption, AWS recommends that you complete the following activities, and reach out to your account team for more information:

- Review the list of capabilities and create a timeline for implementing capabilities, accounting for any dependencies.
- Identify the stakeholders in your organization that are responsible for each capability.
- Create an implementation plan and a timeline to build your cloud environment.

As your requirements change, to help you grow your presence in the AWS Cloud, you can use the defined capabilities to build your own approach using your own tools.

Conclusion

This whitepaper introduces a capability-based approach to establishing the foundation for your AWS environment, and helps you identify the relevant stakeholders needed to make important decisions along your journey. The defined *capabilities* in this paper are based on current AWS best practices, and the experience of thousands of customers that have built their foundational environment on the AWS Cloud.

Contributors

Contributors to this document include:

- Alex Torres, Sr. Solutions Architect, Amazon Web Services
- Fabian Labat, Sr. Solutions Architect, Amazon Web Services
- Glenn Dasmalchi, Sr. Manager Tech Leader, Amazon Web Services
- Sam Elmalak, Tech Leader, Amazon Web Services
- George Rolston, Sr. Solutions Architect, Amazon Web Services
- David Rowe, Sr. Solutions Architect, Amazon Web Services
- Brandy Smith, Sr. Solutions Architect, Amazon Web Services
- Todd Gruet, Sr. Solutions Architect, Amazon Web Services
- Saransh Burman, Solutions Architect, Amazon Web Services
- Rahul Gangal, Technical Account Manager, Amazon Web Services
- Emily Arnautovic, Principal Solutions Architect, Amazon Web Services
- Mutalip Dirik, Sr. Solutions Architect, Amazon Web Services
- Jason DiDomenico, Sr. Solutions Architect, Amazon Web Services
- James Kane, Solutions Architect, Amazon Web Services
- Jarrid Kleinfelter, Sr. Solutions Architect, Amazon Web Services
- Ryan Lempka, Sr. Solutions Architect, Amazon Web Services
- Julian Busic, Solutions Architect, Amazon Web Services
- Matt Berk, Sr. Technical Account Manager, Amazon Web Services
- Geno Erickson, Sr. Technical Account Manager, Amazon Web Services
- Cy Hopkins, Sr. Solutions Architect, Amazon Web Services
- Marco Fiorelli, Technical Account Manager, Amazon Web Services
- Yuliya Linetskaya, Sr. Technical Account Manager, Amazon Web Services
- Levon Stepanion, Principal BDM, Cloud Financial Management, Amazon Web Services

Further reading

For additional information, refer to:

- AWS Architecture Center
- AWS Whitepapers & Guides

Document revisions

To be notified about updates to this whitepaper, subscribe to the RSS feed.

Change	Description	Date
Whitepaper updated	Whitepaper updated to add guidance for additional capabilities	September 13, 2023
Whitepaper updated	Whitepaper updated to add guidance for additional capabilities	February 8, 2023
Whitepaper updated	Whitepaper updated to add guidance for additional capabilities	September 30, 2022
Whitepaper updated	Whitepaper updated to include capabilities guidance	May 2, 2022
Initial publication	Whitepaper published	November 17, 2021

Appendix A: Capability structure and example

Capability structure

Definition

The definition includes a high-level description of what the capability will help you enable in your cloud environment.

Scenarios

Scenarios are a set of use cases that expand the capability definition, and detail what parts of your environment the guidance included in the capability solves. Each capability provides a baseline, which establishes the minimum requirement for the capability, and can be expanded and customized to add additional scenarios based on your requirements or as your business needs mature and your AWS presence grows.

Guidance

This section outlines prescriptive recommendations for how the capability should be built in your environment to implement the included scenarios. It also includes responsible stakeholders, and a description of how the capability will work in the overall environment. Additionally, this section includes the people and recommended skillsets necessary to successfully establish the capability in your environment.

Implementation guidance

Each capability provides prescriptive, opinionated AWS guidance to establish the capability in your environment. Runbooks are included to help you operate the capability efficiently in your environment using AWS services.

Capability example - Log Storage

Definition

The Log Storage capability enables you to securely collect and store your environment logs centrally within an immutable storage. This will enable you to evaluate, monitor, alert, and audit access and actions performed on your AWS resources and objects.

Scenarios

- Your cloud team wants to log *individual user access* to resources, and what systems are accessed and actions taken (*Individual user access* also includes access by system administrators and system operators).
- Your cloud team wants to set controls to prevent modification of the related logs.
- Your cloud team wants to set controls to prevent unauthorized access to logs.
- Your cloud team wants to generate logs that can show if inappropriate or unusual activity has occurred.
- Your cloud team wants to store logs in near real-time for resiliency during a determined period of time (matching your governance requirements).
- Your cloud team wants the stored logs to be encrypted at rest.

Guidance

The Log Storage capability primary mapping is to the **Security Functional Area**. This means the **Security team** should be responsible for implementing this capability.

When establishing your capability, the builders owning the implementation will need to receive inputs from the owners of additional functional areas to ensure the proper interlock of the functions in the cloud environment. The list of secondary functional areas required are:

- Operations
- Central IT

Having a separated Log Storage allows you to establish a secure location where the logs become the source of truth to show what is happening in your environment related to security and operations. As your environment expands to accommodate your business needs, centrally aggregating the information will enable you to later build monitoring and observability capabilities, to monitor in near real-time what is happening across your environment.

The Log Storage must be secured, built for resilience, to avoid tampering with the logs, and only accessed by controlled, automated, and monitored mechanisms, based on least privilege access by role. The following controls need to be implemented around the Log Storage to protect the integrity and availability of the logs and their management process. The logs delivered to Log

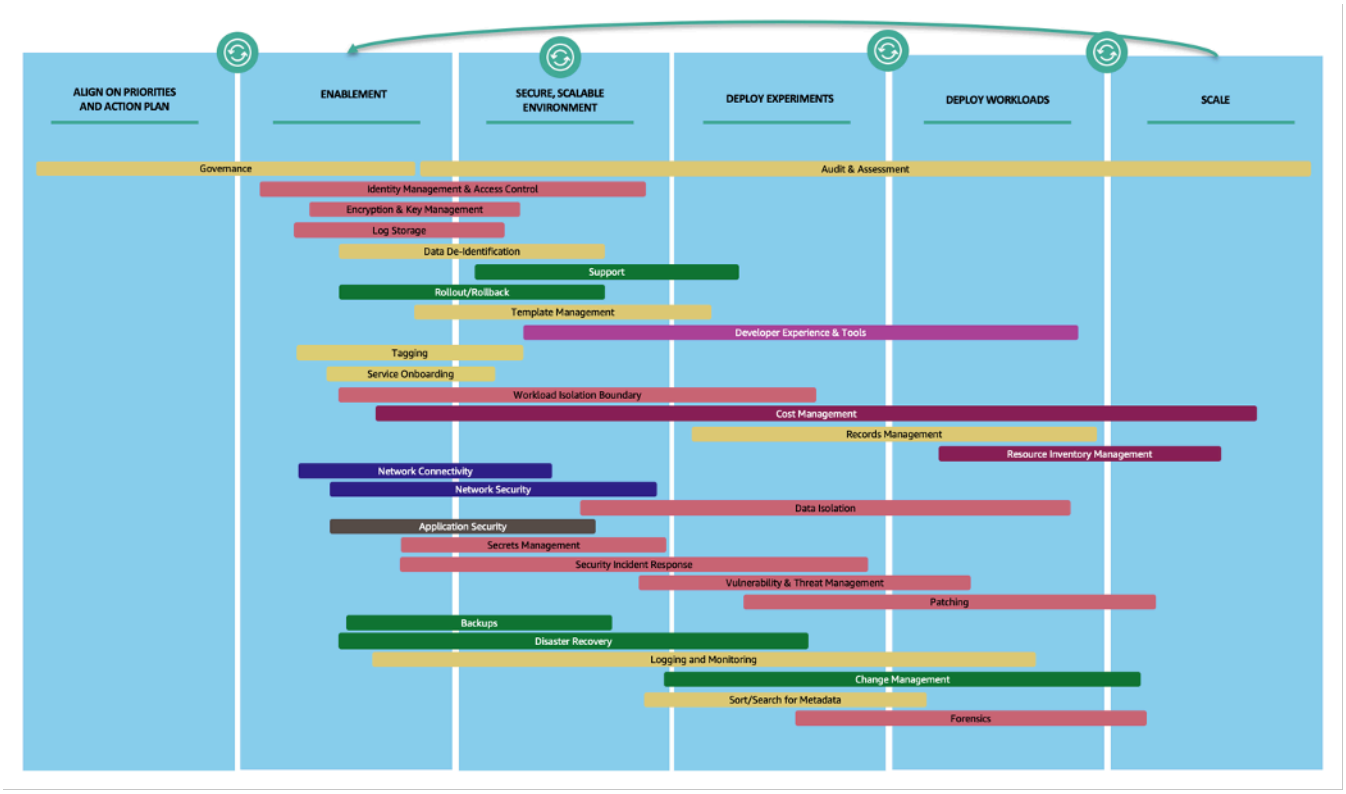
Storage should be encrypted, and the encryption key access and permissions should also be based on least privilege permissions.

- **Detective controls** should be implemented to alert and remediate the collection of permissions used on the log storage, and to actively monitor access to the logs within the Log Storage.
- **Preventive controls** should be implemented to protect from changes to your configuration and access in your Log Storage, and restricting permissions on your Log Storage.

The Log Storage should also have retention policies, establishing a lifecycle for your logs based on your governance and data retention policy requirements (for example, automatically archiving infrequent access or delete the logs over time to reduce the cost while meeting retention requirements).

Appendix B: Sample timeline

In this section you can find a sample timeline that includes all 30 capabilities that are needed to meet your requirements when establishing a foundational cloud environment on AWS. Enable your teams to work with the capabilities, and start building an environment which initially allows you to deploy experiments and then workloads. As you scale or your business needs evolve, you can assess your established capabilities, and enhance them as necessary to meet your requirements.



Sample timeline of capabilities

Notices

Customers are responsible for making their own independent assessment of the information in this document. This document: (a) is for informational purposes only, (b) represents current AWS product offerings and practices, which are subject to change without notice, and (c) does not create any commitments or assurances from AWS and its affiliates, suppliers or licensors. AWS products or services are provided “as is” without warranties, representations, or conditions of any kind, whether express or implied. The responsibilities and liabilities of AWS to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.

© 2023 Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS Glossary

For the latest AWS terminology, see the AWS glossary in the *AWS Glossary Reference*.