



Benutzer-Leitfaden

Amazon Elastic File System



Amazon Elastic File System: Benutzer-Leitfaden

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

Die Marken und Handelsmarken von Amazon dürfen nicht in einer Weise in Verbindung mit nicht von Amazon stammenden Produkten oder Services verwendet werden, die geeignet ist, Kunden irrezuführen oder Amazon in irgendeiner Weise herabzusetzen oder zu diskreditieren. Alle anderen Handelsmarken, die nicht Eigentum von Amazon sind, gehören den jeweiligen Besitzern, die möglicherweise zu Amazon gehören oder nicht, mit Amazon verbunden sind oder von Amazon gesponsert werden.

Table of Contents

Was ist Amazon Elastic File System?	1
Sind Sie ein Erstnutzer von Amazon EFS?	3
Funktionsweise	3
So funktioniert Amazon EFS mit Amazon EC2	5
So funktioniert Amazon EFS mit AWS Direct Connect und AWS verwaltetem VPN	7
So funktioniert Amazon EFS mit AWS Backup	8
Features	9
Authentifizierung und Zugriffskontrolle	9
Datenkonsistenz in Amazon EFS	9
Verfügbarkeit und Beständigkeit	10
Replikation	17
Erste Schritte	18
Voraussetzungen	18
Erstellen Sie ein Dateisystem und starten Sie die EC2-Instance	19
Übertragen Sie Dateien in Ihr Dateisystem	19
Voraussetzungen	20
Bereinigen von Ressourcen	21
Installation des EFS-Clients	22
Abhängigkeiten für EFS-Tools	22
Unterstützte Distributionen	23
Manuelles Installieren des EFS-Clients	25
Installation des Amazon EFS-Clients auf Amazon EC2 EC2-Linux-Instances	25
Installation des Amazon-EFS-Clients auf anderen Linux-Distributionen	26
Installation des EFS-Clients auf EC2-Mac-Instances	27
Automatisches Installieren des EFS-Clients	27
Was macht der Amazon-EFS-Client während der Installation	28
Von Systems Manager unterstützte Betriebssysteme	28
Konfiguration AWS Systems Manager für die Installation des EFS-Clients	30
Installation und Aktualisierung botocore	31
Upgraden von stunnel	32
Behebung von Problemen bei der Installation von Stunnel	34
Den FIPS-Modus aktivieren	35
Ressourcen erstellen und verwalten	37
Übersicht über die Implementierung	38

Ressource IDs	40
Erstellungstoken und Idempotenz	40
Dateisysteme erstellen	41
Erforderliche IAM-Berechtigungen für die Erstellung von Dateisystemen	41
Konfigurationsoptionen für Dateisysteme	42
Dateisysteme löschen	59
Erstellen von Dateisystemrichtlinien	60
Erstellen von Zugriffspunkten	62
Access Points löschen	65
Taggen von Ressourcen	65
Grundlagen zu Tags (Markierungen)	66
Tag-Einschränkungen	66
Verwenden von Tags für die Zugriffskontrolle	67
Markieren Ihrer -Ressourcen mit Tags (Markierungen)	67
Tutorial: Schreibbare Unterverzeichnisse pro Benutzer erstellen	68
Mounting von Dateisystemen	71
Überlegungen zur Installation von Linux	71
Verwenden der EFS-Mountinghilfe	73
Von EFS Mount Helper verwendete Mount-Einstellungen	75
Abrufen von Support-Protokollen	76
Voraussetzungen	77
Mounting auf EC2-Linux	79
Mounting auf EC2 Mac	81
Montage aus einer anderen Region	83
Mounting von One-Zone-Dateisystemen	83
Mounting mit IAM-Autorisierung	87
Mounting mit EFS-Zugangspunkten	88
Mounting auf mehreren EC2-Instances	89
Mounting von einem anderen Konto oder einer anderen VPC	90
Verwenden von NFS	94
Voraussetzungen	94
NFS-Support	95
Installieren des NFS-Clients	96
Empfohlene NFS-Mount-Einstellungen	98
Mouneten auf EC2 mit DNS	100
Mounting mit einer IP-Adresse	102

Automatisches Mounten von Dateisystemen	104
Neue EC2-Linux-Instances	105
Bestehende EC2-Linux-Instanzen	107
Linux- und Mac-Instances, die NFS verwenden	111
Aufheben des Mountings von Dateisystemen	113
Tutorial: Erstellen und mounten Sie ein Dateisystem mit dem AWS CLI	114
Voraussetzungen	115
Einrichtung des AWS CLI	116
Schritt 1: EC2-Ressourcen erstellen	117
Schritt 2: EFS-Ressourcen erstellen	123
Schritt 3: Mounting und Test des Dateisystems	126
Schritt 4: Bereinigen	130
Tutorial: Mounten mit lokalen Clients	131
Voraussetzungen	133
Schritt 1: Erstellen Sie Ihre EFS-Ressourcen	135
Schritt 2: Installieren des NFS-Clients	137
Schritt 3: Mounten Sie das EFS-Dateisystem auf Ihrem lokalen Client	137
Schritt 4: Bereinigen Sie Ressourcen und schützen Sie Ihr AWS Konto	139
Optional: Datenverschlüsselung während der Übertragung	140
Tutorial: Mounten Sie ein Dateisystem von einer anderen VPC	143
Voraussetzungen	144
Schritt 1: Ermitteln Sie die ID der Availability Zone des Mount-Ziels	145
Schritt 2: Ermitteln Sie die Mount-Ziel-IP-Adresse	145
Schritt 3: Fügen Sie einen Hosteintrag für das Mount-Ziel hinzu	147
Schritt 4: Mounten Sie Ihr Dateisystem mit dem EFS-Mount-Helper	147
Schritt 5: Säubern Sie Ressourcen und schützen Sie Ihre AWS-Konto	149
Beheben von Mountingproblemen	150
Das Dateisystem-Mounting auf der Windows Instance schlägt fehl	150
Zugriff vom Server verweigert	150
Automatisches Mounting schlägt fehl und die Instance reagiert nicht	151
Mounting mehrerer Amazon-EFS-Dateisysteme in /etc/fstab schlägt fehl	151
Mounting-Befehl schlägt mit der Fehlermeldung „falscher fs-Typ“ fehl	152
Der Mounting-Befehl schlägt mit der Fehlermeldung „Inkorrekte Mounting-Option“ fehl	153
Mounting mit Zugangspunkt schlägt fehl	153
Das Mounting des Dateisystems schlägt sofort nach der Erstellung des Dateisystems fehl ..	154
Das Mounting des Dateisystems hängt und schlägt dann mit einem Timeout-Fehler fehl	154

Mounting eines Dateisystems mit NFS unter Verwendung eines DNS-Namens schlägt fehl .	155
Das Mounting des Dateisystems schlägt mit der Fehlermeldung „nfs reagiert nicht“ .	156
Der Lebenszyklusstatus des Mounting-Ziels hängt fest	156
Der Lebenszyklusstatus des Mount-Ziels zeigt einen Fehler	156
Mounting reagiert nicht	157
Gemounteter Client wird nicht mehr verbunden	157
Operationen auf einem neu gemounteten Dateisystem geben den Fehler „bad file handle“ zurück	158
Unmounten eines Dateisystems schlägt fehl	158
Datenübertragung	160
Benutzen AWS DataSync	160
Benutzen AWS Transfer Family	161
Voraussetzungen für die Verwendung AWS Transfer Family mit Amazon EFS	162
Konfiguration Ihres EFS-Dateisystems für AWS Transfer Family	162
Verwalten von Dateisystemen	168
Grundlegendes zum Dateisystemstatus	168
Verwalten der Mountingziele	169
Mounten Sie Ziele und Availability Zones	170
Erstellen von Bereitstellungszielen	173
Mount-Ziele werden gelöscht	179
Mount-Ziel-VPC ändern	181
Ändern der Mount-Ziel-Sicherheitsgruppen	181
Verwalten des Durchsatzes	183
Verwaltung des Speicherlebenszyklus	185
Dateisystemoperationen für die Lebenszyklusverwaltung	186
Konfiguration von Lebenszyklusrichtlinien	187
Überwachen	190
Tools zur Überwachung	191
Automatisierte Tools	191
Manuelle Überwachungstools	192
Messen eines Dateisystems	192
Messen von Objekten	193
Gemessene Größe eines Dateisystems	194
Mess-Durchsatz	196
Anzeigen der Größe der Speicherklasse	197
Metriken überwachen mit CloudWatch	198

CloudWatch Metriken	199
Zugriff auf CloudWatch Metriken	206
CloudWatch Metriken verwenden	207
Verwenden von metrischer Mathematik mit CloudWatch Metriken	209
Überwachung erfolgreicher und fehlgeschlagener Einhängerversuche	215
Erstellen von -Alarmen	217
Protokollierung von CloudTrail-API-Aufrufen mit	219
Amazon EFS-Informationen in CloudTrail	219
Erläuterungen der Amazon-EFS-Protokolldateieinträge	220
Amazon EFS-Protokolldateieinträge für encrypted-at-rest Dateisysteme	227
Verwenden von Abrechnungs- und Nutzungsberichten	229
Verwenden von -Kostenzuordnungs-Tags	229
Benutzerdefinierte Kostenzuordnungs-Markierungen	230
Abrechnungsberichte	231
Nutzungsberichte	232
Abrechnungs- und Nutzungsberichte verstehen	233
Nachverfolgung von Vorgängen in Ihren Nutzungsberichten	238
Leistungsspezifikationen	239
Zusammenfassung der Leistung	239
Speicherklassen	241
Leistungsmodi	242
Durchsatzmodi	242
Auswählen eines Durchsatzmodus	243
Elastischer Durchsatz	244
Bereitgestellter Durchsatz	244
Einschränkungen beim Umschalten des Durchsatzes und beim Ändern der bereitgestellten Menge	247
Tipps zur Leistung	247
Durchschnittliche I/O Größe	248
Optimierung von Workloads, die einen hohen Durchsatz und IOPS erfordern	248
Gleichzeitige Verbindungen	248
Anforderungsmodell	249
NFS-Client-Mount-Einstellungen	249
Optimierung der Leistung kleiner Dateien	250
Optimieren der Verzeichnisleistung	251
Optimierung der NFS-Größe von read_ahead_kb	251

Behebung von Leistungsproblemen	252
Ein EFS-Dateisystem kann nicht erstellt werden	253
Zugriff auf zulässige Dateien im NFS-Dateisystem verweigert	253
Fehler beim Zugriff auf die Amazon-EFS-Konsole	254
Amazon EC2-Instance hängt sich auf	254
Anwendung, die große Datenmengen schreibt, bleibt hängen	254
Schlechte Leistung beim Öffnen vieler Dateien gleichzeitig	255
Benutzerdefinierte NFS-Einstellungen verursachen Schreibverzögerungen	256
Die Erstellung von Sicherungen mit Oracle Recovery Manager ist langsam	256
Beheben von AML- und Kernel-Problemen	257
Eigentümerschaft kann nicht geändert werden	257
Aufgrund des Client-Bug wiederholt das Dateisystem Vorgänge immer wieder	258
Blockierter Client	258
Das Auflisten von Dateien in einem großen Verzeichnis dauert zu lange	258
Daten sichern und replizieren	260
Sichern von Dateisystemen	260
Wie AWS Backup funktioniert mit Amazon EFS	261
Erforderliche IAM-Berechtigungen	264
Backup-Leistung	264
Verwaltung automatischer Backups	265
Replizieren von Dateisystemen	266
Kosten	268
Replikationsleistung	268
Erforderliche IAM-Berechtigungen	268
Konfiguration der Replikation auf ein neues Dateisystem	269
Konfiguration der Replikation in ein vorhandenes Dateisystem	274
Kontenübergreifende Replikation AWS	280
Replikationsdetails anzeigen	287
Löschen von Replikationskonfigurationen	291
Das Replikat wird verwendet	293
Sicherung Ihrer Daten	295
Datenschutz	296
Datenverschlüsselung	297
Datenschutz zwischen Netzwerken	308
Identity and Access Management	309
Zielgruppe	310

Authentifizierung mit Identitäten	310
Verwalten des Zugriffs mit Richtlinien	312
Funktionsweise von Amazon Elastic File System mit IAM	313
Beispiele für identitätsbasierte Richtlinien	319
Beispiele für eine ressourcenbasierte Richtlinie	324
AWS verwaltete Richtlinien	327
Verwenden von Markierungen	333
Verwenden von servicegebundenen Rollen	336
Fehlerbehebung	341
Steuern des Datenzugriffs auf das Dateisystem	343
Standard-Dateisystemrichtlinie	344
EFS-Aktionen für Clients	344
EFS-Bedingungsschlüssel für Clients	345
Beispiele für Dateisystemrichtlinien	346
Compliance-Validierung	346
Ausfallsicherheit	346
Kontrolle des Netzwerkzugriffs	347
Verwenden von VPC-Sicherheitsgruppen	347
Arbeiten mit VPC-Endpunkten	350
Benutzer, Gruppen und Berechtigungen auf NFS-Ebene	352
Datei- und Verzeichnisberechtigungen	353
Beispiele für Anwendungsfälle und Berechtigungen für das EFS-Dateisystem	354
Benutzer- und Gruppen-ID-Berechtigungen für Dateien und Verzeichnisse innerhalb eines Dateisystems	355
Kein Root-Squashing	357
Zwischenspeichern von Berechtigungen	359
Ändern des Besitzes an Dateisystemobjekten	359
EFS-Zugangspunkte	360
Arbeiten mit Zugriffspunkten	360
Access Points funktionieren mit Mount-Zielen	360
Durchsetzen einer Benutzeridentität	361
Erzwingen eines Stammverzeichnisses	362
Verwenden von Zugangspunkten in IAM-Richtlinien	364
Sperrern des öffentlichen Zugriffs auf Dateisysteme	366
Sperrern des öffentlichen Zugriffs mit AWS Transfer Family	366
Die Bedeutung von „öffentlich“	367

Netzwerkisolierung	370
Kontingente	371
Amazon-EFS-Kontingente, die Sie erhöhen können	371
Beantragen einer Kontingenterhöhung	374
Amazon-EFS-Ressourcenkontingente, die Sie nicht ändern können	374
Kontingente für NFS-Clients	376
Kontingente für Amazon-EFS-Dateisysteme	377
Nicht unterstützte NFSv4 2.0- und 4.1-Funktionen	378
Weitere Überlegungen	379
Behebung von Fehlern bei Dateivorgängen im Zusammenhang mit Kontingenten	380
Suche nach offenen Dateien und Dateisperren	380
Der Befehl schlägt mit dem Fehler „Disk quota exceeded“ fehl	381
Befehl schlägt mit „E/A-Fehler“ fehl	381
Befehl schlägt mit der Fehlermeldung „Dateiname ist zu lang“ fehl	382
Befehl schlägt fehl mit dem Fehler „Datei nicht gefunden“	382
Befehl schlägt mit der Fehlermeldung „Zu viele Links“ fehl	383
Befehl schlägt mit der Fehlermeldung „Datei zu groß“ fehl.	383
Amazon-EFS-API	384
API-Endpunkte	384
IPv4 Endpunkte	385
Dual-Stack-Endpunkte (IPv4 und IPv6)	385
Angaben von Endpunkten	386
API-Version	386
Verwandte Themen	386
Arbeiten mit der Abfrage-API-Anforderungsrate für Amazon EFS	387
Abrufen	387
Wiederholversuche oder Stapelverarbeitung	387
Berechnen des Energiesparintervalls	387
Aktionen	388
CreateAccessPoint	390
CreateFileSystem	400
CreateMountTarget	418
CreateReplicationConfiguration	437
CreateTags	447
DeleteAccessPoint	451
DeleteFileSystem	454

DeleteFileSystemPolicy	459
DeleteMountTarget	463
DeleteReplicationConfiguration	468
DeleteTags	472
DescribeAccessPoints	476
DescribeAccountPreferences	482
DescribeBackupPolicy	486
DescribeFileSystemPolicy	491
DescribeFileSystems	497
DescribeLifecycleConfiguration	504
DescribeMountTargets	509
DescribeMountTargetSecurityGroups	517
DescribeReplicationConfigurations	522
DescribeTags	528
ListTagsForResource	534
ModifyMountTargetSecurityGroups	539
PutAccountPreferences	545
PutBackupPolicy	549
PutFileSystemPolicy	554
PutLifecycleConfiguration	561
TagResource	571
UntagResource	576
UpdateFileSystem	580
UpdateFileSystemProtection	590
Datentypen	596
AccessPointDescription	598
BackupPolicy	601
CreationInfo	602
Destination	604
DestinationToCreate	607
FileSystemDescription	611
FileSystemProtectionDescription	616
FileSystemSize	617
LifecyclePolicy	619
MountTargetDescription	621
PosixUser	625

ReplicationConfigurationDescription	627
ResourceIdPreference	629
RootDirectory	630
Tag	632
Dokumentverlauf	633
.....	dclxi

Was ist Amazon Elastic File System?

Amazon Elastic File System (Amazon EFS) bietet vollständig elastischen Serverless-Dateispeicher, sodass Sie Dateidaten gemeinsam nutzen können, ohne Speicherkapazität und Leistung bereitstellen oder verwalten zu müssen. Amazon EFS ist so konzipiert, dass es bei Bedarf auf Petabytes skaliert werden kann, ohne Anwendungen zu unterbrechen. Es wächst und schrumpft automatisch, wenn Sie Dateien hinzufügen oder entfernen. Da Amazon EFS über eine einfache Webservice-Schnittstelle verfügt, können Sie Dateisysteme schnell und einfach erstellen und konfigurieren. Der Service übernimmt die Verwaltung der gesamten Dateispeicherinfrastruktur für Sie. Auf diese Weise kann der Aufwand der Bereitstellung, des Patchings und der Wartung komplexer Dateisystemkonfigurationen vermieden werden.

Amazon EFS unterstützt das Network File System-Protokoll Version 4.1 (NFSv4.1 und NFSv4 .0), sodass die Anwendungen und Tools, die Sie heute verwenden, problemlos mit Amazon EFS zusammenarbeiten. Amazon EFS ist für die meisten Arten von Amazon Web Services Services-Recheninstanzen zugänglich EC2, einschließlich Amazon, Amazon ECS AWS Lambda, Amazon EKS und AWS Fargate.

Der Service ist hochgradig skalierbar, hochverfügbar und äußerst langlebig. Amazon EFS bietet die folgenden Dateisystemtypen, um Ihren Anforderungen an Verfügbarkeit und Haltbarkeit gerecht zu werden:

- **Regional (empfohlen)** — Regionale Dateisysteme (empfohlen) speichern Daten redundant in mehreren geografisch getrennten Availability Zones innerhalb derselben AWS-Region. Das Speichern von Daten in mehreren Availability Zones gewährleistet eine kontinuierliche Verfügbarkeit der Daten, selbst wenn eine oder mehrere Availability Zones in einer nicht verfügbar AWS-Region sind.
- **Eine Zone** — Dateisysteme mit einer Zone speichern Daten innerhalb einer einzigen Availability Zone. Das Speichern von Daten in einer einzigen Availability Zone gewährleistet eine kontinuierliche Verfügbarkeit der Daten. Im unwahrscheinlichen Fall eines Verlusts oder einer Beschädigung der gesamten Availability Zone oder eines Teils davon können jedoch Daten, die in diesen Dateisystemen gespeichert sind, verloren gehen.

Weitere Informationen über Dateisystemtypen finden Sie unter [EFS-Dateisystemtypen](#).

Amazon EFS bietet den Durchsatz, die IOPS und die niedrige Latenz, die für eine breite Palette von Workloads erforderlich sind. EFS-Dateisysteme können bis in den Petabyte-Bereich wachsen, bieten

einen hohen Durchsatz und ermöglichen einen massiv parallelen Zugriff von Compute-Instances auf Ihre Daten. Für die meisten Workloads empfehlen wir die Verwendung der Standardmodi, d. h. den allgemeinen Leistungsmodus und den Elastischen Durchsatzmodus.

- **Allgemeiner Zweck** — Der Allzweck-Leistungsmodus ist ideal für latenzempfindliche Anwendungen wie Web-Server-Umgebungen, Content-Management-Systeme, Home-Verzeichnisse und allgemeine Dateibereitstellung.
- **Elastisch** — Der Elastic Throughput-Modus ist so konzipiert, dass er die Durchsatzleistung automatisch nach oben oder unten skaliert, um den Anforderungen Ihrer Workload-Aktivität gerecht zu werden.

Weitere Informationen zu EFS-Leistungs- und Durchsatzmodi finden Sie unter [Leistungsspezifikationen von Amazon EFS](#).

Amazon EFS bietet file-system-access Semantik, wie z. B. starke Datenkonsistenz und Dateisperren. Weitere Informationen finden Sie unter [Datenkonsistenz in Amazon EFS](#). Amazon EFS unterstützt auch die Steuerung des Zugriffs auf Ihre Dateisysteme durch POSIX-Berechtigungen (Portable Operating System Interface). Weitere Informationen finden Sie unter [Sicherung Ihrer Daten in Amazon EFS](#).

Amazon EFS unterstützt Authentifizierungs-, Autorisierungs- und Verschlüsselungsfunktionen, damit Sie Ihre Sicherheits- und Compliance-Anforderungen erfüllen können. Amazon EFS unterstützt zwei Formen der Verschlüsselung für Dateisysteme: Verschlüsselung bei der Übertragung und Verschlüsselung im Ruhezustand. Sie können die Verschlüsselung im Ruhezustand aktivieren, wenn Sie ein EFS-Dateisystem erstellen. Wenn Sie dies tun, werden alle Ihre Daten und Metadaten verschlüsselt. Sie können die Verschlüsselung während der Übertragung aktivieren, wenn Sie das Dateisystem mounten. Der NFS-Client-Zugriff auf Amazon EFS wird sowohl durch AWS Identity and Access Management (IAM-) Richtlinien als auch durch Netzwerksicherheitsrichtlinien wie Sicherheitsgruppen gesteuert. Weitere Informationen finden Sie unter [Datenverschlüsselung in Amazon EFS](#), [Identitäts- und Zugriffsmanagement für Amazon EFS](#) und [Steuern des Netzwerkzugriffs auf EFS-Dateisysteme für NFS-Clients](#).

 Note

Die Verwendung von Amazon EFS mit Microsoft Windows-basierten EC2 Amazon-Instances wird nicht unterstützt.

Sind Sie ein Erstnutzer von Amazon EFS?

Wenn Sie Amazon EFS zum ersten Mal verwenden, empfehlen wir Ihnen, die folgenden Abschnitte der Reihe nach zu lesen:

1. Eine Übersicht über die Produkte und Preise von Amazon EFS finden Sie unter [Amazon EFS](#).
2. Einen technischen Überblick über Amazon EFS finden Sie unter [So funktioniert Amazon EFS](#).
3. Probieren Sie die Übung aus [Erste Schritte](#).

Wenn Sie mehr über Amazon EFS erfahren möchten, finden Sie in den folgenden Themen nähere Informationen zu diesem Service:

- [EFS-Ressourcen erstellen und verwalten](#)
- [Verwaltung von EFS-Dateisystemen](#)
- [Amazon-EFS-API](#)

So funktioniert Amazon EFS

Amazon Elastic File System (EFS) bietet ein einfaches, serverloses, set-and-forget elastisches Dateisystem. Mit Amazon EFS können Sie ein Dateisystem erstellen, das Dateisystem in eine Amazon EC2-Instance mounten und dann Daten auf Ihrem Dateisystem lesen und von dort schreiben. Sie können ein EFS-Dateisystem in Ihrer Virtual Private Cloud (VPC) über das Network File System der Versionen 4.0 und 4.1 (NFSv4) mounten. Wir empfehlen die Verwendung eines Linux NFSv4 1.1-Clients der aktuellen Generation, wie sie in den neuesten Versionen von Amazon Linux, Amazon Linux 2, Red Hat, Ubuntu und macOS Big Sur zu finden sind AMLs, zusammen mit dem EFS-Mount-Helper. Detaillierte Anweisungen finden Sie unter [Den Amazon EFS-Client installieren](#).

Eine Liste der Amazon EC2 Linux- und macOS Amazon Machine Images (AMIs), die dieses Protokoll unterstützen, finden Sie unter [NFS-Support](#). Für einige AMIs müssen Sie einen NFS-Client installieren, um Ihr Dateisystem auf Ihrer Amazon EC2-Instance zu mounten. Detaillierte Anweisungen finden Sie unter [Installieren des NFS-Clients](#).

Sie können gleichzeitig von mehreren NFS-Clients aus auf Ihr EFS-Dateisystem zugreifen, sodass Anwendungen, die über eine einzelne Verbindung hinaus skalieren, auf ein Dateisystem zugreifen können. Amazon EC2 und andere AWS Recheninstanzen, die in mehreren Availability Zones

innerhalb derselben ausgeführt werden, AWS-Region können auf das Dateisystem zugreifen, sodass viele Benutzer auf eine gemeinsame Datenquelle zugreifen und diese gemeinsam nutzen können.

Eine Liste der Orte AWS-Regionen , an denen Sie ein EFS-Dateisystem erstellen können, finden Sie unter [Allgemeine Amazon Web Services-Referenz](#).

Um auf Ihr EFS-Dateisystem in einer VPC zuzugreifen, erstellen Sie ein oder mehrere Mount-Ziele in der VPC. Ein Mount-Ziel stellt eine IP-Adresse für einen NFSv4-Endpunkt bereit, an dem Sie ein EFS-Dateisystem mounten können. Sie mounten Ihr Dateisystem mithilfe des zugehörigen Domain Name Service(DNS)-Namens, der in die IP-Adresse des EFS-Mounting-Ziels in derselben Availability Zone wie Ihre EC2-Instance aufgelöst wird. Sie können in jeder Availability Zone ein Mounting-Ziel in einer AWS-Region. Falls die Availability Zone in Ihrer VPC über mehrere Subnetze verfügt, erstellen Sie in einem der Subnetze ein Mounting-Ziel. Anschließend können alle EC2-Instances in dieser Availability Zone dieses Mounting-Ziel gemeinsam verwenden.

Note

Ein EFS-Dateisystem kann Mount-Ziele jeweils nur in einer VPC haben.

Die Mountingziele selbst sind hochverfügbar. Wenn Sie Hochverfügbarkeit und Failover zu anderen Availability Zones planen, sollten Sie bedenken, dass die IP-Adressen und DNS für Ihre Mounting-Ziele in jeder Availability Zone zwar statisch sind, aber es sich um redundante Komponenten handelt, die von mehreren Ressourcen unterstützt werden. Weitere Informationen zu Mount-Zielen finden Sie unter [Verwalten der Mountingziele](#).

Nachdem Sie das Dateisystem mithilfe seines DNS-Namens bereitgestellt haben, verwenden Sie es wie jedes andere Dateisystem. Für Informationen zu Berechtigungen auf NFS-Ebene und dazugehörige Überlegungen vgl. [Benutzer, Gruppen und Berechtigungen auf NFS-Ebene \(Network File System\)](#).

Sie können Ihre EFS-Dateisysteme auf Ihren lokalen Rechenzentrumsservern mounten, wenn Sie mit oder mit AWS Direct Connect Ihrer Amazon VPC verbunden sind. Site-to-Site VPN Sie können Ihre EFS-Dateisysteme auf lokalen Servern bereitstellen, um Datensätze zu EFS zu migrieren, Cloud-Bursting-Szenarien zu aktivieren oder Ihre lokalen Daten in Amazon EFS zu sichern.

Im Folgenden finden Sie eine Beschreibung, wie Amazon EFS mit anderen Services zusammenarbeitet.

Themen

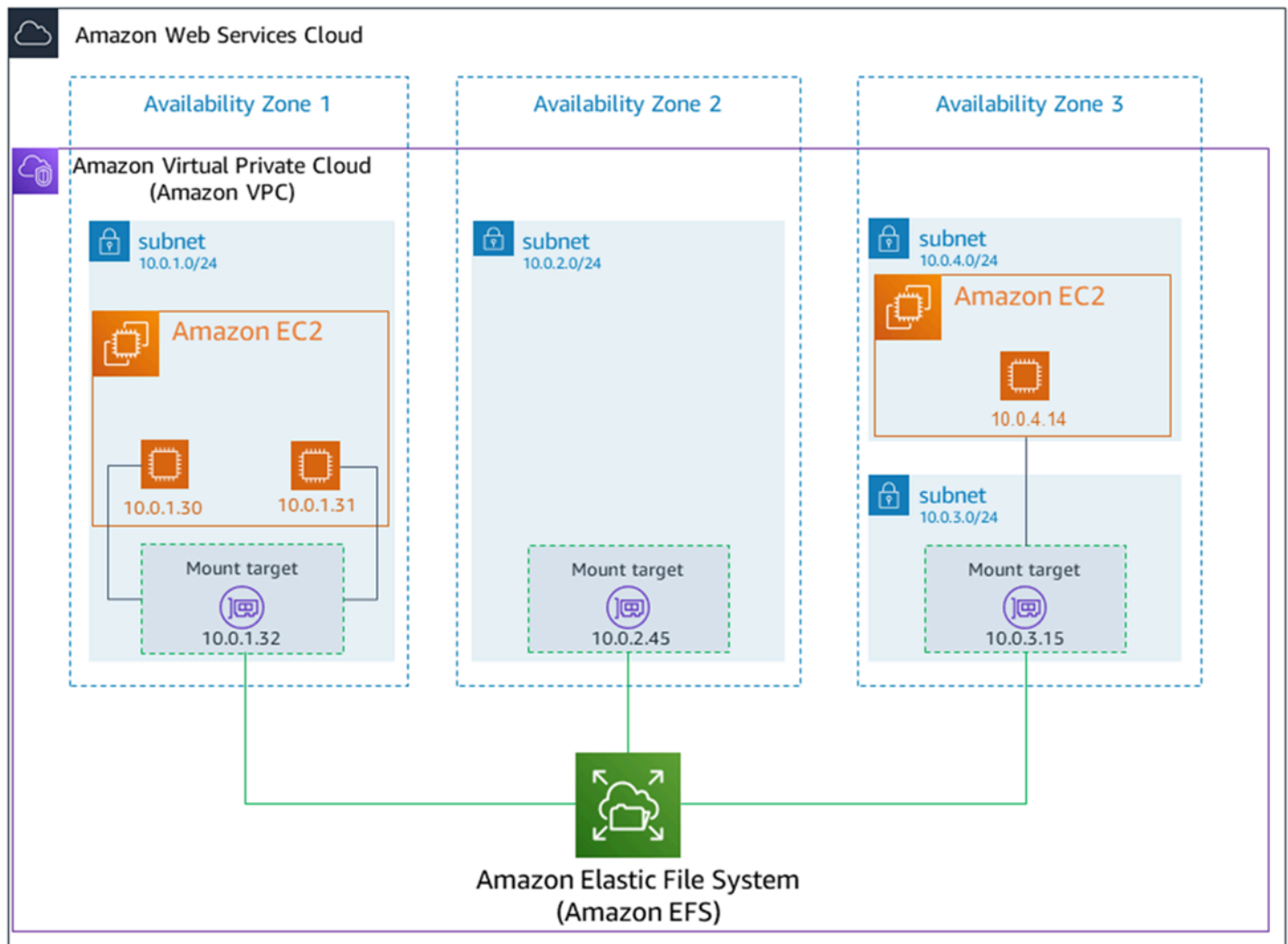
- [So funktioniert Amazon EFS mit Amazon EC2](#)
- [So funktioniert Amazon EFS mit AWS Direct Connect und AWS verwaltetem VPN](#)
- [So funktioniert Amazon EFS mit AWS Backup](#)

So funktioniert Amazon EFS mit Amazon EC2

In diesem Abschnitt wird erklärt, wie die Dateisysteme Amazon EFS Regional und One Zone in EC2-Instances in einer Amazon VPC gemountet werden.

Regionale EFS-Dateisysteme

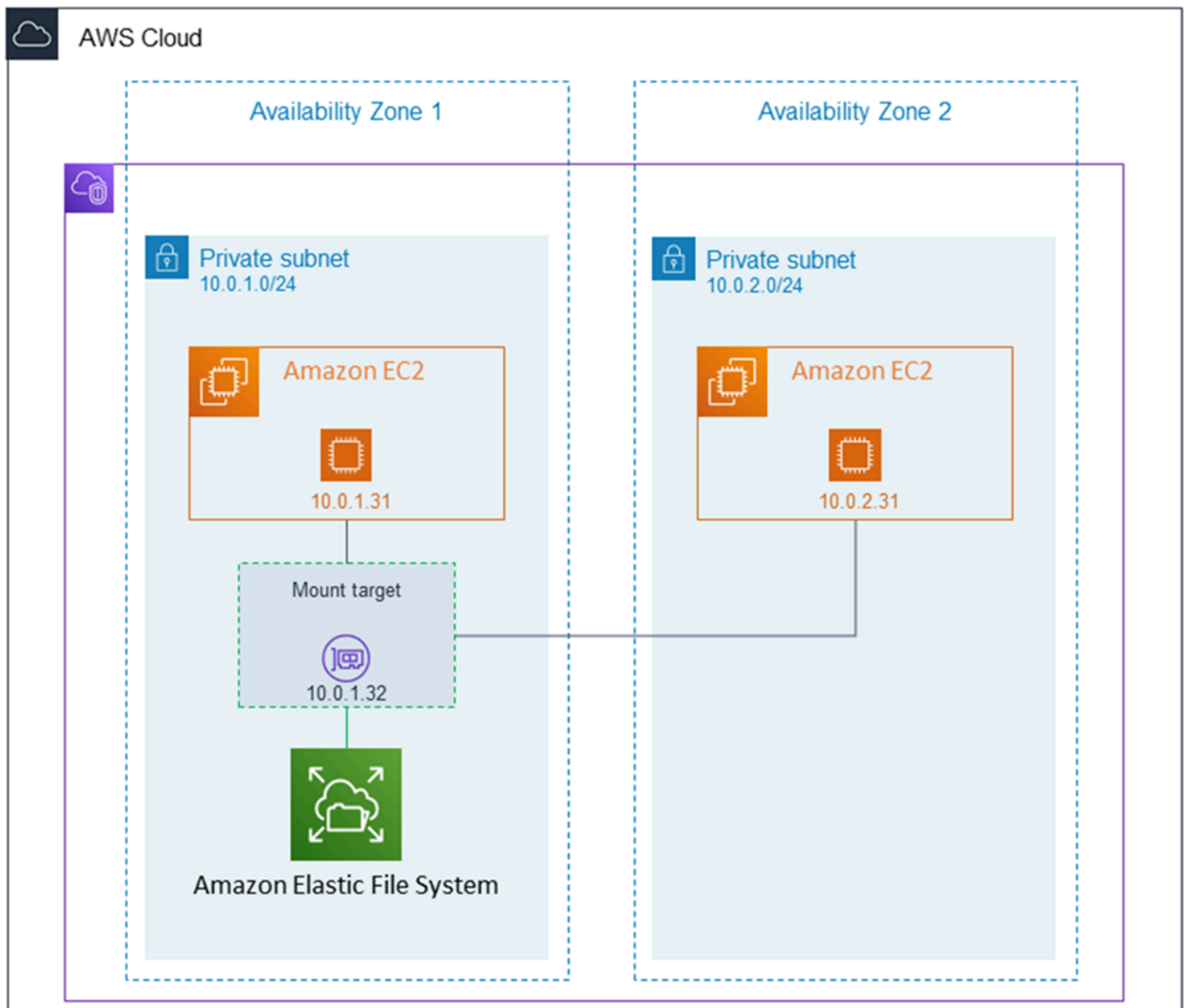
Die folgende Abbildung zeigt mehrere EC2-Instances, die auf ein Amazon-EFS-Dateisystem zugreifen, das für mehrere Availability Zones in einer AWS-Region konfiguriert wurde.



In dieser Abbildung hat die Virtual Private Cloud (VPC) drei Availability Zones. Da das Dateisystem regional ist, wurde in jeder Availability Zone ein Mounting-Ziele erstellt. Aus Leistungs- und Kostengründen empfehlen wir, dass Sie auf das Dateisystem von einem Mounting-Ziel innerhalb derselben Availability Zone zugreifen. Eine der Availability Zones verfügt über zwei Subnetze. Ein Mounting-Ziel wird jedoch nur in einem der Subnetze erstellt. Weitere Informationen finden Sie unter [Mounten von EFS-Dateisystemen mit dem EFS-Mount-Helper](#).

EFS-Dateisysteme für eine Zone

Die folgende Abbildung zeigt mehrere EC2-Instances, die auf ein One-Zone-Dateisystem aus verschiedenen Availability Zones in einer einzigen AWS-Region zugreifen.



In dieser Abbildung hat die VPC zwei Availability Zones mit jeweils einem Subnetz. Da der Dateisystemtyp One Zone ist, kann er nur ein einziges Mounting-Ziel haben. Um die Leistung und die Kosten zu verbessern, empfehlen wir Ihnen, auf das Dateisystem von einem Mounting-Ziel aus zuzugreifen, das sich in derselben Availability Zone befindet wie die EC2-Instance, in die Sie es mounten.

In diesem Beispiel zahlt die EC2-Instance in der Availability Zone us-west-2c EC2-Datenzugriffsgebühren für den Zugriff auf ein Mounting-Ziel in einer anderen Availability Zone. Weitere Informationen finden Sie unter [Mounting von One-Zone-Dateisystemen](#).

So funktioniert Amazon EFS mit AWS Direct Connect und AWS verwaltetem VPN

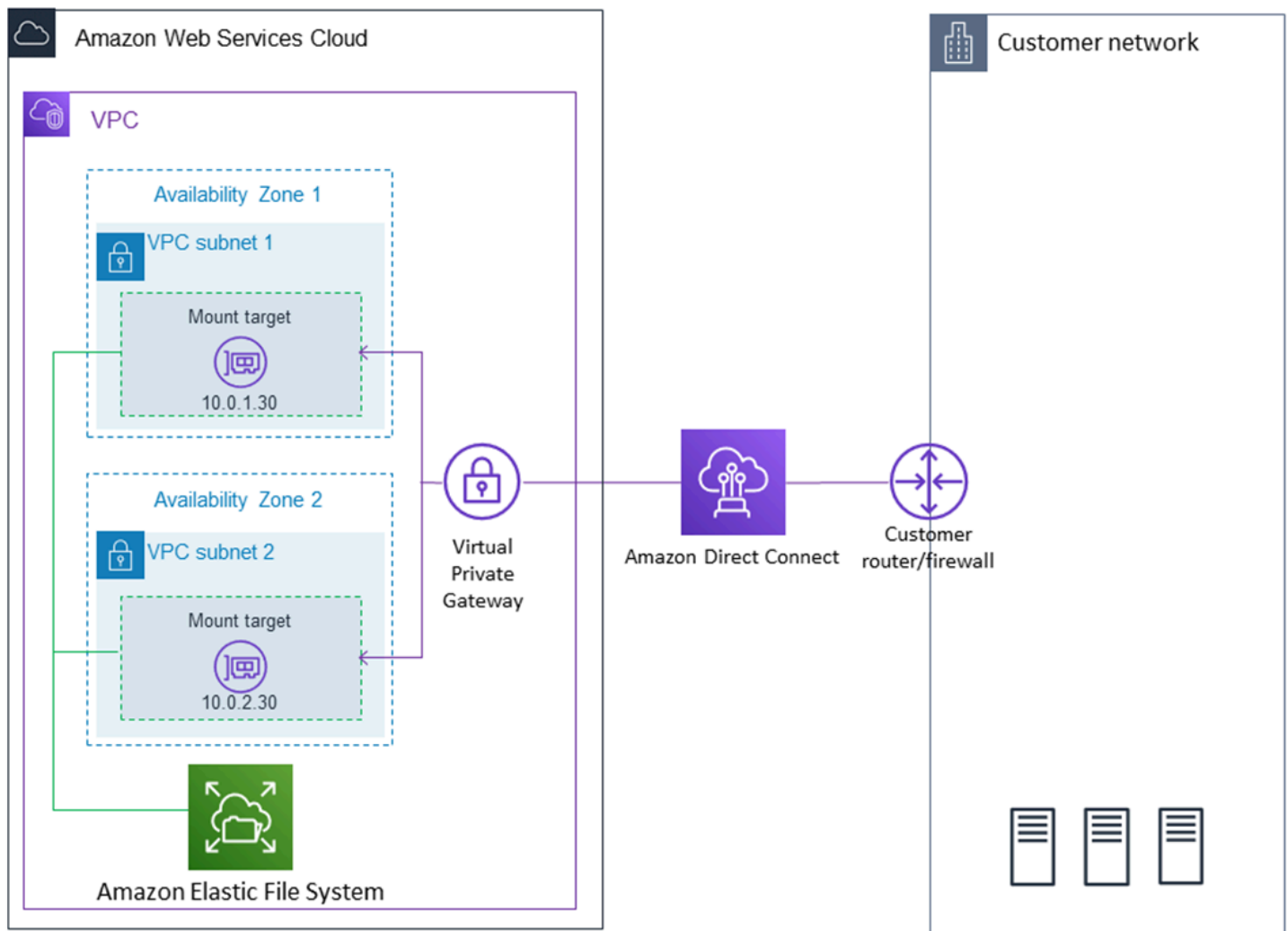
Durch die Verwendung eines auf einem lokalen Server bereitgestellten Amazon EFS-Dateisystems können Sie lokale Daten in das in einem Amazon EFS-Dateisystem AWS Cloud gehostete migrieren. Sie können außerdem die Vorteile des Burstings nutzen. Mit anderen Worten: Sie können Daten von Ihren On-Premises-Servern in Amazon EFS verschieben und sie auf einer Flotte von Amazon EC2-Instances in Ihrer Amazon VPC analysieren. Sie können dann die Ergebnisse dauerhaft in Ihrem Dateisystem speichern oder zurück auf Ihren On-Premises-Server verschieben.

Beachten Sie die folgenden Überlegungen, wenn Sie Amazon EFS mit einem On-Premises-Server verwenden:

- Ihr On-Premises-Server muss über ein auf Linux basierendes Betriebssystem verfügen. Wir empfehlen die Linux-Kernel-Version 4.0 oder höher.
- Der Einfachheit halber empfehlen wir, ein Amazon-EFS-Dateisystem auf einem On-Premises-Server zu mounten, indem Sie eine IP-Adresse des Mounting-Ziels anstelle eines DNS-Namens verwenden.

Es fallen keine zusätzlichen Kosten für den On-Premises-Zugriff auf Ihre Amazon-EFS-Dateisysteme an. Die Direct Connect Verbindung zu Ihrer Amazon VPC wird Ihnen in Rechnung gestellt. Weitere Informationen finden Sie unter [Direct Connect Preise](#).

Die folgende Abbildung zeigt ein Beispiel für den Zugriff auf ein Amazon-EFS-Dateisystem von den On-Premises-Servern aus (auf den On-Premises-Servern sind die Dateisysteme gemountet).



Sie können jedes Mount-Ziel in Ihrer VPC verwenden, wenn Sie das Subnetz dieses Mount-Ziels über eine Direct Connect Verbindung zwischen Ihrem lokalen Server und der VPC erreichen können. Um von einem On-Premises-Server auf Amazon EFS zuzugreifen, fügen Sie Ihrer Sicherheitsgruppe für das Mounting-Ziel eine Regel hinzu, die eingehenden Datenverkehr zum NFS-Port (2049) von Ihrem On-Premises-Server zulässt. Weitere Informationen, einschließlich detaillierter Verfahren, finden Sie unter [Voraussetzungen](#).

So funktioniert Amazon EFS mit AWS Backup

Für eine umfassende Backup-Implementierung für Ihre Dateisysteme können Sie Amazon EFS mit verwenden AWS Backup. AWS Backup ist ein vollständig verwalteter Backup-Service, der es einfach macht, Datensicherungen zwischen AWS Diensten in der Cloud und vor Ort zu zentralisieren und zu automatisieren. Mit ihm AWS Backup können Sie Backup-Richtlinien zentral konfigurieren und die Backup-Aktivitäten für Ihre AWS Ressourcen überwachen. Amazon EFS priorisiert

Dateisystemoperationen immer vor Sicherungsvorgängen. Weitere Informationen zum Sichern von EFS-Dateisystemen mithilfe von AWS Backup finden Sie unter [Sicherung von EFS-Dateisystemen](#).

Funktionen von Amazon EFS

Im Folgenden sind die Funktionen von Amazon EFS aufgeführt.

Themen

- [Authentifizierung und Zugriffskontrolle](#)
- [Datenkonsistenz in Amazon EFS](#)
- [Verfügbarkeit und Haltbarkeit von EFS-Dateisystemen](#)
- [Replikation](#)

Authentifizierung und Zugriffskontrolle

Sie benötigen gültige Anmeldeinformationen, um die Amazon EFS-Managementkonsole zu verwenden und Amazon EFS-API-Anfragen zu stellen, z. B. ein Dateisystem zu erstellen. Darüber hinaus benötigen Sie auch Berechtigungen, um andere EFS und AWS Ressourcen zu erstellen oder darauf zuzugreifen.

Benutzern und Rollen, die Sie in AWS Identity and Access Management (IAM) erstellen, müssen Berechtigungen zum Erstellen von oder Zugreifen auf Ressourcen erteilt werden. Weitere Informationen zu Berechtigungen finden Sie unter [Identitäts- und Zugriffsmanagement für Amazon EFS](#).

Die IAM-Autorisierung für NFS-Clients ist eine zusätzliche Sicherheitsoption für Amazon EFS, die IAM nutzt, um die Zugriffsverwaltung für Network File System (NFS)-Clients im großen Maßstab zu vereinfachen. Mit der IAM-Autorisierung für NFS-Clients können Sie IAM verwenden, um den Zugriff auf ein EFS-Dateisystem auf inhärent skalierbare Weise zu verwalten. Die IAM-Autorisierung für NFS-Clients ist auch für Cloud-Umgebungen optimiert. Weitere Informationen zur Verwendung der IAM-Autorisierung für NFS-Clients finden Sie unter [Verwendung von IAM zur Steuerung des Zugriffs auf Dateisysteme](#).

Datenkonsistenz in Amazon EFS

Amazon EFS bietet die close-to-open Konsistenzsemantik, die Anwendungen von NFS erwarten.

In Amazon EFS werden Schreibvorgänge für regionale Dateisysteme in diesen Situationen dauerhaft über Availability Zones hinweg gespeichert:

- Eine Anwendung führt einen synchronen Schreibvorgang (z. B. unter Verwendung des Linux-Befehls `open` mit dem Flag `O_DIRECT` oder des Linux-Befehls `fsync`) aus.
- Eine Anwendung schließt eine Datei.

Je nach Zugriffsmuster kann Amazon EFS stärkere Konsistenzgarantien als close-to-open Semantik bieten. Anwendungen, die synchrone Datenzugriffe durchführen und Schreibvorgänge ohne Anhängen ausführen, bieten read-after-write Konsistenz beim Datenzugriff.

Sperren von Dateien

NFS-Client-Anwendungen können NFS Version 4-Dateisperren (einschließlich Byte-Range-Sperren) für Lese- und Schreibvorgänge auf Amazon-EFS-Dateien verwenden.

Beachten Sie die folgenden Hinweise zum Sperren von Dateien durch Amazon EFS:

- Amazon EFS unterstützt nur das Sperren von Empfehlungen, und bei read/write Vorgängen wird vor der Ausführung nicht nach widersprüchlichen Sperren gesucht. Um beispielsweise Probleme mit der Dateisynchronisierung bei atomaren Vorgängen zu vermeiden, muss Ihre Anwendung die NFS-Semantik (z. B. close-to-open Konsistenz) berücksichtigen.
- Jede einzelne Datei kann in allen verbundenen Instances bis zu 512 Locks und auf die Datei zugreifende Benutzer verfügen.

Verfügbarkeit und Haltbarkeit von EFS-Dateisystemen

In diesem Abschnitt werden die Dateisystemtypen und Speicherklassenoptionen für Amazon Elastic File System (Amazon EFS)-Dateisysteme beschrieben.

EFS-Dateisystemtypen

Amazon EFS bietet regionale und One-Zone-Dateisystemtypen.

- **Regional** — Regionale Dateisysteme (empfohlen) speichern Daten redundant in mehreren geografisch getrennten Availability Zones innerhalb derselben AWS-Region. Das Speichern von Daten in mehreren Availability Zones gewährleistet eine kontinuierliche Verfügbarkeit der Daten, selbst wenn eine oder mehrere Availability Zones in einer nicht verfügbar AWS-Region sind.

- Eine Zone — Dateisysteme mit einer Zone speichern Daten innerhalb einer einzigen Availability Zone. Das Speichern von Daten in einer einzigen Availability Zone gewährleistet eine kontinuierliche Verfügbarkeit der Daten. Im unwahrscheinlichen Fall eines Verlusts oder einer Beschädigung der gesamten Availability Zone oder eines Teils davon können jedoch Daten, die in diesen Dateisystemen gespeichert sind, verloren gehen.

In dem unwahrscheinlichen Fall, dass eine AWS Availability Zone ganz oder teilweise verloren geht oder beschädigt wird, können Daten in einer Speicherkategorie von One Zone verloren gehen. Beispielsweise können Ereignisse wie Feuer- und Wasserschäden zu Datenverlust führen. Abgesehen von diesen Arten von Ereignissen sind unsere One Zone-Speicherkategorien ähnlich konzipiert wie unsere regionalen Speicherkategorien, sodass Objekte vor den Ausfällen unabhängiger Datenträger oder Hosts und Racks geschützt sind. Jede Kategorie ist auf eine Datenzuverlässigkeit von 99,999999999 % ausgelegt.

Für zusätzlichen Datenschutz sichert Amazon EFS automatisch One Zone-Dateisysteme mit AWS Backup. Sie können Dateisystem-Backups in jeder betriebsbereiten Availability Zone innerhalb einer AWS-Region oder in einer anderen wiederherstellen AWS-Region. EFS-Dateisystem-Backups, die mit Hilfe erstellt und verwaltet werden AWS Backup werden, werden in drei Availability Zones repliziert und sind auf Beständigkeit ausgelegt. Weitere Informationen finden Sie unter [Resilienz in AWS Backup](#).

Note

Dateisysteme mit einer Zone sind nur für bestimmte Availability Zones verfügbar. Eine Tabelle mit einer Liste der Availability Zones, in denen Sie One Zone-Dateisysteme verwenden können, finden Sie unter [Unterstützte Availability Zones für One Zone-Dateisysteme](#).

In der folgenden Tabelle werden die Dateisystemtypen verglichen, einschließlich ihrer Verfügbarkeit, Zuverlässigkeit und anderer Faktoren.

Dateisystemtyp	Konzipiert für	Zuverlässigkeit (Auslegung)	Verfügbarkeit	Verfügbarkeitszonen	Weitere Überlegungen
Regional	Daten, die ein Höchstmaß an	99,999999999 % (11x9)	99,99 %	>=3	Keine

Dateisystemtyp	Konzipiert für	Zuverlässigkeit (Auslegung)	Verfügbarkeit	Verfügbarkeitszonen	Weitere Überlegungen
Zuverlässigkeit und Verfügbarkeit erfordern.					
One Zone	Daten, für die keine höchste Zuverlässigkeit und Verfügbarkeit erforderlich ist.	99,999999 999 % (11x9)	99,99 %	1	Nicht widerstandsfähig gegen den Verlust der Availability Zone

Unterstützte Availability Zones für One Zone-Dateisysteme

Dateisysteme mit einer Zone sind nur für bestimmte Availability Zones verfügbar. In der folgenden Tabelle sind die AWS-Region und die AZ IDs für jede Availability Zone aufgeführt, in der Sie One Zone-Dateisysteme verwenden können. Informationen zur Zuordnung von AZ IDs zu Availability Zones in Ihrem Konto finden Sie unter [Availability Zone IDs for your AWS Resources](#) im AWS Resource Access Manager Manager-Benutzerhandbuch.

Availability Zones, die One Zone-Dateisysteme unterstützen

AWS-Region Name	AWS-Region Code	Unterstützt AZ IDs
USA Ost (Ohio)	us-east-2	use2-az1, use2-az2, use2-az3
USA Ost (Nord-Virginia)	us-east-1	use1-az1, use1-az2, use1-az4, use1-az5, use1-az6
USA West (Nordkalifornien)	us-west-1	usw1-az1, usw1-az3
USA West (Oregon)	us-west-2	usw2-az1, usw2-az2, usw2-az3, usw2-az4
Afrika (Kapstadt)	af-south-1	afs1-az1, afs1-az2, afs1-az3

AWS-Region Name	AWS-Region Code	Unterstützt AZ IDs
Asien-Pazifik (Hongkong)	ap-east-1	ape-1-az1, ape-1-az2, ape-1-az3
Asien-Pazifik (Mumbai)	ap-south-1	aps1-az1, aps1-az2, aps1-az3
Asia Pacific (Osaka)	ap-northeast-3	apne-3-az1, apne-3-az2, apne-3-az3
Asien-Pazifik (Seoul)	ap-northeast-2	apne2-az1, apne2-az2, apne2-az3
Asien-Pazifik (Singapur)	ap-southeast-1	apse1-az1, apse1-az2
Asien-Pazifik (Sydney)	ap-southeast-2	apse2-az1, apse2-az2, apse2-az3
Asien-Pazifik (Tokio)	ap-northeast-1	apne1-az1, apne1-az4
Kanada (Zentral)	ca-central-1	cac1-az1, cac1-az2
China (Beijing)	cn-north-1	cnn1-az1, cnn1-az2
China (Ningxia)	cn-northwest-1	cnnw1-az1, cnnw1-az2, cnnw1-az3
Europa (Frankfurt)	eu-central-1	euc1-az1, euc1-az2, euc1-az3
Europa (Irland)	eu-west-1	euw1-az1, euw1-az2, euw1-az3
Europa (London)	eu-west-2	euw2-az1, euw2-az2
Europa (Milan)	eu-south-1	eus1-az1, eus1-az2, eus1-az3
Europa (Paris)	eu-west-3	euw3-az1, euw3-az3
Europa (Stockholm)	eu-north-1	eun1-az1, eun1-az2, eun1-az3

AWS-Region Name	AWS-Region Code	Unterstützt AZ IDs
Naher Osten (Bahrain)	me-south-1	mes1-az1, mes1-az2, mes1-az3
Südamerika (São Paulo)	sa-east-1	sae1-az1, sae1-az2, sae1-az3
AWS GovCloud (US-Ost)	us-gov-east-1	usge1-az1, usge1-az2, usge1-az3
AWS GovCloud (US-West)	us-gov-west-1	usgw1-az1, usgw1-az2, usgw1-az3

EFS-Speicherklassen

Amazon EFS bietet je nach Anwendungsfall verschiedene Speicherklassen, die für die effektivste Speicherung konzipiert sind.

- **EFS-Standard** – Die EFS-Standard-Speicherklasse verwendet Solid-State-Drive-Speicher (SSD), um die geringste Latenz für häufig aufgerufene Dateien zu gewährleisten. Neue Dateisystemdaten werden zuerst in die EFS-Standardspeicherklasse geschrieben und können dann mithilfe des Lebenszyklusmanagements den Speicherklassen EFS Infrequent Access und EFS Archive zugeordnet werden.
- **EFS Infrequent Access (IA)** — Eine kostenoptimierte Speicherklasse für Daten, auf die nur wenige Male pro Quartal zugegriffen wird.
- **EFS Archive** – Eine kostenoptimierte Speicherklasse für Daten, auf die wenige Male pro Jahr zugegriffen wird.

Die EFS-Archive-Speicherklasse wird auf EFS-Dateisystemen mit elastischem Durchsatz unterstützt. Sie können den Durchsatz Ihres Dateisystems nicht auf „Bursting“ oder „Bereitgestellt“ aktualisieren, sobald das Dateisystem Daten in der Archive-Speicherklasse enthält.

Vergleich der Speicherklassen

Die folgende Tabelle vergleicht die verschiedenen Speicherklassen. Weitere Informationen zur Leistung der einzelnen Speicherklassen finden Sie unter [Leistungsspezifikationen von Amazon EFS](#).

Speicherklasse	Konzipiert für	Latenz beim Lesen beim ersten Byte	Haltbarkeit (konzipiert für) ¹	Verfügbarkeit SLA	Verfügbarkeitszone n	Minimale Abrechnungsgebühr pro Datei ²	Minde
EFS Standard	Aktive Daten, die eine schnelle Latenzzeit von unter einer Millisekunde erfordern	Eine Sekunde unter einer Millisekunde		99,99% (Regional) 99,9% (Eine Zone)	=>3 (Regional)	Nicht zutreffend	Nicht zutreffend
EFS Infrequent Access	Inaktive Daten, auf die nur wenige Male pro Quartal zugegriffen wird.	Zehn Millisekunden	99,999999999 % (11 9)		1 (Eine Zone)	128 KiB	Nicht zutreffend
EFS Archive	Inaktive Daten, auf die mehrmals pro Jahr oder weniger zugegriffen wird	Zehn Millisekunden		99,9% (Regional)	=>3 (Regional)	128 KiB	90 Tage

¹ Da One Zone-Dateisysteme Daten in einer einzigen AWS Availability Zone speichern, können Daten, die in diesen Dateisystemtypen gespeichert sind, im Falle eines Notfalls oder eines anderen Fehlers, der sich auf alle Kopien der Daten innerhalb der Availability Zone auswirkt, oder bei der Zerstörung der Availability Zone verloren gehen.

² Lifecycle-Richtlinien, die am oder nach dem 26. November 2023, 12:00 Uhr PT, aktualisiert wurden, stufen Dateien, die kleiner als 128 KiB sind, der IA-Klasse zu. Weitere Informationen darüber, wie Amazon EFS einzelne Dateien und Metadaten misst und abrechnet, finden Sie unter [Wie Amazon EFS Dateisystem- und Objektgrößen meldet](#).

Abrechnung nach Speicherklassen

Ihre Abrechnung erfolgt gemäß der in jeder Speicherklasse gespeicherten Datenmenge. Ihnen werden auch Datenzugriffsgebühren in Rechnung gestellt, wenn Dateien im IA- oder Archivspeicher gelesen werden oder für Daten, die mithilfe des Lebenszyklusmanagements zwischen Speicherklassen gewechselt werden. Die AWS -Fakturierung zeigt die Kapazität für jede Speicherklasse und den gemessenen Zugriff für die Dateisystem-Speicherklasse an. Weitere Informationen finden Sie unter [Amazon EFS – Preise](#).

Darüber hinaus fallen für die Speicherklassen Infrequent Access (IA) und Archive eine Mindestabrechnungsgebühr pro Datei von 128 KiB an. Die Support für Dateien, die kleiner als 128 KiB sind, ist nur für Lebenszyklusrichtlinien verfügbar, die am oder nach 12:00 Uhr PT am 26. November 2023 aktualisiert wurden. Weitere Informationen darüber, wie Amazon EFS einzelne Dateien und Metadaten misst und abrechnet, finden Sie unter [Wie Amazon EFS Dateisystem- und Objektgrößen meldet](#).

Je nach Durchsatzmodus gelten zusätzliche Überlegungen zur Abrechnung.

- Bei Dateisystemen, die Elastic Throughput verwenden, wird Ihnen unabhängig von den Speichergebühren die monatliche Gesamtmenge der für Ihre Dateisysteme übertragenen Metadaten und Daten in Rechnung gestellt.
- Für Dateisysteme mit dem Durchsatzmodus „Bereitgestellt“ wird der Durchsatz abgerechnet, der über dem Volumen liegt, das basierend auf der in der EFS-Standardspeicherklasse gespeicherten Datenmenge bereitgestellt wird.
- Für Dateisysteme mit Bursting-Durchsatz richtet sich der zulässige Durchsatz ausschließlich nach der Menge der in der EFS-Standard-Speicherklasse gespeicherten Daten.

Weitere Informationen zu den EFS-Durchsatzmodi finden Sie unter [Durchsatzmodi](#).

Note

Bei der Sicherung von EFS-Dateisystemen mit AWS Backup Lifecycle Management-Unterstützung fallen keine Datenzugriffsgebühren an. Weitere Informationen AWS Backup zu Amazon EFS finden Sie unter [Sicherung von EFS-Dateisystemen](#).

Lebenszyklusmanagement

Verwenden Sie Lifecycle Management, um Ihre Dateisysteme so zu verwalten, dass sie während ihres gesamten Lebenszyklus kostengünstig gespeichert werden. Das Lebenszyklusmanagement überträgt Daten automatisch zwischen Speicherklassen entsprechend der für das Dateisystem definierten Lebenszykluskonfiguration. Die Lebenszykluskonfiguration ist eine Reihe von Lebenszyklusrichtlinien, die festlegen, wann die Dateisystemdaten in eine andere Speicherkategorie überführt werden sollen. Weitere Informationen finden Sie unter [Verwaltung des Speicherlebenszyklus](#).

Replikation

Sie können mithilfe der Replikation ein Replikat Ihres Amazon EFS-Dateisystems nach Ihren Wünschen erstellen. AWS-Region Bei der Replikation werden die Daten und Metadaten auf Ihrem EFS-Dateisystem automatisch und transparent in ein neues EFS-Zielfilesystem repliziert, AWS-Region das in einem von Ihnen ausgewählten Dateisystem erstellt wird. EFS synchronisiert die Quell- und Zielfilesysteme automatisch. Die Replikation erfolgt kontinuierlich und ist auf ein Recovery Point Objective (RPO) und ein Recovery Time Objective (RTO) von Minuten ausgelegt. Diese Features unterstützen Sie dabei, Ihre Ziele im Bereich Compliance und Business Continuity zu erreichen. Weitere Informationen finden Sie unter [EFS-Dateisysteme replizieren](#).

Erste Schritte mit Amazon EFS

Wenn Sie Amazon Elastic File System (Amazon EFS) zum ersten Mal verwenden, führen Sie die folgenden Schritte aus, um mit Ihrem ersten EFS-Dateisystem zu beginnen.

1. [Lesen Sie die Voraussetzungen für den Einstieg](#)
2. [Erstellen Sie Ihr EFS-Dateisystem und starten Sie Ihre EC2-Instance](#)
3. [Übertragen Sie Dateien in Ihr EFS-Dateisystem mit AWS DataSync](#)
4. [Säubern Sie Ressourcen und schützen Sie Ihr AWS Konto](#)

Voraussetzungen

Bevor Sie die Schritte „Erste Schritte“ ausführen, stellen Sie sicher, dass Sie die folgenden Voraussetzungen erfüllen:

- Sie sind mit Amazon EC2 eingerichtet und mit dem Starten von EC2-Instances vertraut. Sie benötigen einen AWS-Konto, einen Benutzer mit Administratorzugriff, ein key pair und eine Sicherheitsgruppe. Weitere Informationen finden Sie unter [Erste Schritte mit Amazon EC2](#) im Amazon-EC2-Benutzerhandbuch.
- Ihre Amazon Virtual Private Cloud (Amazon VPC), EC2- und EFS-Ressourcen befinden sich alle in derselben Region AWS-Region und Sie haben eine Standard-VPC in der Region. Wenn Sie keine Standard-VPC haben oder wenn Sie Ihr Dateisystem von einer neuen VPC mit neuen oder vorhandenen Sicherheitsgruppen mounten möchten, finden Sie weitere Informationen unter [Verwenden von VPC-Sicherheitsgruppen](#)
- Sie haben die Standardregel für eingehenden Datenverkehr für die Standardsicherheitsgruppe nicht geändert.

Sie können auch eine ähnliche Übung für die ersten Schritte durchführen, indem Sie die Befehle AWS Command Line Interface (AWS CLI) verwenden, um die EFS-API-Aufrufe zu tätigen. Weitere Informationen finden Sie unter [Tutorial: Erstellen Sie ein EFS-Dateisystem und mounten Sie es auf einer EC2-Instance mit dem AWS CLI](#).

Erstellen Sie Ihr EFS-Dateisystem und starten Sie Ihre EC2-Instance

Nachdem Sie sichergestellt haben, dass Sie die Voraussetzungen für diese Übung mit den ersten Schritten haben, können Sie Ihr EFS-Dateisystem erstellen und Ihre EC2-Instance starten. Der schnellste Weg, alle notwendigen Schritte für den Einstieg in Ihr erstes EFS-Dateisystem durchzuführen, besteht darin, den EC2-Startassistenten beim Instance-Start zu verwenden.

Note

Sie können Amazon EFS nicht mit Microsoft Windows-basierten EC2-Instances verwenden.

So erstellen Sie Ihr EFS-Dateisystem und starten Ihre EC2-Instance mit dem EC2-Startassistenten

Anweisungen zum Erstellen und Mounten Ihres EFS-Dateisystems bei der Erstellung eines EC2-Instance-Starts finden Sie unter [Verwenden von Amazon EFS mit Amazon EC2](#) im Amazon EC2-Benutzerhandbuch.

Die folgenden Schritte führen Sie aus, wenn Sie beim Instance-Start ein EFS-Dateisystem erstellen.

1. Erstellen Sie eine EC2-Instance, die auf einem Linux-Betriebssystem ausgeführt wird, und verwenden Sie dabei das key pair und die Netzwerkeinstellungen, die Sie wählen.
2. Erstellen Sie ein gemeinsam EFS EFS-Dateisystem mit den empfohlenen Einstellungen, das automatisch in die EC2-Instance eingebunden wird.
3. Starten Sie die EC2-Instance, sodass das EFS-Dateisystem für Dateiübertragungen sofort verfügbar ist.

Alternativ können Sie in der Amazon EFS-Konsole Dateisysteme mit empfohlenen Einstellungen oder benutzerdefinierten Einstellungen erstellen. Sie können auch die AWS CLI und die API verwenden, um Dateisysteme zu erstellen. Weitere Informationen zu all Ihren Optionen zum Erstellen eines Dateisystems finden Sie unter [EFS-Dateisysteme erstellen](#).

Übertragen Sie Dateien in Ihr EFS-Dateisystem mit AWS DataSync

Nachdem Sie ein EFS-Dateisystem erstellt haben, können Sie Dateien von einem vorhandenen Dateisystem auf dieses übertragen, indem Sie AWS DataSync DataSync ist ein

Datenübertragungsdienst, der das Verschieben und Replizieren von Daten zwischen lokalen Speichersystemen und AWS Speicherdiensten über das Internet vereinfacht, automatisiert und beschleunigt. Direct Connect DataSync kann Ihre Dateidaten und auch Dateisystem-Metadaten wie Eigentum, Zeitstempel und Zugriffsberechtigungen übertragen.

Mehr über DataSync erfahren Sie unter [AWS DataSync](#).

Voraussetzungen

Bevor Sie Dateien in das EFS-Dateisystem übertragen, stellen Sie sicher, dass Sie über Folgendes verfügen:

- Ein NFS-Quelldateisystem, von dem Sie Dateien übertragen können. Dieses Quellsystem muss über NFS Version 3, Version 4 oder 4.1 zugreifbar sein. Zu den Dateisystemen gehören beispielsweise Dateisysteme, die sich in einem lokalen Rechenzentrum befinden, selbstverwaltete In-Cloud-Dateisysteme und EFS-Dateisysteme.
- Sie sind für die Verwendung eingerichtet. DataSync Weitere Informationen finden Sie unter [Erste Schritte mit AWS DataSync](#) im AWS DataSync Benutzerhandbuch.

Um Dateien in Ihr EFS-Dateisystem zu übertragen, verwenden Sie AWS DataSync

Anweisungen DataSync zur Übertragung von Dateien in ein EFS-Dateisystem finden Sie unter [Übertragen von Daten mit AWS DataSync](#) im AWS DataSync Benutzerhandbuch.

Im Folgenden sind die Schritte aufgeführt, die Sie ausführen, wenn Sie Dateien mithilfe von in das EFS-Dateisystem übertragen DataSync.

1. Stellen Sie eine Verbindung zu Ihrer EC2- Instance her. Weitere Informationen finden Sie unter [Mit Ihrer EC2-Instance verbinden](#) im Amazon-EC2-Benutzerhandbuch.
2. Laden Sie einen Agenten in Ihrer Umgebung herunter, stellen Sie ihn bereit und aktivieren Sie ihn.
3. Erstellen und konfigurieren Sie einen Quell- und Zielspeicherort.
4. Erstellen und konfigurieren Sie eine Aufgabe.
5. Führen Sie die Aufgabe aus, um Dateien von der Quelle zum Ziel zu übertragen.

Säubern Sie Ressourcen und schützen Sie Ihr AWS Konto

Wenn Sie mit dieser Übung „Erste Schritte“ fertig sind, führen Sie die folgenden Schritte aus, um Ihre Ressourcen zu bereinigen und Ihre zu schützen. AWS-Konto

So bereinigen Sie Ihre Ressourcen und schützen Ihr Konto

1. Stellen Sie eine Verbindung zu Ihrer EC2- Instance her. Weitere Informationen finden Sie unter [Mit Ihrer EC2-Instance verbinden](#) im Amazon-EC2-Benutzerhandbuch.
2. Heben Sie das Mounting des EFS-Dateisystems mit dem folgenden Befehl auf.

```
$ sudo umount efs
```

3. Öffnen Sie die Amazon Elastic File System-Konsole unter <https://console.aws.amazon.com/efs/>.
4. Löschen Sie das EFS-Dateisystem, das Sie im ersten Schritt der Übung Erste Schritte erstellt haben.
 - a. Wählen Sie das EFS-Dateisystem, das Sie löschen möchten, aus der Liste der Dateisysteme aus.
 - b. Klicken Sie bei Aktionen auf Dateisystem löschen.
 - c. Geben Sie im Dialogfeld Dateisystem dauerhaft löschen die Dateisystem-ID für das EFS-Dateisystem ein, das Sie löschen möchten, und klicken Sie auf Dateisystem löschen.
5. Beenden Sie die EC2-Instance, die Sie für diese Übung „Erste Schritte“ gestartet haben. Anweisungen finden Sie im AWS IAM Identity Center Benutzerhandbuch unter [Amazon EC2 EC2-Instances beenden](#).
6. Wenn Sie für diese Übung „Erste Schritte“ eine Sicherheitsgruppe erstellt haben, löschen Sie sie. Anweisungen finden Sie im AWS IAM Identity Center Benutzerhandbuch unter [Löschen einer Sicherheitsgruppe](#).



Warning

Löschen Sie nicht die Standardsicherheitsgruppe für Ihre VPC.

Den Amazon EFS-Client installieren

Wir empfehlen Ihnen, den Amazon EFS-Client (`amazon-efs-utils`) zu installieren, eine Open-Source-Sammlung von Tools für Amazon EFS. Der Amazon EFS-Client enthält einen Mount-Helper, ein Programm, das das Mounten von EFS-Dateisystemen vereinfacht. Der Client ermöglicht auch die Verwendung von Amazon CloudWatch zur Überwachung des Mount-Status eines EFS-Dateisystems und umfasst Tools, die die Verschlüsselung von Daten während der Übertragung für Amazon EFS-Dateisysteme erleichtern.

Sie können den Amazon EFS-Client manuell auf Amazon EC2 (EC2) -Instances installieren, auf denen [unterstützte Distributionen ausgeführt werden](#). Für bestimmte unterstützte Betriebssysteme können Sie alternativ konfigurieren AWS Systems Manager, dass das Paket automatisch installiert oder aktualisiert wird. Eine Liste der Distributionen, die Sie mit verwenden können AWS Systems Manager, finden Sie unter [Von Systems Manager unterstützte Betriebssysteme](#).

Important

Wir empfehlen, immer die aktuelle Version von `amazon-efs-utils` zu verwenden, um den vollen Funktionsumfang nutzen zu können. Beispielsweise wird das Mounten mit IPv6 Adressen in Versionen 2.3 oder höher unterstützt, in früheren Versionen jedoch nicht.

Themen

- [Abhängigkeiten für EFS-Tools](#)
- [Unterstützte Distributionen](#)
- [Manuelles Installieren des Amazon-EFS-Clients](#)
- [Automatisches Installieren oder Aktualisieren des Amazon EFS-Clients mit AWS Systems Manager](#)
- [Installation und Aktualisierung botocore](#)
- [Upgraden von stunnel](#)
- [Den FIPS-Modus aktivieren](#)

Abhängigkeiten für EFS-Tools

Für `amazon-efs-utils` gelten folgende Abhängigkeiten, die beim Installieren des Pakets `amazon-efs-utils` ebenfalls installiert werden:

- NFS-Client
 - `nfs-utils` für RHEL-, CentOS-, Amazon Linux- und Fedora-Distributionen
 - `nfs-common` für Debian- und Ubuntu-Distributionen
- Netzwerk-Relay (Stunnel-Paket, Version 4.56 oder höher)
- Python (Version 3.4 oder höher)
- OpenSSL 1.0.2 oder höher

Note

Wenn der EFS-Mount-Helper mit Transport Layer Security (TLS) verwendet wird, erzwingt der Mount-Helper standardmäßig die Überprüfung des Zertifikat-Hostnamens. Der EFS-Mount-Helper verwendet das `stunnel` Programm für seine TLS-Funktionalität. In manchen Linux-Versionen ist keine `stunnel`-Version enthalten, die diese TLS-Features standardmäßig unterstützt. Wenn Sie eine dieser Linux-Versionen verwenden, schlägt das Mounten eines EFS-Dateisystems mithilfe von TLS fehl.

Nachdem Sie das `amazon-efs-utils` Paket installiert haben, aktualisieren Sie `stunnel`. Siehe [Upgraden von stunnel](#).

Sie können AWS Systems Manager damit Amazon EFS-Clients verwalten und die Aufgaben automatisieren, die für die Installation oder Aktualisierung des `amazon-efs-utils` Pakets auf Ihren EC2-Instances erforderlich sind. Weitere Informationen finden Sie unter [Automatisches Installieren oder Aktualisieren des Amazon EFS-Clients mit AWS Systems Manager](#).

Informationen zu Problemen mit der Verschlüsselung finden Sie unter [Fehlerbehebung bei der Verschlüsselung](#).

Unterstützte Distributionen

Der Amazon-EFS-Client wurde anhand der folgenden Linux- und Mac-Distributionen verifiziert:

Distribution	Pakettyp	init -System
Amazon Linux 2023 (AL2023)	rpm	systemd

Distribution	Pakettyp	init -System
Amazon Linux (2AL2)	rpm	systemd
CentOS 8	rpm	systemd
Amazon Linux 1 (AL1) 2017,09	rpm	upstart
 Note AL1 AMI hat seinen Stand end-of-life am 31. Dezember 2023 erreicht und wird nicht für amazon-efs-utils Pakete unterstützt, die im April 2024 oder später veröffentlicht wurden (Version 2.0 und höher).		
Debian 11	deb	systemd
Fedora 29-32	rpm	systemd
macOS Big Sur		launchd
macOS Monterey		launchd
macOS Ventura		launchd
macOS Sonoma		launchd
OpenSUSE Leap, Tumbleweed	RPM	systemd
Oracle 8	rpm	systemd
RedHat Enterprise Linux (RHEL) 8, 9	rpm	systemd
SUSE Linux Enterprise Server (SLES) 12, 15	RPM	systemd

Distribution	Pakettyp	init -System
Ubuntu 16.04 LTS, 18.04 LTS, 20.04 LTS, 22.04 LTS	deb	systemd

Eine vollständige Liste der unterstützten Distributionen, gegen die das Paket verifiziert wurde, finden Sie auf Github. [amazon-efs-utils README](#)

Manuelles Installieren des Amazon-EFS-Clients

Sie können den Amazon EFS-Client manuell auf Amazon EC2 (EC2) -Linux-Instances und auf EC2-Mac-Instances installieren, auf denen macOS Big Sur, macOS Monterey und macOS Ventura ausgeführt werden. Eine Liste der Distributionen, die den Amazon EFS-Client unterstützen, finden Sie unter [Unterstützte Distributionen](#)

Die Installationsverfahren für unterstützte Betriebssysteme werden in den folgenden Abschnitten beschrieben.

Themen

- [Installation des Amazon EFS-Clients auf Amazon EC2 EC2-Linux-Instances](#)
- [Installation des Amazon-EFS-Clients auf anderen Linux-Distributionen](#)
- [Installation des Amazon-EFS-Clients auf EC2-Mac-Instances, auf denen macOS Big Sur, macOS Monterey oder macOS Ventura ausgeführt wird](#)

Installation des Amazon EFS-Clients auf Amazon EC2 EC2-Linux-Instances

Das `amazon-efs-utils` Paket für die Installation auf Amazon EC2 EC2-Linux-Instances von den folgenden Speicherorten aus:

- Die Amazon Machine Images (AMI) -Paket-Repositorys für Amazon Linux. Die folgenden Anweisungen beziehen sich auf die Installation des `amazon-efs-utils` Pakets aus den AMI-Paket-Repositorys.
- Das AWS [efs-utils](#) GitHub Repository. Weitere Hinweise zur Installation des `amazon-efs-utils` Pakets von GitHub finden Sie unter [Installation des Amazon-EFS-Clients auf anderen Linux-Distributionen](#).

 Note

- Wenn Sie verwenden Direct Connect, finden Sie Installationsanweisungen unter [Voraussetzungen](#).
- Das Amazon Linux 1 (AL1) AMI wurde end-of-life am 31. Dezember 2023 erreicht und wird nicht für `amazon-efs-utils` Pakete unterstützt, die im April 2024 und später veröffentlicht wurden (Version 2.0 und höher). Wir empfehlen Ihnen, Anwendungen auf Amazon Linux 2023 (AL2023) zu aktualisieren, was langfristigen Support bis 2028 beinhaltet.

So installieren Sie das **amazon-efs-utils** Paket aus dem AMI-Paket-Repository auf EC2-Linux-Instances

1. Stellen Sie sicher, dass Sie eine AL2023- oder Amazon Linux 2 (AL2) EC2-Instance erstellt haben. Informationen dazu finden Sie unter [Schritt 1: Starten](#) einer Instance.
2. Greifen Sie über Secure Shell (SSH) auf das Terminal für die Instance zu und melden Sie sich mit dem entsprechenden Benutzernamen an. Weitere Informationen finden Sie unter [Mit Ihrer EC2-Instance verbinden](#) im Amazon-EC2-Benutzerhandbuch.
3. Zur Installation des `amazon-efs-utils` Pakets führen Sie den folgenden Befehl aus:

```
sudo yum install -y amazon-efs-utils
```

Installation des Amazon-EFS-Clients auf anderen Linux-Distributionen

Wenn Sie das `amazon-efs-utils` Paket nicht aus den Amazon Linux AMI-Paket-Repositorys beziehen möchten, ist es auch unter GitHub verfügbar.

Nachdem Sie das Paket geklont haben, können Sie `amazon-efs-utils` mit einer der folgenden Methoden erstellen und installieren, je nachdem, welcher Pakettyp von Ihrer Linux-Distribution unterstützt wird:

- RPM — Dieser Pakettyp wird von AL2023 Amazon Linux 2, Red Hat Linux, CentOS und ähnlichen unterstützt.
- DEB – Dieser Pakettyp wird von Ubuntu, Debian und ähnlichen Distributionen unterstützt.

Anweisungen zur Installation des `amazon-efs-utils` Pakets für andere Linux-Distributionen finden Sie unter [On other Linux Distributionen](#) in der `amazon-efs-utils` README-Datei auf Github.

Installation des Amazon-EFS-Clients auf EC2-Mac-Instances, auf denen macOS Big Sur, macOS Monterey oder macOS Ventura ausgeführt wird

Das `amazon-efs-utils`-Paket ist für die Installation auf EC2-Mac-Instances verfügbar, auf denen macOS Big Sur, macOS Monterey oder macOS Ventura ausgeführt wird.

Anweisungen zur Installation des `amazon-efs-utils` Pakets auf Mac-Instanzen finden Sie unter [Auf macOS Big Sur, macOS Monterey, macOS Sonoma und macOS Ventura Distribution](#) in der `amazon-efs-utils` README-Datei auf Github.

Nächste Schritte

Nachdem Sie `amazon-efs-utils` auf Ihrer EC2-Instance installiert haben, fahren Sie mit den nächsten Schritten zum Mounten Ihres Dateisystems fort:

- [Installieren](#) Sie, `botocore` damit Sie Amazon verwenden können CloudWatch , um den Mount-Status Ihres Dateisystems zu überwachen.
- [Führen Sie ein Upgrade auf die neueste Version von `stunnel`](#), um die Verschlüsselung von Daten während der Übertragung zu aktivieren.
- [Mounten Sie Ihr Dateisystem](#) mit der EFS-Mountinghilfe ein.

Automatisches Installieren oder Aktualisieren des Amazon EFS-Clients mit AWS Systems Manager

Sie können AWS Systems Manager verwenden, um die Verwaltung des Amazon EFS-Clients (`amazon-efs-utils`) zu vereinfachen. AWS Systems Manager ist ein AWS Service, mit dem Sie Ihre Infrastruktur anzeigen und steuern können. Mit AWS Systems Manager können Sie die Aufgaben automatisieren, die für die Installation oder Aktualisierung des `amazon-efs-utils` Pakets auf Ihren Amazon EC2 (EC2) -Instances erforderlich sind. Mit den Systems Manager-Funktionen wie Distributor und State Manager können Sie die folgenden Prozesse automatisieren:

- Aufrechterhaltung der Versionskontrolle über den Amazon-EFS-Client.
- Zentrales Speichern und systematisches Verteilen des Amazon-EFS-Clients an Ihre Amazon EC2-Instances.

- Automatisieren Sie den Prozess, Ihre EC2-Instances in einem definierten Zustand zu halten.

Weitere Informationen finden Sie im [AWS Systems Manager -Benutzerhandbuch](#).

Was macht der Amazon-EFS-Client während der Installation

Sie verwenden den Amazon EFS-Client, um die Überwachung von CloudWatch Amazon-Protokollen für den Mount-Status des Dateisystems `stunnel` zu automatisieren und für ausgewählte Linux-Distributionen ein Upgrade auf die neueste Version durchzuführen. Wenn Sie den Amazon-EFS-Client mithilfe von Systems Manager auf Ihren Amazon EC2-Instances installieren, werden folgende Aktionen ausgeführt:

- Installiert das Paket `botocore` mit denselben Schritten wie in [Installation und Aktualisierung botocore](#) beschrieben. Der Amazon-EFS-Client verwendet `botocore`, um den Mounting-Status des EFS-Dateisystems zu überwachen.
- Ermöglicht die Überwachung des Bereitstellungsstatus des EFS-Dateisystems in CloudWatch Protokollen durch Aktualisierung `efs-utils.conf`. Weitere Informationen finden Sie unter [Überwachung erfolgreicher und fehlgeschlagener Einhängerversuche](#).
- Für EC2-Instances, auf denen RHEL7 oder CentOS7 läuft, führt der Amazon-EFS-Client automatisch ein Upgrade von `stunnel` durch, wie in [Upgraden von stunnel](#) beschrieben. Ein Upgrade von `stunnel` ist erforderlich, um ein EFS-Dateisystem mit TLS erfolgreich zu mounten, und die mit RHEL7 und CentOS7 ausgelieferte Version `stunnel` unterstützt den Amazon-EFS-Client (`amazon-efs-utils`) nicht.

Von Systems Manager unterstützte Betriebssysteme

Auf Ihren EC2-Instances muss eines der folgenden Betriebssysteme ausgeführt werden, damit AWS Systems Manager sie für die automatische Aktualisierung oder Installation des Amazon EFS-Clients verwendet werden können.

Plattform	Version der Plattform	Architektur
Amazon Linux 2023 (AL2023)	AL2023	x86_64, arm64 (Graviton2-Prozessoren oder höher)
Amazon Linux (2AL2)	2.0	x86_64, arm64 (Amazon Linux 2, A1 Instance-Typen)

Plattform	Version der Plattform	Architektur
Amazon Linux (1AL1)	2017.09, 2018.03	x86_64
 Note Amazon Linux 1 (AL1) AMI wurde end-of-life am 31. Dezember 2023 erreicht und wird nicht für amazon-efs-utils Pakete unterstützt, die im April 2024 und höher veröffentlicht wurden (Version 2.0 und höher). Wir empfehlen Ihnen, Anwendungen auf Amazon Linux 2023 (AL2023) zu aktualisieren, was langfristigen Support bis 2028 beinhaltet.		
CentOS	7, 8	x86_64
Red Hat Enterprise Linux (RHEL)	8, 9	x86_64, arm64
SUSE Linux Enterprise Server (SLES)	12, 15	x86_64
Ubuntu Server	16,04, 18,04, 20,04	x86_64, arm64 (Ubuntu Server 16 and later, A1 Instance-Typen)

Konfiguration AWS Systems Manager für die Installation des EFS-Clients

Es sind zwei einmalige Konfigurationen erforderlich, um Systems Manager so einzurichten, dass das `amazon-efs-utils` Paket automatisch installiert oder aktualisiert wird.

1. Konfigurieren Sie ein AWS Identity and Access Management (IAM-) Instanzprofil mit den erforderlichen Berechtigungen.
2. Konfigurieren Sie eine Zuordnung (einschließlich des Zeitplans), die vom State Manager für die Installation oder Aktualisierung verwendet wird.

Schritt 1: Konfigurieren Sie ein (IAM)-Instance-Profil mit den erforderlichen Berechtigungen.

Hat standardmäßig AWS Systems Manager keine Berechtigung, Ihre Amazon EFS-Clients zu verwalten und das `amazon-efs-utils` Paket zu installieren oder zu aktualisieren. Sie müssen den Zugriff auf Systems Manager über ein AWS Identity and Access Management (IAM)-Instance-Profilgewähren. Ein Instance-Profil ist ein Container, der beim Start IAM-Rolleninformationen an eine Amazon EC2 (EC2) -Instance weitergibt.

Verwenden Sie die Richtlinie für `AmazonElasticFileSystemsUtils` AWS verwaltete Berechtigungen, um Rollen die entsprechenden Berechtigungen zuzuweisen. Sie können eine neue Rolle für Ihr Instance-Profil erstellen oder die `AmazonElasticFileSystemsUtils` Berechtigungsrichtlinie zu einer vorhandenen Rolle hinzufügen. Anschließend müssen Sie dieses Instance-Profil verwenden, um Ihre EC2-Instances zu starten. Weitere Informationen finden Sie unter [Erforderliche Instance-Berechtigungen für Systems Manager konfigurieren](#).

Schritt 2: Konfigurieren Sie eine von State Manager verwendete Zuordnung

Das `amazon-efs-utils` Paket ist im Lieferumfang von Distributor enthalten und kann sofort auf verwalteten EC2-Instances bereitgestellt werden. Um die neueste Version anzuzeigen, die für `amazon-efs-utils` die Installation verfügbar ist, können Sie die AWS Systems Manager Konsole oder Ihr bevorzugtes AWS Befehlszeilentool verwenden. Um auf den Distributor zuzugreifen, öffnen Sie den <https://console.aws.amazon.com/systems-manager/> und wählen Sie im linken Navigationsbereich die Option Distributor aus. Suchen Sie Amazon EFSUtils im Bereich Owned by Amazon. Wählen Sie Amazon EFSUtils, um die Paketdetails zu sehen. Weitere Informationen finden Sie unter [Pakete anzeigen](#).

Mit State Manager können Sie das `amazon-efs-utils` Paket sofort oder nach einem Zeitplan auf Ihren verwalteten EC2-Instances installieren oder aktualisieren. Darüber hinaus können Sie sicherstellen, dass `amazon-efs-utils` es automatisch auf neuen EC2-Instances installiert wird. Weitere Informationen zur Installation oder Aktualisierung von Paketen mit Distributor und State Manager finden Sie unter [Arbeiten mit Distributor](#).

Informationen zur automatischen Installation oder Aktualisierung des `amazon-efs-utils` Pakets auf Instanzen mithilfe der Systems Manager Manager-Konsole finden Sie unter [Planung einer Paketinstallation oder eines Updates \(Konsole\)](#). Daraufhin werden Sie aufgefordert, eine Zuordnung für State Manager zu erstellen, die den Status definiert, den Sie auf eine Reihe von Instances anwenden möchten. Verwenden Sie die folgenden Eingaben, wenn Sie Ihre Zuordnung erstellen:

- Wählen Sie für Parameter Aktion > Installation und Installationstyp > Direktes Update.
- Für Ziele lautet die empfohlene Einstellung Alle Instances auswählen, um alle neuen und vorhandenen EC2-Instances als Ziele für die automatische Installation oder Aktualisierung von Amazon EFSUtils zu registrieren. Alternativ können Sie Instance-Tags angeben, Instances manuell auswählen oder eine Ressourcengruppe auswählen, um die Zuordnung auf eine Teilmenge von Instances anzuwenden. Wenn Sie Instance-Tags angeben, müssen Sie Ihre EC2-Instances mit den Tags starten, damit AWS Systems Manager den Amazon EFS-Client automatisch installieren oder aktualisieren kann.
- Für „Zeitplan angeben“ EFSUtils ist die empfohlene Einstellung für Amazon alle 30 Tage. Sie können die Steuerelemente verwenden, um einen Cron- oder Ratenplan für die Vereinigung zu erstellen.

Informationen zum Mounten von EFS-Dateisystemen auf mehreren EC2-Instances finden Sie unter [Mounten von EFS auf mehrere EC2-Instances](#). AWS Systems Manager

Installation und Aktualisierung **botocore**

Der Amazon EFS-Client verwendet `botocore`, um mit anderen AWS Diensten zu interagieren. Es ist erforderlich, wenn Sie den Erfolg oder Misserfolg des Bereitstellungsversuchs für Ihre EFS-Dateisysteme in CloudWatch Logs überwachen möchten. Weitere Informationen finden Sie unter [Überwachung erfolgreicher und fehlgeschlagener Einhängeversuche](#).

Anweisungen zur Installation und zum Upgrade `botocore` finden Sie unter [Installation botocore](#) in der `amazon-efs-utils` README-Datei auf Github.

Upgraden von **stunnel**

Für die Verschlüsselung von Daten während der Übertragung mit dem EFS-Mount-Helper ist OpenSSL Version 1.0.2 oder neuer erforderlich, und eine Version davon unterstützt sowohl **stunnel** das Online Certificate Status Protocol (OCSP) als auch die Überprüfung von Zertifikats-Hostnamen. Der EFS-Mount-Helper verwendet das **stunnel** Programm für seine TLS-Funktionalität. Beachten Sie, dass einige Linux-Versionen nicht über eine Version von **stunnel** verfügen, die diese TLS-Features standardmäßig unterstützt. Wenn Sie eine dieser Linux-Distributionen verwenden, schlägt das Mounten eines EFS-Dateisystems mithilfe von TLS fehl.

Nach der Installation des EFS-Mount-Helpers können Sie die Stunnel-Version Ihres Systems mit den folgenden Anweisungen aktualisieren.

So aktualisieren Sie **stunnel** unter Amazon Linux, Amazon Linux 2 und anderen unterstützten Linux-Distributionen (mit Ausnahme von [SLES 12](#))

1. Rufen Sie in einem Webbrowser die **stunnel** Download-Seite <https://stunnel.org/downloads.html> auf.
2. Suchen Sie die neueste **stunnel**-Version, die im `tar.gz`-Format verfügbar ist. Notieren Sie den Dateinamen, da Sie diesen in den folgenden Schritten benötigen.
3. Öffnen Sie ein Terminal auf Ihrem Linux-Client und führen Sie die folgenden Befehle wie angegeben aus.

- a. Für RPM:

```
sudo yum install -y gcc openssl-devel tcp_wrappers-devel
```

Für DEB:

```
sudo apt-get install build-essential libwrap0-dev libssl-dev
```

- b. ***latest-stunnel-version*** Ersetzen Sie es durch den Namen der Datei, die Sie zuvor in Schritt 2 notiert haben.

```
sudo curl -o latest-stunnel-version.tar.gz https://www.stunnel.org/downloads/latest-stunnel-version.tar.gz
```

- c.

```
sudo tar xvfz latest-stunnel-version.tar.gz
```

- d. `cd latest-stunnel-version/`
- e. `sudo ./configure`
- f. `sudo make`
- g. Das aktuelle stunnel-Paket ist in `bin/stunnel` installiert. Damit die neue Version installiert werden kann, müssen Sie dieses Verzeichnis mit dem folgenden Befehl löschen.

```
sudo rm /bin/stunnel
```

- h. Installation der neuesten Version:

```
sudo make install
```

- i. Erstellen Sie einen Symlink:

```
sudo ln -s /usr/local/bin/stunnel /bin/stunnel
```

Um stunnel auf macOS zu aktualisieren

- Öffnen Sie ein Terminal auf Ihrer EC2-Mac-Instance und führen Sie den folgenden Befehl aus, um auf die neueste Version von stunnel zu aktualisieren.

```
brew upgrade stunnel
```

Stunnel für SLES 12 wird aktualisiert

- Führen Sie die folgenden Befehle aus und folgen Sie den Anweisungen des Zypper-Paketmanagers, um stunnel auf Ihrer laufenden Compute-Instanz zu aktualisieren. SLES12

```
sudo zypper addrepo https://download.opensuse.org/repositories/security:Stunnel/  
SLE_12_SP5/security:Stunnel.repo  
sudo zypper refresh  
sudo zypper install -y stunnel
```

Nachdem Sie eine Version von stunnel mit den erforderlichen Features installiert haben, können Sie Ihr Dateisystem unter Verwendung von TLS mit den von Amazon EFS empfohlenen Einstellungen mounten.

Behebung von Problemen bei der Installation von Stunnel

Wenn Sie Stunnel nicht installieren können, versuchen Sie, die Überprüfung des Zertifikat-Hostnamens zu deaktivieren. Sorgen Sie außerdem für die größtmögliche Sicherheit, indem Sie das Online Certificate Status Protocol (OCSP) aktivieren.

Themen

- [Deaktivieren der Überprüfung des Hostnamens des Zertifikats](#)
- [Aktivieren des Online Certificate Status Protocol](#)

Deaktivieren der Überprüfung des Hostnamens des Zertifikats

Wenn Sie nicht in der Lage sind, die erforderlichen Abhängigkeiten zu installieren, können Sie optional die Überprüfung des Hostnamens des Zertifikats in der Konfiguration der Amazon-EFS-Mountinghilfe deaktivieren. Dies wird jedoch in Produktionsumgebungen nicht empfohlen. Gehen Sie wie folgt vor, um die Überprüfung des Hostnamens des Zertifikats zu deaktivieren:

1. Greifen Sie über Secure Shell (SSH) auf das Terminal für die EC2-Instance zu und melden Sie sich mit dem entsprechenden Benutzernamen an. Weitere Informationen finden Sie unter [Mit Ihrer EC2-Instance verbinden](#) im Amazon-EC2-Benutzerhandbuch.
2. Öffnen Sie mit einem Texteditor Ihrer Wahl die Datei `/etc/amazon/efs/efs-utils.conf`.
3. Legen Sie für den `stunnel_check_cert_hostname`-Wert „false“ fest.
4. Speichern Sie die Änderungen und schließen Sie die Datei.

Weitere Informationen zur Verwendung von Datenverschlüsselung während der Übertragung finden Sie unter [Mounting von EFS-Dateisystemen](#).

Aktivieren des Online Certificate Status Protocol

Um die Verfügbarkeit des Dateisystems für den Fall zu maximieren, dass die Zertifizierungsstelle von Ihrer VPC aus nicht erreichbar ist, ist das Online Certificate Status Protocol (OCSP) standardmäßig nicht aktiviert, wenn Sie sich für die Verschlüsselung von Daten während der Übertragung

entscheiden. Amazon EFS verwendet eine [Amazon Zertifizierungsstelle](#) (CA), um seine TLS-Zertifikate auszustellen und zu signieren, und die CA weist den Client an, OCSP zu verwenden, um auf widerrufenen Zertifikate zu prüfen. Um den Status eines Zertifikats überprüfen zu können, muss der OCSP-Endpunkt von Ihrer Virtual Private Cloud aus über das Internet zugänglich sein. Innerhalb des Service überwacht Amazon EFS kontinuierlich den Status der Zertifikate und stellt neue Zertifikate aus, um alle erkannten widerrufenen Zertifikate zu ersetzen.

Für höchste Sicherheit können Sie OCSP so aktivieren, dass Ihre Linux-Clients eine Prüfung auf widerrufenen Zertifikate ausführen können. OCSP schützt vor der böartigen Verwendung widerrufenen Zertifikate. Es ist jedoch unwahrscheinlich, dass dies innerhalb Ihrer VPC auftritt. Für den Fall, dass ein EFS-TLS-Zertifikat gesperrt wird, veröffentlicht Amazon ein Sicherheitsbulletin und veröffentlicht eine neue Version von EFS Mount Helper, die das gesperrte Zertifikat zurückweist.

So aktivieren Sie OCSP auf Ihrem Linux-Client für alle zukünftigen TLS-Verbindungen zu EFS

1. Öffnen Sie ein Terminal auf Ihrem Linux-Client.
2. Öffnen Sie mit einem Texteditor Ihrer Wahl die Datei `/etc/amazon/efs/efs-utils.conf`.
3. Legen Sie den `stunnel_check_cert_validity`-Wert auf „true“ fest.
4. Speichern Sie die Änderungen und schließen Sie die Datei.

So aktivieren Sie OCSP als Teil des **mount**-Befehls

- Verwenden Sie den folgenden Mounting-Befehl, um OCSP beim Mounten des Dateisystems zu aktivieren.

```
$ sudo mount -t efs -o tls,ocsp fs-12345678:/ /mnt/efs
```

Den FIPS-Modus aktivieren

Wenn Ihr Betriebssystem beim Mounten Ihres Dateisystems FIPS-Endpunkte (Federal Information Processing Standards) verwendet, müssen Sie den FIPS-Modus im Amazon EFS-Client aktivieren. Um den FIPS-Modus zu aktivieren, muss die `efs-utils.conf` Datei auf dem Betriebssystem geändert werden.

 Note

Der FIPS-Modus erfordert, dass die installierte Version von OpenSSL mit FIPS kompiliert wurde. Weitere Informationen zur Konfiguration von OpenSSL mit FIPS finden Sie in der OpenSSL [FIPS-README-Datei](#).

So aktivieren Sie den FIPS-Modus im Amazon EFS-Client

1. Greifen Sie über Secure Shell (SSH) auf das Terminal für Ihre Amazon EC2 EC2-Instance zu und melden Sie sich mit dem entsprechenden Benutzernamen an. Weitere Informationen finden Sie unter [Connect to your EC2 Instance](#) im Amazon Elastic Compute Cloud-Benutzerhandbuch.
2. Öffnen Sie mit einem Texteditor Ihrer Wahl die Datei `/etc/amazon/efs/efs-utils.conf`.
3. Suchen Sie die Zeile mit dem folgenden Text:

```
"fips_mode_enabled = false"
```

4. Ändern Sie den Text wie folgt:

```
"fips_mode_enabled = true"
```

5. Speichern Sie Ihre Änderungen.

EFS-Ressourcen erstellen und verwalten

Amazon EFS bietet elastischen, gemeinsam genutzten Dateispeicher. Das Dateisystem, das Sie erstellen, unterstützt gleichzeitigen Lese- und Schreibzugriff von mehreren Amazon-EC2-Instances aus. Auf das Dateisystem kann auch von allen Availability Zones aus zugegriffen werden, in AWS-Region denen es erstellt wurde.

Sie können ein Amazon EFS-Dateisystem auf EC2-Instances in Ihrer auf Amazon VPC basierenden Virtual Private Cloud (VPC) bereitstellen, indem Sie das Network File System-Protokoll der Versionen 4.0 und 4.1 () verwenden. NFSv4 Weitere Informationen finden Sie unter [So funktioniert Amazon EFS](#).

Angenommen, Sie haben in Ihrer VPC eine oder mehrere EC2-Instances gestartet. Jetzt möchten Sie ein Dateisystem auf diesen Instances erstellen und verwenden. Nachfolgend sehen Sie die typischen Schritte, die Sie durchführen müssen, um Amazon-EFS-Dateisysteme in der VPC zu verwenden:

- Erstellen Sie ein Amazon-EFS-Dateisystem – Beim Erstellen eines Dateisystems empfehlen wir die Verwendung des Name-Tags. Der Name-Tag-Wert wird in der Konsole angezeigt und erleichtert die Identifizierung des Dateisystems. Sie können dem Dateisystem auch andere optionale Tags hinzufügen.
- Erstellen von Mountingzielen für das Dateisystem – Für den Zugriff auf das Dateisystem in Ihrer VPC und das Mounting des Dateisystems auf Ihrer Amazon-EC2-Instance müssen Sie in den VPC-Subnetzen Mount-Ziele erstellen.
- Erstellen von Sicherheitsgruppen – Sowohl eine Amazon-EC2-Instance als auch ein Mount-Ziel müssen zugewiesene Sicherheitsgruppen haben. Diese Sicherheitsgruppen fungieren als virtuelle Firewall zur Steuerung des Datenverkehrs zwischen ihnen. Sie können die Sicherheitsgruppe, die Sie dem Mount-Ziel zugeordnet haben, verwenden, um den eingehenden Datenverkehr in Ihr Dateisystem zu kontrollieren. Fügen Sie dazu der Sicherheitsgruppe des Mount-Ziels eine Regel für eingehenden Datenverkehr hinzu, die den Zugriff von einer bestimmten EC2-Instance aus ermöglicht. Anschließend können Sie das Dateisystem nur auf dieser EC2-Instance mounten.

Themen

- [Übersicht über die Implementierung](#)
- [Ressource IDs](#)
- [Erstellungstoken und Idempotenz](#)
- [EFS-Dateisysteme erstellen](#)

- [Löschen von EFS-Dateisystemen](#)
- [Erstellen von Dateisystemrichtlinien](#)
- [Erstellen von Zugriffspunkten](#)
- [Access Points löschen](#)
- [Taggen von EFS-Ressourcen](#)
- [Tutorial: Schreibbare Unterverzeichnisse pro Benutzer erstellen](#)

Übersicht über die Implementierung

In Amazon EFS ist ein Dateisystem die primäre Ressource. Jedes Dateisystem hat Eigenschaften wie ID, Erstellungstoken, Erstellungszeit, Dateisystemgröße in Byte, Anzahl der für das Dateisystem erstellten Mount-Ziele und Lebenszyklusrichtlinien des Dateisystems.

Amazon EFS unterstützt auch andere Ressourcen, um die primäre Ressource zu konfigurieren. Dazu gehören Mounting-Ziele und Zugriffspunkte:

- Mounting-Ziel – Für den Zugriff auf Ihr Dateisystem müssen Sie in Ihrer VPC Mounting-Ziele erstellen. Jedes Mounting-Ziel hat die folgenden Eigenschaften: die Mounting-Ziel-ID, die ID des Subnetzes, in dem es erstellt wurde, die ID des Dateisystems, für das es erstellt wurde, eine IP-Adresse, unter der das Dateisystem gemountet werden kann, VPC-Sicherheitsgruppen sowie den Status des Mounting-Ziels. Sie können die IP-Adresse oder den DNS-Namen in Ihrem mount-Befehl verwenden.

Jedes Dateisystem verfügt über einen DNS-Namen in der folgenden Form.

```
file-system-id.efs.aws-region.amazonaws.com
```

Sie können diesen DNS-Namen in Ihrem mount-Befehl angeben, um das Amazon-EFS-Dateisystem einzuhängen. Angenommen, Sie erstellen ein `efs-mount-point`-Unterverzeichnis außerhalb Ihres Stammverzeichnisses auf Ihrer EC2-Instance oder auf Ihrem On-Premises-Server. Sie können dann den Mounting-Befehl zum Mounten des Dateisystems verwenden. Zum Beispiel können Sie auf einem Amazon Linux-AMI den folgenden mount-Befehl verwenden.

```
$ sudo mount -t nfs -o  
nfsvers=4.1,rsize=1048576,wsiz=1048576,hard,timeo=600,retrans=2,noresvport file-  
system-DNS-name:/ ~/efs-mount-point
```

Weitere Informationen finden Sie unter [Verwalten der Mountingziele](#).

- **Zugangspunkte** – Ein Zugriffspunkt wendet einen Betriebssystembenutzer, eine Gruppe und einen Dateisystempfad auf jede Dateisystemanfrage an, die über den Zugangspunkt erfolgt. Der Betriebssystembenutzer und die Gruppe des Zugriffspunkts überschreiben alle vom NFS-Client bereitgestellten Identitätsinformationen. Der Dateisystempfad wird dem Client als Stammverzeichnis des Zugriffspunkts angezeigt. Dadurch wird sichergestellt, dass jede Anwendung beim Zugriff auf freigegebene dateibasierte Datasets immer die richtige Betriebssystemidentität und das richtige Verzeichnis verwendet. Anwendungen, die den Zugriffspunkt verwenden, können nur auf Daten in einem eigenen Verzeichnis und darunter zugreifen. Weitere Informationen finden Sie unter [Arbeiten mit Zugriffspunkten](#).

Mounting-Ziele und -Tags sind Unterressourcen, die einem Dateisystem zugeordnet sind. Sie können sie nur im Kontext eines vorhandenen Dateisystems erstellen.

Amazon EFS bietet API-Vorgänge, mit denen Sie diese Ressourcen erstellen und verwalten können. Zusätzlich zu den Operationen zum Erstellen und Löschen jeder Ressource unterstützt Amazon EFS eine Operation zum Beschreiben, mit der Sie Ressourceninformationen abrufen können. Sie haben die folgenden Optionen für das Erstellen und Verwalten dieser Ressourcen:

- Verwenden Sie die Amazon-EFS-Konsole – Ein Beispiel finden Sie unter [Erste Schritte](#).
- Verwenden Sie die Amazon-EFS-Befehlszeilenschnittstelle (CLI) – Ein Beispiel finden Sie unter [Tutorial: Erstellen Sie ein EFS-Dateisystem und mounten Sie es auf einer EC2-Instance mit dem AWS CLI](#).
- Sie können diese Ressourcen auch wie folgt programmgesteuert verwalten:
 - Verwenden Sie die AWS SDKs — AWS SDKs Vereinfachen Sie Ihre Programmieraufgaben, indem Sie die zugrunde liegende Amazon EFS-API einschließen. Dazu authentifizieren die SDK-Clients auch Ihre Anforderungen mithilfe der von Ihnen bereitgestellten Zugriffsschlüssel. Weitere Informationen finden Sie unter [Beispiel-Code und Bibliotheken](#).
 - Rufen Sie die Amazon EFS-API direkt von Ihrer Anwendung aus auf — Wenn Sie die aus SDKs irgendeinem Grund nicht verwenden können, können Sie die Amazon EFS-API-Aufrufe direkt von Ihrer Anwendung aus tätigen. Allerdings müssen Sie den erforderlichen Code zur Authentifizierung Ihrer Anforderungen schreiben, wenn Sie diese Option verwenden. Weitere Informationen über die Amazon EFS API finden Sie unter [Amazon-EFS-API](#).

Ressource IDs

Amazon EFS weist allen EFS-Ressourcen bei ihrer Erstellung eindeutige Ressourcen-Identifikatoren (IDs) zu. Alle EFS-Ressourcen IDs bestehen aus einer Ressourcen-ID und einer Kombination aus Ziffern 0—9 und Kleinbuchstaben a—f.

Vor Oktober 2021 verwendeten die IDs neu erstellten Dateisystem- und Mount-Zielressourcen 8 Zeichen nach dem Bindestrich (z. B.). fs-12345678 Von Mai 2021 bis Oktober 2021 haben wir diese Ressourcentypen IDs dahingehend geändert, dass sie 17 Zeichen nach dem Bindestrich verwenden (z. B. fs-1234567890abcdef0). Je nachdem, wann Ihr Konto erstellt wurde, verfügen Sie möglicherweise über Dateisystem- und Mount-Zielressourcen mit dem IDs Kürzeren. Neue Ressourcen dieser Art erhalten jedoch die längeren IDs Ressourcen. Die Ressourcen-ID ändert sich nie.

Erstellungstoken und Idempotenz

Idempotenz stellt sicher, dass eine API-Anforderung nur einmal durchgeführt wird. Wenn bei idempotenten Anforderungen die ursprüngliche Anforderung erfolgreich abgeschlossen wird, haben nachfolgende Anforderungen keine zusätzliche Auswirkung. Dies ist nützlich, um zu verhindern, dass doppelte Jobs erstellt werden, wenn Sie mit der Amazon-EFS-API interagieren.

Die Amazon-EFS-API unterstützt Idempotenz mit Client-Anforderungstoken. Ein Client-Anforderungs-Token ist eine eindeutige Zeichenfolge, die Sie beim Senden einer API-Anforderung angeben.

Ein Client-Anforderungs-Token kann eine beliebige Zeichenfolge sein, die bis zu 64 ASCII-Zeichen enthält. Wenn Sie ein Client-Anforderungs-Token innerhalb einer Minute nach einer erfolgreichen Anforderung wiederverwenden, gibt die API die Anforderungsdetails der ursprünglichen Anforderung zurück.

Wenn Sie die Konsole verwenden, generiert diese den Token für Sie. Wenn Sie den Ablauf Benutzerdefiniert erstellen in der Konsole verwenden, hat das für Sie generierte Erstellungstoken das folgende Format:

```
"CreationToken": "console-d215fa78-1f83-4651-b026-facafd8a7da7"
```

Wenn Sie Quick Create verwenden, um ein Dateisystem mit den vom Dienst empfohlenen Einstellungen zu erstellen, hat das Erstellungstoken das folgende Format:

```
"CreationToken": "quickCreated-d7f56c5f-e433-41ca-8307-9d9c0f8a77a2"
```

EFS-Dateisysteme erstellen

Sie können ein Dateisystem erstellen, wenn Sie eine neue EC2-Startinstanz erstellen, wie in der [Übung Erste Schritte](#) beschrieben. Sie können Dateisysteme jedoch auch mithilfe der Amazon EFS-Konsole, der AWS Command Line Interface (AWS CLI) oder der Amazon EFS-API erstellen.

Wenn Sie das Dateisystem mit der Amazon EFS-Konsole erstellen, haben Sie unterschiedliche Optionen, je nachdem, ob das Dateisystem die empfohlenen Einstellungen verwenden soll oder ob Sie die Einstellungen anpassen möchten.

- Sie können Quick Create verwenden, um schnell ein Dateisystem mit den folgenden empfohlenen Einstellungen zu erstellen.
 - Regionale Verfügbarkeit
 - Lebenszyklusrichtlinien zur Umstellung des Dateisystems auf EFS Infrequent Access (IA) - Speicher nach 30 Tagen, auf EFS-Archivspeicher nach 90 Tagen und nicht zur Umstellung auf EFS-Standardspeicher
 - Verschlüsselung von Daten im Ruhezustand ist aktiviert
 - Elastischer Durchsatzmodus
 - Leistungsmodus für allgemeine Zwecke
 - Hängen Sie Ziele ein, die in jeder Availability Zone konfiguriert sind, AWS-Region in der das Dateisystem erstellt wird, und verwenden Sie dabei ein IPv4 im angegebenen Subnetz verfügbares
- Sie können Customize verwenden, um ein Dateisystem mit den von Ihnen ausgewählten Einstellungen zu erstellen.

Eine Tabelle mit den Dateisystemeinstellungen und den empfohlenen Werten finden Sie unter [Konfigurationsoptionen für Dateisysteme](#).

Erforderliche IAM-Berechtigungen für die Erstellung von Dateisystemen

Um EFS-Ressourcen wie ein Dateisystem und Zugriffspunkte zu erstellen, benötigen Sie AWS Identity and Access Management (IAM-) Berechtigungen für den entsprechenden API-Vorgang und die entsprechende Ressource.

Erstellen Sie IAM-Benutzer und gewähren Sie ihnen Berechtigungen für Amazon-EFS-Aktionen mit Benutzerrichtlinien. Sie können auch Rollen verwenden, um kontoübergreifende Berechtigungen zu gewähren. Amazon Elastic File System verwendet auch eine mit dem IAM-Dienst verknüpfte Rolle, die die erforderlichen Berechtigungen beinhaltet, um andere in AWS-Services Ihrem Namen anzurufen. Weitere Informationen zum Verwalten von Berechtigungen für die API-Operationen finden Sie unter [Identitäts- und Zugriffsmanagement für Amazon EFS](#).

Konfigurationsoptionen für Dateisysteme

EFS-Dateisysteme werden mit den in der folgenden Tabelle aufgeführten Einstellungen konfiguriert.

- Wenn Sie Quick Create verwenden, um das Dateisystem zu erstellen, wird das Dateisystem mit dem angegebenen empfohlenen Wert für die Einstellung erstellt.
- Wenn Sie mithilfe von Anpassen ein benutzerdefiniertes Dateisystem erstellen, können Sie den empfohlenen Wert für die Einstellung ändern.

EFS-Dateisystemeinstellungen

Einstellung	Description
Typ des Dateisystems	Empfohlen: Regional Der Dateisystemtyp bestimmt die Verfügbarkeit und Haltbarkeit, mit der ein EFS-Dateisystem Daten in einem speichert AWS-Region. • Regionale Dateisysteme speichern Daten und Metadaten redundant in allen Availability Zones innerhalb einer AWS-Region. Sie können in jeder Availability Zone ein Mount-Ziel in einer AWS-Region erstellen. Regional bietet ein Höchstmaß an Verfügbarkeit und Haltbarkeit. • Dateisysteme mit einer Zone speichern Daten und Metadaten redundant in einer einzigen Availability Zone. Dateisysteme mit einer Zone können nur ein einziges Mount-

Einstellung	Description
	Ziel haben. Dieses Mount-Ziel muss sich in derselben Availability Zone befinden, in der das Dateisystem erstellt wurde.  Note Dateisysteme mit einer Zone sind nur für bestimmte Availability Zones verfügbar. Weitere Informationen über Dateisystemtypen finden Sie unter Verfügbarkeit und Haltbarkeit von EFS-Dateisystemen.

Einstellung	Description
Lebenszyklusmanagement	Empfohlen: Umstellung auf IA-Speicher nach 30 Tagen und auf Archivspeicher nach 90 Tagen. Wechseln Sie nicht in den Standardpeicher. Das Lebenszyklusmanagement verwendet Lebenszyklusrichtlinien, um Dateien auf der Grundlage von Zugriffsmustern automatisch in die kostengünstigere Speicherklasse Infrequent Access (IA) zu verschieben und diese wieder herauszuholen. Wenn Sie ein Dateisystem mithilfe von erstellen AWS-Managementkonsole, wird die Lebenszyklusrichtlinie des Dateisystems mit den folgenden Standardereinstellungen konfiguriert: • Übergang in IA ist auf 30 Tage seit dem letzten Zugriff festgelegt. • Der Übergang ins Archiv ist auf 90 Tage seit dem letzten Zugriff festgelegt. • Übergang zum Standard ist auf Keine gesetzt. Wenn Sie ein Dateisystem mithilfe der AWS CLI Amazon EFS-API oder erstellen AWS SDKs, können Sie nicht gleichzeitig eine Lebenszyklusrichtlinie festlegen. Sie müssen warten, bis das Dateisystem erstellt ist, und dann die PutLifecycleConfiguration-API-Operation verwenden, um die Lebenszyklusrichtlinie zu aktualisieren.

Einstellung	Description
	Weitere Informationen zur Lebenszyklusverwaltung finden Sie unter Verwaltung des Speicherlebenszyklus .
Verschlüsselung im Ruhezustand	Empfohlen: Aktiviert Amazon EFS verwendet standardmäßig Ihren AWS Key Management Service (AWS KMS) EFS-Serviceschlüssel (<code>aws/elasticfilesystem</code>), um Daten im Ruhezustand zu verschlüsseln. Bei der Verschlüsselung werden alle darauf gespeicherten Daten und Metadaten verschlüsselt. Nachdem Sie ein EFS-Dateisystem erstellt haben, können Sie dessen Verschlüsselungseinstellung nicht ändern. Um ein unverschlüsseltes Dateisystem in ein verschlüsseltes zu konvertieren, müssen Sie ein neues Dateisystem erstellen. Weitere Informationen zur Verschlüsselung in Amazon EFS finden Sie unter Datenverschlüsselung in Amazon EFS.

Einstellung	Description
Durchsatzmodus	Empfohlen: Elastic Sie können aus den folgenden Durchsatzmodi wählen: • Elastisch — Bietet einen Durchsatz, der automatisch in Echtzeit hoch- und herunterskaliert wird, um den Leistungsanforderungen Ihres Workloads gerecht zu werden. • Bereitgestellt — Geben Sie unabhängig von der Größe des Dateisystems den gewünschten Durchsatz an. • Bursting – Stellt einen Durchsatz bereit, der mit der Datenmenge im Standardspeicher skaliert.  Note In Verbindung mit der Nutzung der Durchsätze „Elastic“ und „Bereitgestellt“ fallen zusätzliche Gebühren an. Weitere Informationen finden Sie unter Amazon EFS – Preise. Weitere Informationen zu den Durchsatzmodi finden Sie unter Durchsatzmodi.

Einstellung	Description
Leistungsmodus	Empfohlen: Universell einsetzbar Der allgemeine Leistungsmodus hat die niedrigste Latenz pro Vorgang und wird für alle Dateisysteme empfohlen. Max I/O ist ein Leistungstyp der vorherigen Generation, der für stark parallelisierte Workloads konzipiert wurde, die höhere Latenzen tolerieren können als der Allzweckmodus. I/O Der Max-Modus wird für One Zone-Dateisysteme oder Dateisysteme, die Elastic Throughput verwenden, nicht unterstützt.  Important Aufgrund der höheren Latenzen pro Vorgang beim Modus „Max. E/A“ empfehlen wir, für alle Dateisysteme den Allzweckleistungsmodus zu verwenden. Weitere Informationen finden Sie unter Leistungsmodi.

Einstellung	Description
Netzwerkzugriff	Empfohlen: Das Mount-Ziel wurde in jeder Availability Zone erstellt, in der das Dateisystem verfügbar ist Die Einstellungen für das Mount-Ziel lauten wie folgt: • VPC — Sie können Mount-Ziele für das Dateisystem jeweils in einer VPC erstellen. Standardmäßig ist das Standardsubnetz der EC2-Instance ausgewählt. Ändern Sie die VPC, wenn Sie von einer anderen VPC aus auf das Dateisystem zugreifen müssen. - Wenn Sie die VPC nach der Erstellung des Dateisystems ändern möchten, müssen Sie zuerst die Mount-Ziele aus der aktuellen VPC löschen. Weitere Informationen finden Sie unter Änderung der Mount-Ziel-VPC. • Availability Zone und Subnetz-ID — Sie können die Availability Zone oder das Subnetz für das Mount-Ziel ändern. • IP-Adresstyp — Standardmäßig verwendet Amazon EFS eine IPv4 Adresse, unter der das Mount-Ziel erstellt wird. Sie können sich alternativ dafür entscheiden, IPv6 nur IPv6 Adressen zu unterstützen, oder Dual-Stack, um beide IPv6 Adressen IPv4 zu unterstützen.  Note Der IP-Adresstyp muss mit dem IP-Typ des Subnetzes übereinstimmen. Darüber hinaus hat der IP-Adress

Einstellung	Description
	typ Vorrang vor dem IP-Adressierungsattribut Ihres Subnetzes. Wenn der IP-Adresstyp beispielsweise IPv4 -only ist und das IPv6 Adressierungsattribut für Ihr Subnetz aktiviert ist, erhalten die im Subnetz erstellten Netzwerkschnittstellen eine IPv4 Adresse aus dem Bereich des Subnetzes. Weitere Informationen finden Sie unter Ändern des öffentlichen IP-Adressierungsattributs Ihres Subnetzes. • IP-Adresse — Standardmäßig erstellt Amazon EFS das Mount-Ziel an einer verfügbaren Adresse im angegebenen Subnetz. Alternativ können Sie die IP-Adresse für das Mount-Ziel angeben. Sie können die IP-Adresse eines Mount-Ziels nicht ändern, nachdem es erstellt wurde. Sie müssen das Mount-Ziel löschen und ein neues mit der neuen Adresse erstellen. • Sicherheitsgruppe — Die dem VPC-Ziel zugewiesene Sicherheitsgruppe wird standardmäßig verwendet. Sie können Sicherheitsgruppen hinzufügen oder entfernen. Weitere Informationen zu Sicherheitsgruppen finden Sie unter Sicherheitsgruppen für Mount-Ziele ändern. Wenn Sie ein Dateisystem mithilfe der AWS CLI, Amazon EFS-API oder erstellen AWS SDKs, können Sie nicht gleichzeitig ein Mount-Ziel erstellen. Sie müssen warten, bis

Einstellung	Description
	das Dateisystem erstellt ist, und dann den CreateMountTarget API-Vorgang verwenden, um die Mount-Ziele zu erstellen. Weitere Hinweise zu Mount-Zielen finden Sie unter Verwalten der Mountingziele.

Schnelles Erstellen mit der Konsole

Verwenden Sie die Amazon EFS-Konsole, um ein Amazon EFS-Dateisystem mit den empfohlenen Einstellungen zu erstellen. Wenn Sie ein Dateisystem mit einer benutzerdefinierten Konfiguration erstellen möchten, finden Sie weitere Informationen unter [Benutzerdefiniertes Erstellen mithilfe der Konsole](#).

So erstellen Sie schnell ein Amazon EFS-Dateisystem mit den empfohlenen Einstellungen

1. Melden Sie sich bei der an AWS-Managementkonsole und öffnen Sie die Amazon EFS-Konsole unter <https://console.aws.amazon.com/efs/>.
2. Klicken Sie auf Dateisystem erstellen, um das Dialogfeld Dateisystem erstellen zu öffnen.
3. (Optional) Geben Sie einen Namen für das Dateisystem ein.
4. Wählen Sie für Virtual Private Cloud (VPC) Ihre VPC aus oder behalten Sie Ihre Standard-VPC bei.
5. Wählen Sie Erstellen, um ein Dateisystem zu erstellen, das die folgenden vom Service empfohlenen Einstellungen verwendet:
 - Regionale Verfügbarkeit
 - Lebenszyklusrichtlinien zur Umstellung des Dateisystems auf EFS Infrequent Access (IA) - Speicher nach 30 Tagen, auf EFS-Archivspeicher nach 90 Tagen und nicht zur Umstellung auf EFS-Standardspeicher
 - Verschlüsselung von Daten im Ruhezustand ist aktiviert
 - Elastischer Durchsatzmodus
 - Leistungsmodus für allgemeine Zwecke

- Hängen Sie Ziele ein, die in jeder Availability Zone konfiguriert sind, AWS-Region in der das Dateisystem erstellt wurde, und verwenden Sie dabei ein IPv4 im angegebenen Subnetz verfügbares

Die Seite Dateisysteme erscheint mit einem Banner oben, das den Status des von Ihnen erstellten Dateisystems anzeigt. Ein Link zum Zugriff auf die Seite mit den Dateisystemdetails wird im Banner angezeigt, sobald das Dateisystem verfügbar ist.

Weitere Informationen zum Dateisystemstatus finden Sie unter [Grundlegendes zum Dateisystemstatus](#).

Benutzerdefiniertes Erstellen mithilfe der Konsole

In diesem Abschnitt wird beschrieben, wie Sie mithilfe der Amazon-EFS-Konsole ein EFS-Dateisystem mit benutzerdefinierten Einstellungen erstellen, anstatt die vom Service empfohlenen Einstellungen zu verwenden. Weitere Informationen zum Erstellen eines Dateisystems mithilfe der empfohlenen Einstellungen finden Sie unter [Schnelles Erstellen mit der Konsole](#).

Das Erstellen eines EFS-Dateisystems mit benutzerdefinierten Einstellungen mithilfe der Konsole erfolgt in vier Schritten:

- Schritt 1 – Konfigurieren Sie allgemeine Dateisystemeinstellungen, einschließlich der Speicherklasse und des Durchsatzmodus.
- Schritt 2 – Konfigurieren Sie die Dateisystem-Netzwerkeinstellungen, einschließlich der Virtual Private Cloud (VPC) und der Mount-Ziele. Legen Sie für jedes Mount-Ziel die Availability Zone, das Subnetz, die IP-Adresse und die Sicherheitsgruppen fest.
- Schritt 3 – (Optional) Erstellen Sie eine Dateisystemrichtlinie, um den NFS-Client-Zugriff auf das Dateisystem zu steuern.
- Schritt 4 – Überprüfen Sie die Dateisystemeinstellungen, nehmen Sie alle Änderungen vor und erstellen Sie dann das Dateisystem.

Schritt 1: Konfigurieren der Dateisystemeinstellungen

1. Melden Sie sich bei der an AWS-Managementkonsole und öffnen Sie die Amazon EFS-Konsole unter <https://console.aws.amazon.com/efs/>.
2. Klicken Sie auf Dateisystem erstellen, um das Dialogfeld Dateisystem erstellen zu öffnen.

3. Wählen Sie Anpassen, um ein benutzerdefiniertes Dateisystem zu erstellen, anstatt ein Dateisystem mit den empfohlenen Einstellungen zu erstellen. Die Seite mit den Dateisystemeinstellungen wird geöffnet.
4. Geben Sie für Allgemeine Einstellungen Folgendes ein:
 - a. (Optional) Geben Sie einen Namen für das Dateisystem ein.
 - b. Als Dateisystemtyp ist standardmäßig Regional ausgewählt. Wählen Sie Eine Zone, wenn Sie ein Dateisystem erstellen möchten, das Dateisystemdaten und Metadaten redundant in einer einzigen Availability Zone speichert. Wenn Sie One Zone auswählen, wählen Sie die Availability Zone aus, in der das Dateisystem erstellt werden soll, oder behalten Sie den Standardwert bei.
 - c. Für Verwaltung des Lebenszyklus ändern Sie bei Bedarf die Lebenszyklusrichtlinien.
 - Übergang in IA – Wählen Sie aus, wann Dateien in die Speicherklasse Infrequent Access (IA) umgestellt werden sollen, basierend auf der Zeit seit dem letzten Zugriff im Standardspeicher.
 - Transition into Archive (Übertragung zu Archive) – Wählen Sie aus, wann Dateien in die Speicherklasse Infrequent Access (IA) umgestellt werden sollen, basierend auf der Zeit seit dem letzten Zugriff im Standardspeicher.
 - Transition into Standard (Übergang in den Standard) – Wählen Sie aus, ob das Dateisystem in die Speicherklasse umgestellt werden soll.

Weitere Informationen zu Lebenszyklusrichtlinien finden Sie unter [Verwaltung des Speicherlebenszyklus](#).

- d. Amazon EFS verwendet standardmäßig Ihren AWS Key Management Service (AWS KMS) EFS-Serviceschlüssel (aws/elasticfilesystem), um Daten im Ruhezustand zu verschlüsseln. Um einen anderen KMS-Schlüssel für die Verschlüsselung auszuwählen, erweitern Sie Anpassen der Verschlüsselungseinstellungen und wählen Sie einen Schlüssel aus der Liste aus. Oder geben Sie eine KMS-Schlüssel-ID oder einen Amazon-Ressourcennamen (ARN) für den KMS-Schlüssel ein, den Sie verwenden möchten.

Wenn Sie einen neuen Schlüssel erstellen müssen, wählen Sie Create an, AWS KMS key um die AWS KMS Konsole zu starten und einen neuen Schlüssel zu erstellen.

Sie können die Verschlüsselung von Daten im Ruhezustand deaktivieren, indem Sie das Kontrollkästchen deaktivieren.

Sie können die Verschlüsselungseinstellung nicht ändern, nachdem das Dateisystem erstellt wurde. Weitere Informationen finden Sie unter [Datenverschlüsselung in Amazon EFS](#).

5. Für Leistungseinstellungen nehmen Sie folgendes vor:

- a. Für den Durchsatzmodus ist standardmäßig der Modus Elastic ausgewählt.
 - Um den bereitgestellten Durchsatz zu verwenden, wählen Sie Bereitgestellt aus und geben Sie im Feld Bereitgestellter Durchsatz (MiB/s) die Menge des Durchsatzes ein, der für Dateisystemanfragen bereitgestellt werden soll. Der Maximale Lesedurchsatz wird dreimal so hoch angezeigt wie der von Ihnen eingegebene Durchsatz.
 - Um den Bursting-Durchsatz zu verwenden, wählen Sie Bursting.

Nachdem Sie den Durchsatzmodus ausgewählt haben, wird eine Schätzung der monatlichen Kosten für das Dateisystem angezeigt. Sie können den Durchsatzmodus ändern, nachdem das Dateisystem verfügbar ist.

Weitere Informationen zur Auswahl des richtigen Durchsatzmodus für Ihre Leistungsanforderungen finden Sie unter [Durchsatzmodi](#).

- b. Belassen Sie bei Leistungsmodus die Standardoption Allgemeine Zwecke. Um den Leistungsmodus zu ändern, erweitern Sie Zusätzliche Einstellungen und wählen Sie dann Max. I/O aus.

Sie können den Leistungsmodus nicht mehr ändern, nachdem das Dateisystem verfügbar ist. Weitere Informationen finden Sie unter [Leistungsmodi](#).

 Important

Aufgrund der höheren Latenzen pro Vorgang beim Modus „Max. E/A“ empfehlen wir, für alle Dateisysteme den Allzweckleistungsmodus zu verwenden.

6. (Optional) Fügen Sie Tag-Schlüsselwertpaare zu Ihrem Dateisystem hinzu.
7. Wählen Sie Weiter aus, um den Netzwerkzugriff für das Dateisystem zu konfigurieren.

Schritt 2: Konfigurieren des Netzwerkzugriffs

In Schritt 2 konfigurieren Sie die Netzwerkeinstellungen des Dateisystems, einschließlich der VPC- und Mount-Ziele.

1. Wählen Sie die Virtual Private Cloud (VPC) aus, in der EC2-Instances eine Verbindung zu Ihrem Dateisystem herstellen sollen. Weitere Informationen finden Sie unter [Verwalten der Mountingziele](#).
2. Für Mount-Ziele erstellen Sie ein oder mehrere Mount-Ziele für Ihr Dateisystem. Legen Sie für jedes Mount-Ziel die folgenden Eigenschaften fest:
 - Availability Zone – Standardmäßig ist in jeder Availability Zone in einer AWS-Region ein Mount-Ziel konfiguriert. Wenn Sie kein Mount-Ziel in einer bestimmten Availability Zone haben möchten, wählen Sie Entfernen aus, um das Mount-Ziel für diese Zone zu löschen. Erstellen Sie ein Mount-Ziel in jeder Availability Zone, von der aus Sie auf Ihr Dateisystem zugreifen möchten – dies ist kostenlos.
 - Subnetz-ID – Wählen Sie aus den verfügbaren Subnetzen in einer Availability Zone aus. Das Standardsubnetz ist vorausgewählt.
 - IP-Adresstyp — Wählen Sie IPv4, IPv6 oder Dual-Stack, um beide IPv4 und IPv6 Adressen zu unterstützen.
 - IPv4 oder IPv6 Adresse — Wenn Sie die IP-Adresse kennen, unter der Sie das Mount-Ziel platzieren möchten, geben Sie sie in das IP-Adressfeld ein, das dem IP-Adresstyp entspricht. Wenn Sie keinen Wert angeben, wählt Amazon EFS eine ungenutzte IP-Adresse aus dem angegebenen Subnetz aus.
 - Sicherheitsgruppen — Standardmäßig wählt Amazon EFS die Standardsicherheitsgruppe für die VPC. Um die Sicherheitsgruppe zu ändern, löschen Sie die zugewiesene Gruppe und wählen Sie dann die Gruppe aus der Liste Sicherheitsgruppen auswählen aus. Sie können eine oder mehrere Sicherheitsgruppen für das Mount-Ziel angeben. Weitere Informationen finden Sie unter [Verwenden von VPC-Sicherheitsgruppen](#).
3. Wählen Sie Mountingziel hinzufügen, um ein Mount-Ziel für eine Availability Zone zu erstellen, in der es noch kein Mount-Ziel gibt. Wenn für jede Availability Zone ein Mount-Ziel konfiguriert ist, ist diese Option nicht verfügbar.
4. Wählen Sie Weiter aus, um die Dateisystemrichtlinie festzulegen.

Schritt 3: Erstellen einer Dateisystemrichtlinie (optional)

Optional können Sie eine Dateisystemrichtlinie für Ihr Dateisystem erstellen. Eine EFS-Dateisystemrichtlinie ist eine IAM-Ressourcenrichtlinie, die zum Steuern des NFS-Client-Zugriffs auf ein Dateisystem verwendet wird. Weitere Informationen finden Sie unter [Verwendung von IAM zur Steuerung des Zugriffs auf Dateisysteme](#).

1. In den Richtlinienoptionen können Sie eine beliebige Kombination der vorkonfigurierten Dateisystemrichtlinien auswählen:
 - Standardmäßig Root-Zugriff verhindern – Mit dieser Option wird `ClientRootAccess` aus der Gruppe der zulässigen EFS-Aktionen entfernt.
 - Standardmäßig nur Lesezugriff erzwingen – Mit dieser Option wird `ClientWriteAccess` aus der Gruppe der zulässigen EFS-Aktionen entfernt.
 - Anonymen Zugriff verhindern – Mit dieser Option wird `ClientMount` aus der Gruppe der zulässigen EFS-Aktionen entfernt.
 - Verschlüsselung während der Übertragung für alle Clients erzwingen – Mit dieser Option wird unverschlüsselten Clients der Zugriff verweigert.

Wenn Sie eine vorkonfigurierte Richtlinie auswählen, wird das Richtlinien-JSON-Objekt im Bereich des Richtlinien-Editors angezeigt.

2. Verwenden Sie Zusätzliche Berechtigungen gewähren, um weiteren IAM-Prinzipalen, einschließlich anderen, Dateisystemberechtigungen zu gewähren. AWS-Konto Wählen Sie Hinzufügen aus und geben Sie den Prinzipal-ARN der Entität ein, der Sie Berechtigungen gewähren. Wählen Sie die Berechtigungen aus, die Sie erteilen möchten. Die zusätzlichen Berechtigungen werden im Richtlinien-Editor angezeigt.
3. Sie können den Richtlinien-Editor verwenden, um eine vorkonfigurierte Richtlinie anzupassen oder Ihre eigene Dateisystemrichtlinie zu erstellen. Wenn Sie den Editor verwenden, sind die vorkonfigurierten Richtlinienoptionen nicht mehr verfügbar. Um die aktuelle Dateisystemrichtlinie zu löschen und mit der Erstellung einer neuen Richtlinie zu beginnen, wählen Sie Löschen aus.
4. Wählen Sie Weiter aus, um das Dateisystem zu überprüfen und zu erstellen.

Schritt 4: Überprüfen und Erstellen

1. Überprüfen Sie die einzelnen Dateisystem-Konfigurationsgruppen. Sie können zu diesem Zeitpunkt Änderungen an jeder Gruppe vornehmen, indem Sie Bearbeiten auswählen.
2. Wählen Sie Dateisystem erstellen, um Ihr Dateisystem zu erstellen und zur Seite Dateisysteme zurückzukehren.

Ein Banner oben zeigt, dass das neue Dateisystem gerade erstellt wird. Wenn das Dateisystem verfügbar ist, erscheint im Banner ein Link, über den Sie die Detailseite des neuen Dateisystems aufrufen können.

Erstellen Sie mit dem AWS CLI

Wenn Sie die verwenden AWS CLI, erstellen Sie diese Ressourcen der Reihe nach. Zuerst erstellen Sie ein Dateisystem. Anschließend können Sie mithilfe der entsprechenden AWS CLI Befehle Mount-Ziele und alle zusätzlichen optionalen Tags für das Dateisystem erstellen.

Die folgenden Beispiele verwenden `adminuser` als Werte für den Parameter `--profile`. Sie müssen ein entsprechendes Benutzerprofil verwenden, um Ihre Anmeldeinformationen anzugeben. Weitere Informationen finden Sie unter [Voraussetzungen für die Verwendung](#) von AWS CLI im AWS Command Line Interface Benutzerhandbuch.

- Um ein verschlüsseltes Dateisystem mit aktivierten automatischen Backups zu erstellen, verwenden Sie den `amazon create-file-system` EFS-CLI-Befehl (der entsprechende Vorgang ist [CreateFileSystem](#)), wie im Folgenden dargestellt.

```
aws efs create-file-system \  
--creation-token creation-token \  
--encrypted \  
--backup \  
--performance-mode generalPurpose \  
--throughput-mode elastic \  
--region aws-region \  
--tags Key=key,Value=value Key=key1,Value=value1 \  
--profile adminuser
```

Mit dem folgenden `create-file-system` Befehl wird beispielsweise ein Dateisystem erstellt, das den Elastic Throughput in der verwendet `us-west-2` AWS-Region. Der Befehl gibt `MyFirstFS` als Erstellungstoken an. Eine Liste der Orte, an AWS-Regionen denen Sie ein Amazon EFS-Dateisystem erstellen können, finden Sie unter [Amazon EFS-Endpunkte und Kontingente](#) in der Allgemeine Amazon Web Services-Referenz.

```
aws efs create-file-system \  
--creation-token MyFirstFS \  
--backup \  
--encrypted \  
--performance-mode generalPurpose \  
--throughput-mode elastic \  
--region us-west-2 \  
--tags Key=Name,Value="Test File System" Key=developer,Value=rhoward \  
--profile adminuser
```

Nach der erfolgreichen Erstellung des Dateisystems gibt Amazon EFS die Dateisystembeschreibung als JSON aus, wie im folgenden Beispiel gezeigt.

```
{
  "OwnerId": "123456789abcd",
  "CreationToken": "MyFirstFS",
  "Encrypted": true,
  "FileSystemId": "fs-c7a0456e",
  "CreationTime": 1422823614.0,
  "LifecycleState": "creating",
  "Name": "Test File System",
  "NumberOfMountTargets": 0,
  "SizeInBytes": {
    "Value": 6144,
    "ValueInIA": 0,
    "ValueInStandard": 6144,
    "ValueInArchive": 0
  },
  "PerformanceMode": "generalPurpose",
  "ThroughputMode": "elastic",
  "Tags": [
    {
      "Key": "Name",
      "Value": "Test File System"
    }
  ]
}
```

- Im folgenden Beispiel wird mithilfe der Eigenschaft ein Dateisystem erstellt, das den Bursting-Durchsatz in der us-west-2a Availability Zone verwendet. `availability-zone-name`

```
aws efs create-file-system \
--creation-token MyFirstFS \
--availability-zone-name us-west-2a \
--backup \
--encrypted \
--performance-mode generalPurpose \
--throughput-mode bursting \
--region us-west-2 \
--tags Key=Name,Value="Test File System" Key=developer,Value=rhoward \
--profile adminuser
```

Nach der erfolgreichen Erstellung des Dateisystems gibt Amazon EFS die Dateisystembeschreibung als JSON aus, wie im folgenden Beispiel gezeigt.

```
{
  "AvailabilityZoneId": "usw-az1",
  "AvailabilityZoneName": "us-west-2a",
  "OwnerId": "123456789abcd",
  "CreationToken": "MyFirstFS",
  "Encrypted": true,
  "FileSystemId": "fs-c7a0456e",
  "CreationTime": 1422823614.0,
  "LifecycleState": "creating",
  "Name": "Test File System",
  "NumberOfMountTargets": 0,
  "SizeInBytes": {
    "Value": 6144,
    "ValueInIA": 0,
    "ValueInStandard": 6144,
    "ValueInArchive": 0
  },
  "PerformanceMode": "generalPurpose",
  "ThroughputMode": "bursting",
  "Tags": [
    {
      "Key": "Name",
      "Value": "Test File System"
    }
  ]
}
```

Dazu bietet Amazon EFS den CLI-Befehl `describe-file-systems` (die entsprechende API-Operation lautet [DescribeFileSystems](#)), mit dem Sie eine Liste der Dateisysteme in Ihrem Konto abrufen können, wie nachfolgend gezeigt:

```
aws efs describe-file-systems \
--region aws-region \
--profile adminuser
```

Amazon EFS gibt eine Liste der Dateisysteme zurück, die Sie in der angegebenen Region AWS-Konto erstellt haben.

Löschen von EFS-Dateisystemen

Das Löschen eines Dateisystems ist ein endgültiger Vorgang, der nicht rückgängig gemacht werden kann. Das Dateisystem und alle darin enthaltenen Daten gehen dabei verloren. Alle Daten, die Sie in einem Dateisystem löschen, gehen endgültig verloren und können nicht wiederhergestellt werden. Wenn Benutzer Daten aus einem Dateisystem löschen, werden diese Daten sofort unbenutzbar. Die EFS-Force-Einstellung überschreibt die Daten auf letztendliche Art.

Note

Dateisysteme, die Teil einer Replikationskonfiguration sind, können nicht gelöscht werden. Sie müssen zuerst die Replikationskonfiguration löschen. Weitere Informationen finden Sie unter [Löschen von Replikationskonfigurationen](#).

Important

Bevor Sie ein Dateisystem löschen, sollten Sie immer den Dateisystem-Mount aufheben.

Verwenden der Konsole

So löschen Sie ein Dateisystem

1. Öffnen Sie die Amazon Elastic File System-Konsole unter <https://console.aws.amazon.com/efs/>.
2. Wählen Sie das Dateisystem, das Sie löschen möchten, aus der Liste der Dateisysteme aus.
3. Wählen Sie Löschen aus.
4. Geben Sie im Dialogfeld Dateisystem löschen die angezeigte Dateisystem-ID ein und klicken Sie auf Bestätigen, um den Löschvorgang zu bestätigen.

Die Konsole macht das Löschen des Dateisystems leichter. Sie löscht zuerst die zugehörigen Mountingziele und dann das Dateisystem.

Mit dem AWS CLI

Bevor Sie den AWS CLI Befehl zum Löschen eines Dateisystems verwenden können, müssen Sie alle Mount-Ziele und Access Points löschen, die für das Dateisystem erstellt wurden.

AWS CLI Beispielbefehle finden Sie unter [Schritt 4: Bereinigen](#).

Erstellen von Dateisystemrichtlinien

Sie können eine Dateisystemrichtlinie mithilfe der Amazon-EFS-Konsole oder mit der AWS CLI erstellen. Sie können eine Dateisystemrichtlinie auch programmgesteuert erstellen, indem Sie die Amazon EFS-API direkt verwenden AWS SDKs . Die Zeichenbeschränkung von EFS-Dateisystemrichtlinien liegt bei 20.000. Weitere Informationen zur Verwendung einer EFS-Dateisystemrichtlinie und Beispiele finden Sie unter [Verwendung von IAM zur Steuerung des Zugriffs auf Dateisysteme](#).

Note

Es kann mehrere Minuten dauern, bis Änderungen der Amazon-EFS-Dateisystemrichtlinien wirksam werden.

Verwenden der Konsole

1. Öffnen Sie die Amazon Elastic File System-Konsole unter <https://console.aws.amazon.com/efs/>.
2. Wählen Sie File Systems (Dateisysteme) aus.
3. Wählen Sie auf der Seite File systems (Dateisysteme) das Dateisystem aus, für das Sie eine Dateisystemrichtlinie bearbeiten oder erstellen möchten.
4. Wählen Sie Dateisystemrichtlinie und dann Bearbeiten aus.
5. In den Richtlinienoptionen können Sie eine beliebige Kombination der vorkonfigurierten Dateisystemrichtlinien auswählen:
 - Standardmäßig Root-Zugriff verhindern – Mit dieser Option wird `ClientRootAccess` aus der Gruppe der zulässigen EFS-Aktionen entfernt.
 - Standardmäßig nur Lesezugriff erzwingen – Mit dieser Option wird `ClientWriteAccess` aus der Gruppe der zulässigen EFS-Aktionen entfernt.
 - Anonymen Zugriff verhindern – Mit dieser Option wird `ClientMount` aus der Gruppe der zulässigen EFS-Aktionen entfernt.
 - Verschlüsselung während der Übertragung für alle Clients erzwingen – Mit dieser Option wird unverschlüsselten Clients der Zugriff verweigert.

Wenn Sie eine vorkonfigurierte Richtlinie auswählen, wird das Richtlinien-JSON-Objekt im Bereich des Richtlinien-Editors angezeigt.

6. Verwenden Sie Zusätzliche Berechtigungen gewähren, um weiteren IAM-Prinzipalen, einschließlich anderen, Dateisystemberechtigungen zu gewähren. AWS-Konto Wählen Sie Hinzufügen aus und geben Sie den Prinzipal-ARN der Entität ein, der Sie Berechtigungen gewähren. Wählen Sie die Berechtigungen aus, die Sie erteilen möchten. Die zusätzlichen Berechtigungen werden im Richtlinien-Editor angezeigt.
7. Sie können den Richtlinien-Editor verwenden, um eine vorkonfigurierte Richtlinie anzupassen oder Ihre eigene Dateisystemrichtlinie zu erstellen. Wenn Sie den Editor verwenden, sind die vorkonfigurierten Richtlinienoptionen nicht mehr verfügbar. Um die aktuelle Dateisystemrichtlinie zu löschen und mit der Erstellung einer neuen Richtlinie zu beginnen, wählen Sie Löschen aus.

Wenn Sie den Editor löschen, sind die vorkonfigurierten Richtlinien wieder verfügbar.

8. Nachdem Sie die Bearbeitung der Richtlinie abgeschlossen haben, wählen Sie Speichern aus.

Verwenden von AWS CLI

Im folgenden Beispiel erstellt der [put-file-system-policy](#) CLI-Befehl eine Dateisystemrichtlinie, die den angegebenen AWS-Konto schreibgeschützten Zugriff auf das EFS-Dateisystem ermöglicht. Der äquivalente API-Befehl lautet [PutFileSystemPolicy](#).

```
aws efs put-file-system-policy --file-system-id fs-01234567 --policy '{
  "Id": "1",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticfilesystem:ClientMount"
      ],
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:root"
      }
    }
  ]
}'
```

Erstellen von Zugriffspunkten

Sie können Amazon EFS-Zugriffspunkte mithilfe der AWS-Managementkonsole, der AWS Command Line Interface (AWS CLI) und der Amazon EFS-API und löschen SDKs. Sie können einen Zugangspunkt nicht mehr ändern, nachdem er einmal erstellt wurde. Ein Dateisystem kann maximal 10.000 Zugriffspunkte haben, sofern Sie keine Erhöhung beantragen.

Note

Wenn mehrere Anfragen zur Erstellung von Access Points auf demselben Dateisystem schnell hintereinander gesendet werden und sich das Dateisystem dem Limit für Zugriffspunkte nähert, kann es bei diesen Anfragen zu einer Drosselung der Antwort kommen. Dadurch wird sichergestellt, dass das Dateisystem das angegebene Kontingent für Zugangspunkte nicht überschreitet.

Weitere Hinweise zu EFS-Zugangspunkten finden Sie unter [Arbeiten mit Zugriffspunkten](#).

Verwenden der Konsole

1. Öffnen Sie die Amazon Elastic File System-Konsole unter <https://console.aws.amazon.com/efs/>.
2. Wählen Sie Zugangspunkte aus, um das Fenster Zugangspunkte zu öffnen.
3. Wählen Sie auf der Seite Zugangspunkt erstellen die Option Zugangspunkt erstellen aus.

Sie können die Seite Zugangspunkt erstellen auch öffnen, indem Sie Dateisysteme auswählen. Wählen Sie einen Dateisystemnamen oder eine Dateisystem-ID und dann Zugangspunkte und Zugangspunkt erstellen, um einen Zugangspunkt für dieses Dateisystem zu erstellen.

a. Geben Sie im Bereich Details die folgenden Informationen ein:

- Dateisystem – Geben Sie einen Dateisystemnamen oder eine ID ein und wählen Sie das passende Dateisystem aus. Sie können das Dateisystem auch aus der Liste auswählen, die angezeigt wird, wenn Sie das Eingabefeld auswählen.
- (Optional) Name – Geben Sie einen Namen für den Zugangspunkt ein.
- (Optional) Stammverzeichnispfad – Sie können ein Stammverzeichnis für den Zugangspunkt angeben. Das Standard-Stammverzeichnis für den Zugangspunkt ist /. Verwenden Sie das Format /foo/bar, um einen Stammverzeichnispfad einzugeben.

Weitere Informationen finden Sie unter [Erzwingen eines Stammverzeichnisses mit einem Zugangspunkt](#).

- b. (Optional) Im Bereich POSIX-Benutzer können Sie die vollständige POSIX-Identität angeben, die verwendet werden soll, um Benutzer- und Gruppeninformationen für alle Dateioperationen von NFS-Clients, die den Zugangspunkt verwenden, durchzusetzen. Weitere Informationen finden Sie unter [Erzwingen einer Benutzeridentität mithilfe eines Zugangspunkts](#).
- Benutzer-ID – Geben Sie die numerische POSIX-Benutzer-ID für den Benutzer ein.
 - Gruppen-ID – Geben Sie die numerische POSIX-Gruppen-ID für den Benutzer ein.
 - Sekundäre Gruppe IDs — Geben Sie eine optionale, durch Kommas getrennte Liste der sekundären Gruppe ein. IDs
- c. (Optional) Für Berechtigungen zum Erstellen des Stammverzeichnisses können Sie die Berechtigungen angeben, die verwendet werden sollen, wenn Amazon EFS den Stammverzeichnispfad erstellt, sofern angegeben, und das Stammverzeichnis noch nicht vorhanden ist. Weitere Informationen finden Sie unter [Erzwingen eines Stammverzeichnisses mit einem Zugangspunkt](#).

 Note

Wenn Sie keinen Besitz und keine Berechtigungen für das Stammverzeichnis angeben und das Stammverzeichnis noch nicht existiert, erstellt EFS das Stammverzeichnis nicht. Versuche, das Dateisystem mithilfe des Zugangspunkts zu mounten, schlagen fehl.

- Besitzerbenutzer-ID – Geben Sie die numerische POSIX-Benutzer-ID ein, die als Besitzer des Stammverzeichnisses verwendet werden soll.
 - Benutzergruppen-ID – Geben Sie die numerische POSIX-Gruppen-ID ein, die als Besitzergruppe des Stammverzeichnisses verwendet werden soll.
 - Berechtigungen – Geben Sie den Unix-Modus des Verzeichnisses ein. Eine allgemeine Konfiguration ist 755. Stellen Sie sicher, dass das Ausführungs-Bit für den Benutzer des Zugriffspunkts festgelegt ist, damit er mounten kann.
4. Wählen Sie Zugangspunkt erstellen aus, um den Zugangspunkt mit dieser Konfiguration zu erstellen.

Verwenden Sie den AWS CLI

Im folgenden Beispiel erstellt der `create-access-point`-CLI-Befehl einen Zugangspunkt für das EFS-Dateisystem. Der äquivalente API-Befehl lautet [CreateAccessPoint](#).

```
aws efs create-access-point --file-system-id fs-abcdef0123456789a --client-token
010102020-3 \
--root-directory "Path=/efs/mobileapp/
east,CreationInfo={OwnerId=0,OwnerGid=11,Permissions=775}" \
--posix-user "Uid=22,Gid=4" \
--tags Key=Name,Value=east-users
```

Wenn die Anfrage erfolgreich ist, antwortet die CLI mit der Beschreibung des Zugangspunkts.

```
{
  "ClientToken": "010102020-3",
  "Name": "east-users",
  "AccessPointId": "fsap-abcd1234ef5678901",
  "AccessPointArn": "arn:aws:elasticfilesystem:us-east-2:111122223333:access-point/
fsap-abcd1234ef5678901",
  "FileSystemId": "fs-01234567",
  "LifecycleState": "creating",
  "OwnerId": "111122223333",
  "PosixUser": {
    "Gid": 4,
    "Uid": 22
  },
  "RootDirectory": {
    "CreationInfo": {
      "OwnerGid": 0,
      "OwnerId": 11,
      "Permissions": "775"
    },
    "Path": "/efs/mobileapp/east",
  },
  "Tags": []
}
```

Note

Wenn mehrere Anfragen zum Erstellen von Access Points auf demselben Dateisystem schnell hintereinander gesendet werden und das Dateisystem sich dem Limit für

Zugriffspunkte nähert, kann es bei diesen Anfragen zu einer Drosselung der Antwort kommen. Dadurch wird sichergestellt, dass das Dateisystem das angegebene Kontingent für Zugangspunkte nicht überschreitet.

Access Points löschen

Wenn Sie einen Zugangspunkt löschen, verlieren alle Clients, die den Zugangspunkt verwenden, den Zugang auf das Amazon-EFS-Dateisystem, für das er konfiguriert ist.

Verwenden der Konsole

1. Öffnen Sie die Amazon Elastic File System-Konsole unter <https://console.aws.amazon.com/efs/>.
2. Wählen Sie im linken Navigationsbereich Zugangspunkte aus, um die Seite Zugangspunkte zu öffnen.
3. Wählen Sie den Zugangspunkt aus, der gelöscht werden soll.
4. Wählen Sie Löschen aus.
5. Wählen Sie Bestätigen aus, um die Aktion zu bestätigen und den Zugangspunkt zu löschen.

Mit dem AWS CLI

Im folgenden Beispiel löscht der `delete-access-point`-CLI-Befehl den angegebenen Zugangspunkt. Der äquivalente API-Befehl lautet [DeleteAccessPoint](#). Wenn der Befehl erfolgreich ist, gibt der Service eine HTTP-204-Antwort mit leerem HTTP-Textinhalt zurück.

```
aws efs delete-access-point --access-point-id fsap-092e9f80b3fb5e6f3 --client-token 010102020-3
```

Taggen von EFS-Ressourcen

Um Ihnen bei der Verwaltung Ihrer EFS-Ressourcen zu helfen, können Sie jeder Ressource Ihre eigenen Metadaten in Form von Tags zuweisen. Mithilfe von Tags können Sie Ihre AWS Ressourcen auf unterschiedliche Weise kategorisieren, z. B. nach Zweck, Eigentümer oder Umgebung. Diese Kategorisierung ist nützlich, wenn Sie viele Ressourcen desselben Typs haben — In diesem Fall können Sie schnell bestimmte Ressourcen basierend auf den zugewiesenen Tags (Markierungen) bestimmen. In diesem Thema werden Tags (Markierungen) und deren Erstellung beschrieben.

Grundlagen zu Tags (Markierungen)

Ein Tag ist eine Bezeichnung, die Sie einer AWS Ressource zuweisen. Jeder Tag (Markierung) besteht aus einem Schlüssel und einem optionalen Wert, beides können Sie bestimmen.

Mithilfe von Tags können Sie Ihre AWS Ressourcen auf unterschiedliche Weise kategorisieren, z. B. nach Zweck, Eigentümer oder Umgebung. Sie können zum Beispiel eine Reihe von Tags für die Amazon-EFS-Dateisysteme Ihres Kontos definieren, mit denen Sie den Besitzer jedes Dateisystems verfolgen können.

Wir empfehlen die Verwendung von Tag (Markierung)-Schlüsseln, die die Anforderungen der jeweiligen Ressourcentypen erfüllen. Die Verwendung einheitlicher Tag-Schlüssel vereinfacht das Verwalten der -Ressourcen. Sie können die Ressourcen auf Grundlage der hinzugefügten Tags (Markierungen) filtern und danach suchen.

Tags (Markierungen) haben keine semantische Bedeutung für Amazon EFS und werden ausschließlich als Zeichenfolgen interpretiert. Außerdem werden Tags (Markierungen) nicht automatisch Ihren Ressourcen zugewiesen. Sie können Tag (Markierung)-Schlüssel und -Werte bearbeiten und Tags (Markierungen) jederzeit von einer Ressource entfernen. Sie können den Wert eines Tags (Markierung) zwar auf eine leere Zeichenfolge, jedoch nicht null festlegen. Wenn Sie ein Tag (Markierung) mit demselben Schlüssel wie ein vorhandener Tag (Markierung) für die Ressource hinzufügen, wird der alte Wert mit dem neuen überschrieben. Wenn Sie eine Ressource löschen, werden alle Tags (Markierungen) der Ressource ebenfalls gelöscht.

Tag-Einschränkungen

Die folgenden grundlegenden Einschränkungen gelten für Tags (Markierungen):

- Maximale Anzahl von Tags (Markierungen) pro Ressource: 50
- Jeder Tag (Markierung) muss für jede Ressource eindeutig sein. Jeder Tag (Markierung) kann nur einen Wert haben.
- Maximale Schlüssellänge: 128 Unicode-Zeichen in UTF-8
- Maximale Wertlänge — 256 Unicode-Zeichen in UTF-8
- Amazon EFS lässt beliebige Zeichen in seinen Tags (Markierungen) zu. Allerdings kann es in anderen -Services mehr Einschränkungen geben. Erlaubte Zeichen in Services sind: Buchstaben, Zahlen und Leerzeichen, die in UTF-8 darstellbar sind, und die folgenden Sonderzeichen: + - = . _ : / @.
- Bei Tag (Markierung)-Schlüsseln und -Werten muss die Groß-/Kleinschreibung beachtet werden.

- Das `aws:` Präfix ist für AWS die Verwendung reserviert. Wenn der Tag (Markierung) über einen Tag (Markierung)-Schlüssel mit diesem Präfix verfügt, können Sie den Schlüssel oder Wert des Tags (Markierung) nicht bearbeiten oder löschen. Tags (Markierungen) mit dem Präfix `aws:` werden nicht als Ihre Tags (Markierungen) pro Ressourcenlimit angerechnet.

Sie können Ressourcen nicht allein auf Grundlage ihrer Tags (Markierungen) aktualisieren oder löschen. Sie müssen den Ressourcenbezeichner angeben. Um Dateisysteme zu löschen, die Sie mit dem Tag (Markierung)-Schlüssel `DeleteMe` markiert haben, müssen Sie die `DeleteFileSystem`-Aktion mit den Ressourcenbezeichnern des Dateisystems verwenden, z. B. `fs-1234567890abcdef0`.

Wenn Sie öffentliche oder gemeinsam genutzte Ressourcen markieren, sind die von Ihnen zugewiesenen Tags nur für Ihr AWS-Konto verfügbar. Kein anderer AWS-Konto wird Zugriff auf diese Tags haben. Für die tagbasierte Zugriffskontrolle auf gemeinsam genutzte Ressourcen AWS-Konto muss jede Ressource ihren eigenen Satz von Tags zuweisen, um den Zugriff auf die Ressource zu kontrollieren.

Sie können Amazon-EFS-Dateisystem- und Zugangspunktressourcen markieren.

Verwenden von Tags für die Zugriffskontrolle

Sie können Tags verwenden, um den Zugriff auf Amazon-EFS-Ressourcen zu steuern und die attributbasierte Zugriffskontrolle (ABAC) zu implementieren.

Note

Die Replikation unterstützt die Verwendung von Tags für die attributbasierte Zugriffskontrolle (ABAC) nicht.

Markieren Ihrer -Ressourcen mit Tags (Markierungen)

Sie können Amazon-EFS-Dateisystem- und Zugangspunktressourcen markieren, die bereits in Ihrem Konto bestehen.

Verwenden der Konsole

- Sie können die Amazon-EFS-Konsole verwenden, um Tags auf vorhandene Ressourcen anzuwenden, indem Sie die Registerkarte Tags auf dem Bildschirm mit den Ressourcendetails

verwenden. In der Amazon-EFS-Konsole können Sie Tags für eine Ressource angeben, wenn Sie die Ressource erstellen. Beispielsweise können Sie ein Tag mit dem Schlüssel von Name und einem von Ihnen angegebenen Wert hinzufügen.

In den meisten Fällen wendet die Konsole Tags (Markierungen) direkt nach dem Erstellen der Ressource an und nicht während des Erstellens. Die Konsole strukturiert Ressourcen gemäß des Name-Tags. Allerdings hat der Tag keine semantische Bedeutung für den Amazon-EFS-Service.

Mit dem AWS CLI

- Wenn Sie die Amazon EFS-API, das AWS CLI oder ein AWS SDK verwenden, können Sie die `TagResource` EFS-API-Aktion verwenden, um Tags auf vorhandene Ressourcen anzuwenden. Zudem können Sie mit einigen Aktionen zur Ressourcenerstellung Tags beim Erstellen einer Ressource angeben.

Die AWS CLI Befehle für die Verwaltung von Tags und die entsprechenden Amazon EFS-API-Aktionen sind in der folgenden Tabelle aufgeführt.

CLI-Befehl	Description	Äquivalente API-Operation
tag-resource	Neue Tags hinzufügen oder vorhandene Tags aktualisieren	TagResource
list-tags-for-resource	Vorhandene Tags abrufen	ListTagsForResource
untag-resource	Vorhandene Tags löschen	UntagResource

Tutorial: Schreibbare Unterverzeichnisse pro Benutzer erstellen

Nachdem Sie ein EFS-Dateisystem erstellt und es lokal auf Ihrer Amazon EC2 (EC2) -Instance bereitgestellt haben, wird ein leeres Verzeichnis mit dem Namen verfügbar gemacht. *file system root* Ein häufiger Anwendungsfall für dieses Dateisystem-Stammverzeichnis besteht darin, für jeden Benutzer, den Sie auf der EC2-Instance erstellen, ein „beschreibbares“ Unterverzeichnis zu erstellen und das Unterverzeichnis im Home-Verzeichnis des Benutzers zu mounten. Alle Dateien und Unterverzeichnisse, die der Benutzer in seinem Home-Verzeichnis erstellt, werden dann im EFS-Dateisystem erstellt.

 Note

Sie können der [Erste Schritte](#) Übung folgen, um ein EFS-Dateisystem auf Ihrer EC2-Instance zu erstellen und zu mounten.

In den folgenden Schritten erstellen Sie einen Benutzer, erstellen ein Unterverzeichnis für den Benutzer, machen den Benutzer zum Eigentümer des Unterverzeichnisses und mounten dann das Amazon EFS-Unterverzeichnis im Home-Verzeichnis des Benutzers.

1. Erstellen des Benutzers „Mike“:

- Melden Sie sich bei Ihrer EC2-Instance an. Mit Root-Rechten (in diesem Fall mit dem `sudo` Befehl) erstellen Sie den Benutzer und weisen ihm ein Passwort zu.

Mit dem folgenden Befehl wird beispielsweise der Benutzer `mike` erstellt.

```
$ sudo useradd -c "Mike Smith" mike
$ sudo passwd mike
```

Für den Benutzer wird auch ein Home-Verzeichnis erstellt. Beispiel, `/home/mike`.

2. Erstellen Sie unter ein Unterverzeichnis `EFSroot` für den Benutzer.

Mit dem folgenden Befehl wird beispielsweise ein Unterverzeichnis unter `mike` erstellt. `EFSroot`

```
$ sudo mkdir /EFSroot/mike
```

Sie müssen es durch Ihren lokalen Verzeichnisnamen `EFSroot` ersetzen.

- #### 3. Der Root-Benutzer und die Root-Gruppe sind die Eigentümer des Unterverzeichnisses (Sie können dies mit dem `ls -l` Befehl überprüfen). Um dem Benutzer die vollen Rechte für dieses Unterverzeichnis zu gewähren, gewähren Sie dem Benutzer die Inhaberschaft des Verzeichnisses.

Beispiel:

```
$ sudo chown mike:mike /EFSroot/mike
```

- #### 4. Verwenden Sie den `mount` Befehl, um das Unterverzeichnis in das Home-Verzeichnis des Benutzers einzuhängen.

Beispiel:

```
$ sudo mount -t nfs -o  
nfsvers=4.1,rsiz=1048576,wsiz=1048576,hard,timeo=600,retrans=2,noresvport mount-  
target-DNS:/mike /home/mike
```

Die *mount-target-DNS* Adresse identifiziert den Remote-EFS-Dateisystemstamm.

Wenn Sie die Bereitstellung dieses Mount-Ziels aufheben, kann der Benutzer ohne erneutes Mounten nicht auf das Verzeichnis zugreifen, wofür Root-Rechte erforderlich sind.

Mounting von EFS-Dateisystemen

Zum Mounten von EFS-Dateisystemen empfehlen wir die Verwendung des EFS-Mount-Helpers. Der EFS-Mount-Helper hilft Ihnen beim Mounten Ihrer EFS-Dateisysteme auf EC2-Linux- und Mac-Instances, auf denen die unterstützten Distributionen ausgeführt werden. Der Mount-Helper ist Teil der Open-Source-Toolsammlung, die bei der Installation des Amazon EFS-Clients (`amazon-efs-utils`) installiert wird. Weitere Informationen zum Amazon EFS-Client und den unterstützten Distributionen finden Sie unter [Den Amazon EFS-Client installieren](#).

Alternativ können Sie EFS-Dateisysteme mithilfe des standardmäßigen Linux-NFS-Clients manuell mounten. Amazon EFS unterstützt beim Mounten Ihrer Dateisysteme auf Amazon-EC2-Instances die Network File System-Versionen 4.0 und 4.1 (NFSv4) und NFSv4.0-Protokolle.

Darüber hinaus können Sie den EFS-Mount-Helper oder NFS verwenden, um eine EC2-Instance so zu konfigurieren, dass beim Start der Instance automatisch ein EFS-Dateisystem bereitgestellt wird.

Topics

- [Überlegungen zur Installation für Linux](#)
- [Mounten von EFS-Dateisystemen mit dem EFS-Mount-Helper](#)
- [Verwenden des Netzwerkdateisystems zum Mounten von EFS-Dateisystemen](#)
- [Automatisches Mounten von EFS-Dateisystemen](#)
- [Aufheben des Mountings von Dateisystemen](#)
- [Tutorial: Erstellen Sie ein EFS-Dateisystem und mounten Sie es auf einer EC2-Instance mit dem AWS CLI](#)
- [Tutorial: Mounten mit lokalen Linux-Clients](#)
- [Tutorial: Mounten Sie ein Dateisystem von einer anderen VPC](#)
- [Beheben von Mountingproblemen](#)

Überlegungen zur Installation für Linux

Wir empfehlen die folgenden Werte für die Mountingoptionen unter Linux:

- `rsize=1048576` – Legt die maximale Byteanzahl der Daten fest, die der NFS-Client für jede Netzwerk-READ-Anforderung erhalten kann. Dieser Wert gilt beim Lesen von Daten aus einer

Datei in einem EFS-Dateisystem. Zur Vermeidung von Leistungseinbußen empfehlen wir die Verwendung der maximal möglichen Größe (bis zu 1048576).

- `wsize=1048576` – Legt die maximale Byteanzahl der Daten fest, die der NFS-Client für jede Netzwerk-WRITE-Anforderung senden kann. Dieser Wert gilt beim Schreiben von Daten in eine Datei in einem EFS-Dateisystem. Zur Vermeidung von Leistungseinbußen empfehlen wir die Verwendung der maximal möglichen Größe (bis zu 1048576).
- `hard` – Legt das Wiederherstellungsverhalten des NFS-Clients nach dem Timeout einer NFS-Anforderung fest, damit NFS-Anforderungen so lange wiederholt werden, bis der Server antwortet. Zur Sicherstellung der Datenintegrität wird die Verwendung der dauerhaften Mountingoption (`hard`) empfohlen. Wenn Sie ein `soft`-Mount verwenden, legen Sie den `timeo`-Parameter auf mindestens 150 Zehntelsekunden (15 Sekunden) fest. Dadurch wird das Risiko einer Datenbeschädigung verringert, die bei `Soft`-Mounts inhärent ist.
- `timeo=600` – Legt den Timeout-Wert, der angibt, wie lange der NFS-Client auf eine Antwort wartet, bis er eine NFS-Anforderung wiederholt, auf 600 Zehntelsekunden (60 Sekunden) fest. Wenn Sie den Timeout-Parameter (`timeo`) ändern müssen, empfehlen wir, dass Sie einen Wert von mindestens 150, entsprechend 15 Sekunden, verwenden. Dadurch wird eine verringerte Leistung vermieden.
- `retrans=2` – Legt die Anzahl der Anforderungswiederholungen eines NFS-Clients vor dem Versuch einer weiteren Wiederherstellungsaktion auf 2 fest.
- `noresvport` – Teilt dem NFS-Client mit, einen neuen nicht privilegierten Quellanschluss für Transmission Control Protocol (TCP) zu verwenden, wenn erneut eine Netzwerkverbindung eingerichtet wird. Dadurch wird der ununterbrochene Zugriff des EFS-Dateisystems nach einem Netzwerkwiederherstellungsereignis sichergestellt.
- `_netdev` – Sofern in `/etc/fstab` vorhanden, wird der Client an dem Versuch gehindert, das EFS-Dateisystem zu mounten, bis das Netzwerk aktiviert wurde.

Vermeiden Sie es generell, jegliche anderen Mounting-Optionen zu verwenden, die sich von den Standardoptionen unterscheiden, denn dies kann zu Leistungseinbußen und anderen Problemen führen. Wenn Sie die vorgenannten Standardwerte nicht verwenden, achten Sie auf Folgendes:

- Änderungen der Puffergröße für Lese- oder Schreibvorgänge oder die Deaktivierung der Attributzwischenspeicherung können zu einer Leistungsverringerung führen.
- Amazon EFS ignoriert Quellports. Wenn Sie Amazon-EFS-Quellports ändern, hat dies keinerlei Auswirkungen.

- Amazon EFS unterstützt keine der Kerberos-Sicherheitsvarianten. Beispielsweise führt der folgende Mounting-Befehl zu einem Fehler.

```
$ mount -t nfs4 -o krb5p <DNS_NAME>:/ /efs/
```

- Mounten Sie Ihr System möglichst mit dessen DNS-Namen. Amazon EFS löst diesen Namen in die IP-Adresse des EFS-Mount-Ziels in derselben Availability Zone wie Ihre Amazon EC2 EC2-Instance auf, ohne externe Ressourcen aufzurufen. Wenn Sie ein Mountingziel in einer Availability Zone mounten, die sich von der EC2-Instance unterscheidet, werden Standard-EC2-Gebühren für Daten erhoben, die zwischen Availability Zones übertragen werden. Sie bemerken bei Dateisystemvorgängen möglicherweise auch erhöhte Latenzen.
- Weitere Bereitstellungsoptionen und detaillierte Erläuterungen der Standardeinstellungen finden Sie in der Linux-Dokumentation.

Note

Wenn Ihre EC2-Instance unabhängig vom Status des gemounteten EFS-Dateisystems starten muss, fügen Sie die Option `nofail` zum Eintrag Ihres Dateisystems in Ihrer Datei `/etc/fstab` hinzu.

Mounten von EFS-Dateisystemen mit dem EFS-Mount-Helper

Nachdem Sie den Amazon EFS-Client (`amazon-efs-utils`) installiert haben, können Sie den EFS-Mount-Helper verwenden, um EFS-Dateisysteme auf Ihren EC2-Linux- und Mac-Instances zu mounten, auf denen [unterstützte Distributionen](#) ausgeführt werden. Amazon EFS unterstützt das Mounten von EC2-Windows-Instances nicht.

Important

Wir empfehlen, immer die aktuelle Version von `amazon-efs-utils` zu verwenden, um eine erfolgreiche Installation sicherzustellen. Beispielsweise unterstützen Versionen `amazon-efs-utils` vor 2.3 das Mounten mit IPv6 Adressen nicht.

Beim Mounten eines Dateisystems definiert der Mount-Helper einen neuen Netzwerkdateisystemtyp `namensefs`, der vollständig kompatibel mit dem `mount` Standardbefehl in Linux ist. Der Mount-

Helper unterstützt auch das automatische Mounten eines EFS-Dateisystems beim Start der Instanz mithilfe von Einträgen in der `/etc/fstab` Konfigurationsdatei auf EC2-Linux-Instances.

 Warning

Verwenden Sie beim automatischen Mounting Ihres Dateisystems die Option `_netdev`, um es als Netzwerkdateisystem zu identifizieren. Wenn `_netdev` fehlt, reagiert die EC2-Instance möglicherweise nicht mehr. Der Grund dafür ist, dass zuerst das Netzwerk auf der Datenverarbeitungs-Instance gestartet worden sein muss. Die Netzwerkdateisysteme müssen danach initialisiert werden. Weitere Informationen finden Sie unter [Automatisches Mounting schlägt fehl und die Instance reagiert nicht](#).

Sie können ein Dateisystem mounten, indem Sie eine der folgenden Eigenschaften angeben:

- DNS-Name des Dateisystems – Wenn Sie den DNS-Namen des Dateisystems verwenden und die Mountinghilfe ihn nicht auflösen kann, z. B. wenn Sie ein Dateisystem in einer anderen VPC mounten, wird auf die IP-Adresse des Mountingziels zurückgegriffen. Weitere Informationen finden Sie unter [Mounten von EFS-Dateisystemen von einer anderen AWS-Konto oder VPC](#).
- Dateisystem-ID – Wenn Sie die Dateisystem-ID verwenden, löst die Mountinghilfe sie in die lokale IP-Adresse der Elastic-Network-Schnittstelle (ENI) des Mountingziels auf, ohne externe Ressourcen aufzurufen.
- IP-Adresse des Mountingziels – Sie können die IP-Adresse eines der Mountingziele des Dateisystems verwenden.

Die Werte aller dieser Eigenschaften finden Sie in der Amazon-EFS-Konsole. Der DNS-Name des Dateisystems befindet sich auf dem Bildschirm Anhängen.

Wenn die Verschlüsselung von Daten während der Übertragung als Mount-Option für Ihr EFS-Dateisystem deklariert ist, initialisiert der Mount-Helper einen `stunnel` Client-Prozess und es wird ein Supervisor-Prozess aufgerufen. `amazon-efs-mount-watchdog` Der `amazon-efs-mount-watchdog`-Prozess überwacht den Zustand von TLS-Mounts und wird automatisch gestartet, wenn ein EFS-Dateisystem zum ersten Mal über TLS gemountet wird. Wenn der Client unter Linux ausgeführt wird, wird dieser Prozess je nach Linux-Distribution entweder von `upstart` oder `systemd` verwaltet. Für Clients, die auf einem unterstützten macOS ausgeführt werden, wird der Prozess von `launchd` verwaltet.

Stunnel ist ein Open-Source-Netzwerk-Relay für unterschiedliche Einsatzzwecke. Der Client-stunnel-Prozess überwacht einen lokalen Port auf eingehenden Datenverkehr, und die Mountinghilfe leitet NFS-Client-Datenverkehr an diesen lokalen Port um.

Die Mountinghilfe verwendet TLS Version 1.2 für die Kommunikation mit dem Dateisystem. Für die Verwendung von TLS sind Zertifikate erforderlich, die von einer vertrauenswürdigen Amazon-Zertifizierungsstelle signiert sind. Weitere Informationen zur Funktionsweise von Verschlüsselung finden Sie unter [Datenverschlüsselung in Amazon EFS](#).

Topics

- [Von EFS Mount Helper verwendete Mount-Einstellungen](#)
- [Abrufen von Support-Protokollen](#)
- [Voraussetzungen für die Verwendung der EFS-Mountinghilfe](#)
- [Mounten auf EC2-Linux-Instances mit dem EFS-Mount-Helper](#)
- [Mounten auf EC2-Mac-Instances mit dem EFS-Mount-Helper](#)
- [Mounten von EFS-Dateisystemen von einem anderen AWS-Region](#)
- [Mounting von One-Zone-Dateisystemen](#)
- [Mounting mit IAM-Autorisierung](#)
- [Mounting mit EFS-Zugangspunkten](#)
- [Mounten von EFS auf mehrere EC2-Instances](#)
- [Mounten von EFS-Dateisystemen von einer anderen AWS-Konto oder VPC](#)

Von EFS Mount Helper verwendete Mount-Einstellungen

Der Amazon-EFS-Mountinghilfe-Client verwendet die folgenden Mountingoptionen, die für Amazon EFS optimiert wurden:

- `nfsvers=4.1` – Wird beim Mounting auf EC2-Linux-Instances verwendet.

`nfsvers=4.0` – Wird beim Mounting auf unterstützten EC2-Mac-Instances verwendet, auf denen macOS Big Sur, Monterey und Ventura ausgeführt werden.
- `rsize=1048576` – Legt die maximale Byteanzahl der Daten, die der NFS-Client für jede Netzwerk-READ-Anforderung erhalten kann, auf die größte verfügbare Anzahl 1048576 fest, um einen Leistungsabfall zu vermeiden.

- `wsiz=1048576` – Legt die maximale Byteanzahl der Daten, die der NFS-Client für jede Netzwerk-WRITE-Anforderung senden kann, auf die größte verfügbare Anzahl 1048576 fest, um einen Leistungsabfall zu vermeiden.
- `hard` – Legt das Wiederherstellungsverhalten des NFS-Clients nach dem Timeout einer NFS-Anforderung fest, damit NFS-Anforderungen so lange wiederholt werden, bis der Server antwortet, um Datenintegrität zu gewährleisten.
- `timeo=600` – Legt den Timeout-Wert, der angibt, wie lange der NFS-Client auf eine Antwort wartet, bis er eine NFS-Anforderung wiederholt, auf 600 Zehntelsekunden (60 Sekunden) fest, um einen Leistungsabfall zu vermeiden.
- `retrans=2` – Legt die Anzahl der Anforderungswiederholungen eines NFS-Clients vor dem Versuch einer weiteren Wiederherstellungsaktion auf 2 fest.
- `noresvport` – Teilt dem NFS-Client mit, einen neuen nicht privilegierten Quellanschluss für Transmission Control Protocol (TCP) zu verwenden, wenn erneut eine Netzwerkverbindung eingerichtet wird. Mit der Option `noresvport` können Sie sicherstellen, dass das EFS-Dateisystem nach einer erneuten Verbindung oder einem Netzwerkwiederherstellungsereignis ununterbrochen verfügbar ist.
- `mountport=2049` – Wird nur beim Mounting auf EC2-Mac-Instances verwendet, auf denen macOS Big Sur, Monterey und Ventura ausgeführt werden.

Abrufen von Support-Protokollen

Der EFS-Mount-Helper verfügt über eine integrierte Protokollierung für Ihr EFS-Dateisystem. Sie können diese Protokolle zur Fehlerbehebung an den AWS Support weitergeben. Sie können die unter `/var/log/amazon/efs` auf den Clients gespeicherten Protokolle mithilfe der EFS-Mountinghilfe finden. Diese Protokolle sind für die EFS-Mountinghilfe, den Stunnel-Prozess (standardmäßig deaktiviert) sowie für den `amazon-efs-mount-watchdog`-Prozess zur Überwachung des Stunnel-Prozesses.

Note

Der `amazon-efs-mount-watchdog` Prozess stellt sicher, dass der Stunnel-Prozess jedes Mounts ausgeführt wird, und stoppt den Stunnel-Prozess, wenn das EFS-Dateisystem unmountet wird. Wenn der Stunnel-Prozess aus irgendeinem Grund unerwartet beendet wird, wird er vom Watchdog-Prozess neu gestartet.

Sie können die Protokollkonfiguration in `/etc/amazon/efs/efs-utils.conf` ändern. Damit alle Protokolländerungen wirksam werden, müssen Sie das Dateisystem mithilfe des EFS-Mount-Helpers unmounten und erneut mounten. Die Protokollkapazität für die Mountinghilfe und Watchdog-Protokolle beträgt 20 MiB. Protokolle für den Stunnel-Prozess sind standardmäßig deaktiviert.

Important

Sie können die Protokolle für den Stunnel-Prozess aktivieren. Dies kann jedoch erheblichen Speicherplatz auf Ihrem Dateisystem beanspruchen.

Voraussetzungen für die Verwendung der EFS-Mountinghilfe

Mit dem Amazon EFS-Mount-Helper können Sie ein EFS-Dateisystem auf einer Amazon EC2 EC2-Instance mounten. Damit Sie die Mountinghilfe verwenden können, benötigen Sie Folgendes:

- Dateisystem-ID des zu mountenden Dateisystems – Die Mountinghilfe löst die Dateisystem-ID in die lokale IP-Adresse der Elastic-Network-Schnittstelle (ENI) des Mountingziels auf, ohne externe Ressourcen aufzurufen.
- Ein EFS-Mount-Ziel — Sie erstellen Mount-Ziele in Ihrer Virtual Private Cloud (VPC). Wenn Sie Ihr Dateisystem in der Konsole mit den vom Dienst empfohlenen Einstellungen erstellen, wird in jeder Availability Zone, in der sich das Dateisystem befindet AWS-Region , ein Mount-Ziel erstellt. Anweisungen zur Erstellung von Mountingzielen finden Sie unter [Verwalten der Mountingziele](#).

Note

Wir empfehlen, 60 Sekunden zu warten, nachdem der Lebenszyklusstatus des neu erstellten Mountingziels Verfügbar lautet, bevor Sie das Dateisystem über DNS mounten. Durch diese Wartezeit können sich die DNS-Einträge vollständig in dem Bereich ausbreiten, AWS-Region an dem sich das Dateisystem befindet.

Wenn Sie ein Mountingziel in einer Availability Zone mounten, die sich von der EC2-Instance unterscheidet, werden Standard-EC2-Gebühren für Daten erhoben, die zwischen Availability Zones übertragen werden. Sie bemerken bei Dateisystemvorgängen möglicherweise auch erhöhte Latenzen.

- So mounten Sie One-Zone-Dateisysteme aus einer anderen Availability Zone:

- Name der Availability Zone des Dateisystems – Wenn Sie ein EFS One-Zone-Dateisystem mounten, das sich in einer anderen Availability Zone als die EC2-Instance befindet.
- DNS-Name des Mountingziels – Alternativ können Sie anstelle der Availability Zone den DNS-Namen des Mountingziels angeben.
- Eine EC2-Instance, auf der eine der unterstützten Linux- oder macOS-Distributionen ausgeführt wird — Die unterstützten Distributionen für das Mounten Ihres Dateisystems mit dem Mount Helper sind die folgenden:
 - Amazon Linux 2
 - Amazon Linux 2023
 - Amazon Linux 2017.09 und neuer
 - macOS Big Sur
 - Red Hat Enterprise Linux (und Derivate wie z. B. CentOS), Version 7 und höher
 - Ubuntu 16.04 LTS und höher

 Note

EC2 Mac-Instances, auf denen macOS Big Sur ausgeführt wird, unterstützen nur NFS 4.0.

- Der EFS-Mount-Helper ist auf der EC2-Instance installiert — Der Mount-Helper ist ein Tool im `amazon-efs-utils` Paket der Dienstprogramme. Hinweise zur Installation finden Sie `amazon-efs-utils` unter [Den Amazon EFS-Client installieren](#).
- Die EC2-Instance befindet sich in einer VPC – Die EC2-Instance, die eine Verbindung herstellt, muss sich in einer Virtual Private Cloud (VPC) befinden, die auf dem Amazon-VPC-Service basiert. Es muss auch für die Verwendung des DNS-Servers konfiguriert werden, der von bereitgestellt wird AWS. Informationen zum Amazon DNS-Server finden Sie unter [DHCP-Optionssätze in Amazon VPC](#) im Amazon VPC-Benutzerhandbuch.
- Für die VPC sind DNS-Hostnamen aktiviert – Für die VPC der verbindenden EC2-Instance müssen DNS-Hostnamen aktiviert sein. Weitere Informationen finden Sie unter [DNS-Attribute für Ihre VPC](#) im Amazon VPC-Benutzerhandbuch.
- Für unterschiedliche EC2-Instances und Dateisysteme AWS-Regionen — Wenn sich die EC2-Instance und das Dateisystem, das Sie mounten, auf unterschiedlichen Ebenen befinden AWS-Regionen, müssen Sie die `region` Eigenschaft in der Datei bearbeiten. `efs-utils.conf` Weitere Informationen finden Sie unter [Mounten von EFS-Dateisystemen von einem anderen AWS-Region](#).

Mounten auf EC2-Linux-Instances mit dem EFS-Mount-Helper

Dieser Vorgang erfordert die folgenden Voraussetzungen:

- Sie haben das `amazon-efs-utils` Paket auf der Amazon EC2 EC2-Instance installiert. Weitere Informationen finden Sie unter [Manuelles Installieren des Amazon-EFS-Clients](#).
- Das Dateisystem, das Sie gerade erstellt haben, verfügt über Mountingziele. Weitere Informationen finden Sie unter [Verwalten der Mountingziele](#).

So mounten Sie Ihr EFS-Dateisystem mit dem Mount Helper auf EC2-Linux-Instances

1. Öffnen Sie über Secure Shell (SSH) ein Terminal für die EC2-Instance und melden Sie sich mit dem entsprechenden Benutzernamen an. Weitere Informationen finden Sie unter [Mit Ihrer EC2-Instance verbinden](#) im Amazon-EC2-Benutzerhandbuch.
2. Erstellen Sie mit dem folgenden Befehl ein Verzeichnis `efs`, das Sie als Mountingpunkt für das Dateisystem verwenden:

```
sudo mkdir efs
```

3. Führen Sie einen der folgenden Befehle aus, um das Dateisystem zu mounten.

Note

Wenn sich die EC2-Instance und das Dateisystem, das Sie mounten, in unterschiedlichen AWS-Regionen befinden, finden Sie unter [Mounten von EFS-Dateisystemen von einem anderen AWS-Region](#) weitere Informationen zum Bearbeiten der Eigenschaft `region` in der Datei `efs-utils.conf`.

- So mounten Sie das Dateisystem mithilfe der System-ID:

```
sudo mount -t efs file-system-id efs-mount-point/
```

Verwenden Sie die ID des Dateisystems, an dessen Stelle *file-system-id* und an dessen Stelle Sie das Dateisystem `efs` mounten. *efs-mount-point*

```
sudo mount -t efs fs-abcd123456789ef0 efs/
```

Wenn Sie die Verschlüsselung von Daten bei der Übertragung verwenden möchten, können Sie das Dateisystem auch mit folgendem Befehl mounten.

```
sudo mount -t efs -o tls fs-abcd123456789ef0:/ efs/
```

- So mounten Sie das Dateisystem unter Verwendung des DNS-Namens:

```
sudo mount -t efs -o tls file-system-dns-name efs-mount-point/
```

```
sudo mount -t efs -o tls fs-abcd123456789ef0.efs.us-east-2.amazonaws.com efs/
```

- So mounten Sie das Dateisystem mithilfe der IP-Adresse des Mountingziels:

```
sudo mount -t efs -o tls,mounttargetip=mount-target-ip file-system-id efs-mount-point/
```

```
sudo mount -t efs -o tls,mounttargetip=192.0.2.0 fs-abcd123456789ef0 efs/
```

Sie können die entsprechenden Befehle zum Mounting des Dateisystems im Dialogfeld Anhängen einsehen und kopieren.

- a. Wählen Sie in der Amazon-EFS-Konsole das Dateisystem aus, das Sie mounten möchten, um dessen Detailseite anzuzeigen.
- b. Um die für dieses Dateisystem zu verwendenden Mountingbefehle anzuzeigen, wählen Sie oben rechts die Option Anhängen aus.

Auf dem Bildschirm Anhängen werden die entsprechenden Befehle angezeigt, die zum Mounting des Dateisystems auf folgende Arten verwendet werden können:

- (Mounting über DNS) Unter Verwendung des DNS-Namens des Dateisystems mit der EFS-Mountinghilfe oder einem NFS-Client.
- (Mounting über IP) Unter Verwendung der IP-Adresse des Mountingziels in der ausgewählten Availability Zone mit einem NFS-Client.

Mounten auf EC2-Mac-Instances mit dem EFS-Mount-Helper

Dieser Vorgang erfordert die folgenden Voraussetzungen:

- Sie haben das `amazon-efs-utils` Paket auf der Amazon EC2 Mac-Instance installiert. Weitere Informationen finden Sie unter [Installation des Amazon-EFS-Clients auf EC2-Mac-Instances, auf denen macOS Big Sur, macOS Monterey oder macOS Ventura ausgeführt wird](#).
- Das Dateisystem, das Sie gerade erstellt haben, verfügt über Mountingziele. Sie können Mountingziele zusammen mit dem Dateisystem erstellen und sie zu vorhandenen Dateisystemen hinzufügen. Weitere Informationen finden Sie unter [Verwalten der Mountingziele](#).
- Sie mounten das Dateisystem auf einer EC2-Mac-Instance, auf der macOS Big Sur, Monterey oder Ventura ausgeführt wird. Andere macOS-Versionen werden nicht unterstützt.

Note

Es werden nur EC2-Mac-Instances unterstützt, auf denen macOS Big Sur, Monterey oder Ventura ausgeführt wird. Andere macOS-Versionen werden für die Verwendung mit Amazon EFS nicht unterstützt.

So mounten Sie Ihr EFS-Dateisystem mit dem EFS-Mount-Helper auf EC2-Mac-Instances, auf denen macOS Big Sur, Monterey oder Ventura ausgeführt wird

1. Öffnen Sie über Secure Shell (SSH) ein Terminal für die EC2 Mac-Instance und melden Sie sich mit dem entsprechenden Benutzernamen an. Weitere Informationen finden Sie unter [Mit Ihrer EC2-Instance verbinden](#) im Amazon-EC2-Benutzerhandbuch.
2. Erstellen Sie mit dem folgenden Befehl ein Verzeichnis, das Sie als Mountingpunkt für das Dateisystem verwenden:

```
sudo mkdir efs
```

3. Führen Sie den folgenden Befehl aus, um das Dateisystem zu mounten.

 Note

Standardmäßig überträgt die EFS-Mountinghilfe beim Mounting auf EC2-Mac-Instances die Daten verschlüsselt, unabhängig davon, ob Sie im Mountingbefehl die Option `tls` verwenden oder nicht.

```
sudo mount -t efs file-system-id efs-mount-point/
```

```
sudo mount -t efs fs-abcd123456789ef0 efs/
```

Sie können die Option `tls` auch bei beim Mounting verwenden.

```
sudo mount -t efs -o tls fs-abcd123456789ef0:/ efs
```

Um ein Dateisystem auf einer EC2-Mac-Instance ohne Verschlüsselung bei der Übertragung zu mounten, verwenden Sie die Option `notls` wie im folgenden Befehl gezeigt.

```
sudo mount -t efs -o notls file-system-id efs-mount-point/
```

Sie können die entsprechenden Befehle zum Mounting des Dateisystems im Dialogfeld Anhängen der Managementkonsole wie im Folgenden beschrieben einsehen und kopieren.

- a. Wählen Sie in der Amazon-EFS-Konsole das Dateisystem aus, das Sie mounten möchten, um dessen Detailseite anzuzeigen.
- b. Um die für dieses Dateisystem zu verwendenden Mountingbefehle anzuzeigen, wählen Sie oben rechts die Option Anhängen aus.

Auf dem Bildschirm Anhängen werden die entsprechenden Befehle angezeigt, die zum Mounting des Dateisystems auf folgende Arten verwendet werden können:

- (Mounting über DNS) Unter Verwendung des DNS-Namens des Dateisystems mit der EFS-Mountinghilfe oder einem NFS-Client.
- (Mounting über IP) Unter Verwendung der IP-Adresse des Mountingziels in der ausgewählten Availability Zone mit einem NFS-Client.

Mounten von EFS-Dateisystemen von einem anderen AWS-Region

Um Ihr EFS-Dateisystem von einer EC2-Instance aus zu mounten, die sich in einem anderen Dateisystem AWS-Region als dem Dateisystem befindet, müssen Sie den `region` Eigenschaftswert in der `efs-utils.conf` Datei bearbeiten.

Um die **region** Eigenschaft zu bearbeiten in **efs-utils.conf**

1. Greifen Sie über Secure Shell (SSH) auf das Terminal für die EC2-Instance zu und melden Sie sich mit dem entsprechenden Benutzernamen an. Weitere Informationen finden Sie unter [Mit Ihrer EC2-Instance verbinden](#) im Amazon-EC2-Benutzerhandbuch.
2. Suchen Sie die Datei `/etc/amazon/efs/efs-utils.conf` und öffnen Sie sie in einem Texteditor Ihrer Wahl.
3. Suchen Sie die folgende Zeile:

```
#region = us-east-1
```

- a. Entfernen Sie das Kommentarzeichen für die Zeile.
 - b. Wenn sich das Dateisystem nicht in der `us-east-1` Region befindet, `us-east-1` ersetzen Sie es durch die ID der Region, in der sich das Dateisystem befindet.
 - c. Speichern Sie die Änderungen.
4. Fügen Sie einen Hosteintrag für das regionsübergreifende Mounting hinzu. Weitere Information dazu finden Sie unter [Schritt 3: Fügen Sie einen Hosteintrag für das Mount-Ziel hinzu](#).
 5. Mounting Sie das Dateisystem mit der EFS-Mountinghilfe für [Linux](#)- oder [Mac](#)-Instances.

Mounting von One-Zone-Dateisystemen

EFS One Zone-Dateisysteme unterstützen nur ein einzelnes Mount-Ziel, das sich in derselben Availability Zone wie das Dateisystem befindet. Sie können keine zusätzlichen Mountingziele hinzufügen. In diesem Abschnitt wird beschrieben, was beim Mounting von One-Zone-Dateisystemen zu beachten ist.

Sie können Datenübertragungsgebühren zwischen Availability Zones vermeiden und eine bessere Leistung erzielen, indem Sie über eine Amazon-EC2-Datenverarbeitungs-Instance auf ein EFS-Dateisystem zugreifen, die sich in derselben Availability Zone wie das Mountingziel des Dateisystems befindet.

Voraussetzungen für die in diesem Abschnitt beschriebenen Verfahren:

- Sie haben das `amazon-efs-utils` package-Paket für die EC2-Instance installiert. Weitere Informationen finden Sie unter [Den Amazon EFS-Client installieren](#).
- Das Dateisystem, das Sie gerade erstellt haben, verfügt über ein Mountingziel. Weitere Informationen finden Sie unter [Verwalten der Mountingziele](#).

Mounting von One-Zone-Dateisystemen auf EC2 in einer anderen Availability Zone

Wenn Sie ein One Zone-Dateisystem auf einer Amazon EC2 EC2-Instance mounten, die sich in einer anderen Availability Zone befindet, müssen Sie den Availability Zone-Namen des Dateisystems oder den DNS-Namen des Mount-Ziels des Dateisystems im Mount Helper-Mount-Befehl angeben.

Erstellen Sie mit dem folgenden Befehl das Verzeichnis `efs`, das Sie als Mountingpunkt für das Dateisystem verwenden:

```
sudo mkdir efs
```

Verwenden Sie den folgenden Befehl, um das Dateisystem mithilfe der EFS-Mountinghilfe zu mounten. Der Befehl gibt den Namen der Availability Zone des Dateisystems an.

```
sudo mount -t efs -o az=availability-zone-name,tls file-system-id mount-point/
```

Dies ist der Befehl mit Beispielwerten:

```
sudo mount -t efs -o az=us-east-1a,tls fs-abcd1234567890ef efs/
```

Mit dem folgenden Befehl, in dem der DNS-Name des Mountingziels des Dateisystems angegeben ist, wird das Dateisystem gemountet.

```
sudo mount -t efs -o tls mount-target-dns-name mount-point/
```

Dies ist der Befehl mit einem DNS-Beispielnamen für das Mountingziel.

```
sudo mount -t efs -o tls us-east-1a.fs-abcd1234567890ef9.efs.us-east-1.amazonaws.com  
efs/
```


Automatisches Mounting von One-Zone-Dateisystemen in einer anderen Availability Zone mit der EFS-Mountinghilfe

Wenn Sie `/etc/fstab` zum Mounting eines EFS One-Zone-Dateisystems auf einer EC2-Instance verwenden, die sich in einer anderen Availability Zone befindet, müssen Sie den Availability-Zone-Namen des Dateisystems oder den DNS-Namen des Mountingziels des Dateisystems im Eintrag `/etc/fstab` angeben.

```
availability-zone-name.file-system-id.efs.aws-region.amazonaws.com:/ efs-mount-point  
efs defaults,_netdev,noresvport,tls 0 0
```

```
us-east-1a.fs-abc123def456a7890.efs.us-east-1.amazonaws.com:/ efs-one-zone efs  
defaults,_netdev,noresvport,tls 0 0
```

Automatisches Mounting von One-Zone-Dateisystemen mit NFS

Wenn Sie ein EFS-Dateisystem mithilfe `/etc/fstab` von One Zone Storage auf einer EC2-Instance mounten, die sich in einer anderen Availability Zone befindet, müssen Sie den Availability Zone-Namen des Dateisystems mit dem DNS-Namen des Dateisystems im `/etc/fstab` Eintrag angeben.

```
availability-zone-name.file-system-id.efs.aws-region.amazonaws.com:/ efs-mount-point  
nfs4  
nfsvers=4.1,rsize=1048576,wsiz=1048576,hard,timeo=600,retrans=2,noresvport,_netdev 0  
0
```

```
us-east-1a.fs-abc123def456a7890.efs.us-east-1.amazonaws.com:/ efs-one-zone nfs4  
nfsvers=4.1,rsize=1048576,wsiz=1048576,hard,timeo=600,retrans=2,noresvport,_netdev 0  
0
```

Weitere Informationen zum Bearbeiten der `/etc/fstab`-Datei und zu den in diesem Befehl verwendeten Werten finden Sie unter [Automatisches Mounten von EFS-Dateisystemen](#).

Dateisysteme mit einem One Zone-Dateisystem auf anderen AWS Compute-Instances einhängen

Wenn Sie ein One Zone-Dateisystem mit Amazon Elastic Container Service, Amazon Elastic Kubernetes Service oder verwenden, müssen Sie den Service so konfigurieren AWS Lambda, dass er dieselbe Availability Zone verwendet, in der sich das EFS-Dateisystem befindet. Dies wird wie folgt veranschaulicht und in den folgenden Abschnitten beschrieben.



Herstellen einer von Amazon Elastic Container Service ausgehenden Verbindung

Sie können EFS-Dateisysteme mit Amazon ECS verwenden, um Dateisystemdaten für Ihre Flotte von Container-Instances gemeinsam zu nutzen, sodass Ihre Aufgaben unabhängig von der Instance, auf der sie landen, Zugriff auf denselben persistenten Speicher haben. Um EFS One Zone-Dateisysteme mit Amazon ECS zu verwenden, sollten Sie beim Starten Ihrer Aufgabe nur Subnetze auswählen, die sich in derselben Availability Zone wie Ihr Dateisystem befinden. Weitere Informationen finden Sie unter [Amazon-EFS-Volumes](#) im Entwicklerhandbuch für Amazon Elastic Container Service.

Herstellen einer von Amazon Elastic Kubernetes Service ausgehenden Verbindung

Wenn Sie ein One Zone-Dateisystem von Amazon EKS aus bereitstellen, können Sie den Amazon EFS [Container Storage Interface](#) (CSI) -Treiber verwenden, der EFS-Zugriffspunkte unterstützt, um ein Dateisystem für mehrere Pods in einem Amazon EKS- oder selbstverwalteten Kubernetes-Cluster gemeinsam zu nutzen. Der Amazon EFS CSI-Treiber ist im Fargate-Stack installiert. Wenn Sie den Amazon EFS CSI-Treiber mit EFS One Zone-Dateisystemen verwenden, können Sie beim Starten Ihres Pods `nodeSelector` diese Option verwenden, um sicherzustellen, dass er in derselben Availability Zone wie Ihr Dateisystem geplant wird.

Verbindung wird hergestellt von AWS Lambda

Sie können Amazon EFS mit verwenden AWS Lambda , um Daten über Funktionsaufrufe hinweg gemeinsam zu nutzen, große Referenzdatendateien zu lesen und Funktionsausgaben in einen persistenten und gemeinsam genutzten Speicher zu schreiben. Lambda verbindet die Funktionsinstanzen sicher mit den EFS-Mount-Zielen, die sich in derselben Availability Zone und demselben Subnetz befinden. Wenn Sie Lambda mit One-Zone-Dateisystemen verwenden, konfigurieren Sie die Funktion so, dass nur Aufrufe in Subnetze gestartet werden, die sich in derselben Availability Zone wie das Dateisystem befinden.

Mounting mit IAM-Autorisierung

Verwenden Sie den EFS-Mount-Helper, um Ihr EFS-Dateisystem mithilfe der AWS Identity and Access Management (IAM-) Autorisierung auf Linux-Instances zu mounten. Weitere Hinweise zur Verwendung der IAM-Autorisierung für NFS-Clients finden Sie unter [Verwendung von IAM zur Steuerung des Zugriffs auf Dateisysteme](#).

In den folgenden Abschnitten müssen Sie ein Verzeichnis erstellen, das als Einhängpunkt für das Dateisystem verwendet werden soll. Mit dem folgenden Befehl können Sie das Verzeichnis `efs` für den Mountingpunkt erstellen:

```
sudo mkdir efs
```

Anschließend können Sie Instances von *efs-mount-point* durch `efs` ersetzen.

Mounting mit IAM mithilfe eines EC2-Instance-Profils

Wenn Sie mit IAM-Autorisierung zu einer Amazon-EC2-Instance mit einem Instance-Profil mounten, verwenden Sie die nachfolgend gezeigten Mountingoptionen `tls` und `iam`.

```
$ sudo mount -t efs -o tls,iam file-system-id efs-mount-point/
```

Um automatisch mit IAM-Autorisierung eine EC2-Instance mit einem Instance-Profil zu mounten, fügen Sie der `/etc/fstab` Datei auf der EC2-Instance die folgende Zeile hinzu.

```
file-system-id:/ efs-mount-point efs _netdev,tls,iam 0 0
```

Mounting mit IAM mithilfe eines benannten Profils

Sie können das Mounten mit IAM-Autorisierung durchführen, indem Sie die IAM-Anmeldeinformationen verwenden `~/.aws/credentials`, die sich in der AWS CLI Anmeldeinformationsdatei oder der AWS CLI Konfigurationsdatei befinden. `~/.aws/config` Wenn "awsprofile" nicht angegeben ist, wird das „Standard“-Profil verwendet.

Um eine Linux-Instance unter Verwendung einer Datei für Anmeldeinformationen mit IAM-Autorisierung zu mounten, verwenden Sie die Mounting-Optionen `tls`, `awsprofile` und `iam` (siehe unten).

```
$ sudo mount -t efs -o tls,iam,awsprofile=namedprofile file-system-id efs-mount-point/
```

Um automatisch mit IAM-Autorisierung zu einer Linux-Instance mithilfe einer Anmeldeinformationendatei zu mounten, fügen Sie der `/etc/fstab`-Datei auf der EC2-Instance die folgende Zeile hinzu.

```
file-system-id:/ efs-mount-point efs _netdev,tls,iam,awsprofile=namedprofile 0 0
```

Mounting mit EFS-Zugangspunkten

Sie müssen die EFS-Mountinghilfe verwenden, um ein EFS-Dateisystem mit einem EFS-Zugangspunkt zu mounten.

Note

Sie müssen ein oder mehrere Mountingziele für das Dateisystem konfigurieren, wenn Sie ein Dateisystem mithilfe von EFS-Zugangspunkten mounten.

Wenn Sie ein Dateisystem mithilfe eines Zugangspunkts mounten, enthält der Mountingbefehl zusätzlich zu den regulären Mountingoptionen die Mountingoptionen `access-point-id` und `tls`. Ein Beispiel.

```
$ sudo mount -t efs -o tls,accesspoint=access-point-id file-system-id efs-mount-point
```

Wenn Sie ein Dateisystem mithilfe eines Zugangspunkts automatisch mounten möchten, fügen Sie der `/etc/fstab`-Datei auf der EC2-Instance die folgende Zeile hinzu.

```
file-system-id efs-mount-point efs _netdev,tls,accesspoint=access-point-id 0 0
```

Weitere Hinweise zu EFS-Zugangspunkten finden Sie unter [Arbeiten mit Zugriffspunkten](#).

Mounten von EFS auf mehrere EC2-Instances

Sie können EFS-Dateisysteme remote und sicher auf mehreren Amazon EC2 EC2-Instances bereitstellen, ohne sich mit dem Befehl AWS Systems Manager Run bei den Instances anmelden zu müssen. Weitere Informationen zu AWS Systems Manager Run Command finden Sie unter [AWS Systems Manager Run Command](#) im AWS Systems Manager Benutzerhandbuch. Es gelten die folgenden Voraussetzungen, bevor EFS-Dateisysteme mit dieser Methode gemountet werden können:

1. Die EC2-Instances werden mit einem Instance-Profil gestartet, das die Berechtigungsrichtlinie AmazonElasticFileSystemsUtils enthält. Weitere Informationen finden Sie unter [Schritt 1: Konfigurieren Sie ein \(IAM\)-Instance-Profil mit den erforderlichen Berechtigungen](#).
2. Version 1.28.1 oder höher des Amazon EFS-Clients (amazon-efs-utils Paket) ist auf den EC2-Instances installiert. Sie können AWS Systems Manager verwenden, um das Paket automatisch auf Ihren Instanzen zu installieren. Weitere Informationen finden Sie unter [Schritt 2: Konfigurieren Sie eine von State Manager verwendete Zuordnung](#).

So mounten Sie mehrere EFS-Dateisysteme mithilfe der Konsole auf mehreren EC2-Instances

1. Öffnen Sie die AWS Systems Manager Konsole unter <https://console.aws.amazon.com/systems-manager/>.
2. Wählen Sie im Navigationsbereich Run Command aus.
3. Wählen Sie die Option Run a command.
4. Geben Sie im Suchfeld Befehle **AWS-RunShellScript** ein.
5. Wählen Sie AWS- RunShellScript.
6. Geben Sie unter Befehlsparameter den Mountingbefehl für jedes EFS-Dateisystem ein, das Sie mounten möchten. Beispiel:

```
sudo mount -t efs -o tls fs-12345678:/ /mnt/efs  
sudo mount -t efs -o tls,accesspoint=fsap-12345678 fs-01233210 /mnt/efs
```

Weitere Informationen zur Verwendung von EFS-Mountingbefehlen mithilfe des Amazon-EFS-Clients finden Sie unter [Mounten auf EC2-Linux-Instances mit dem EFS-Mount-Helper](#) oder [Mounten auf EC2-Mac-Instances mit dem EFS-Mount-Helper](#).

7. Wählen Sie die AWS Systems Manager verwalteten EC2-Zielinstanzen aus, auf denen der Befehl ausgeführt werden soll.
8. Nehmen Sie bei Bedarf weitere Einstellungen vor. Wählen Sie dann Ausführen aus, um den Befehl auszuführen und die im Befehl angegebenen EFS-Dateisysteme zu mounten.

Sobald Sie den Befehl ausgeführt haben, sehen Sie seinen Status im Befehlsverlauf.

Mounten von EFS-Dateisystemen von einer anderen AWS-Konto oder VPC

Sie können Ihr EFS-Dateisystem mithilfe der IAM-Autorisierung für NFS-Clients und EFS-Zugriffspunkte mithilfe des EFS-Mount-Helpers mounten. Standardmäßig verwendet die EFS-Mountinghilfe DNS (Domain Name Service), um die IP-Adresse Ihres EFS-Mountingziels aufzulösen. Wenn Sie das Dateisystem von einem anderen Konto oder einer anderen Virtual Private Cloud (VPC) mounten, müssen Sie das EFS-Mountingziel manuell auflösen.

Im Folgenden finden Sie Anweisungen zum Bestimmen der richtigen IP-Adresse des EFS-Mountingziels für Ihren NFS-Client. Sie finden auch Anweisungen zum Konfigurieren des Clients zum Mounten des EFS-Dateisystems unter Verwendung dieser IP-Adresse.

Topics

- [Einhängen von EFS-Dateisystemen von einem anderen AWS-Konto](#)
- [Mounten von EFS-Dateisystemen von einer anderen VPC](#)

Einhängen von EFS-Dateisystemen von einem anderen AWS-Konto

Mit Shared VPCs können Sie ein EFS-Dateisystem, das einer anderen gehört, AWS-Konto von Amazon EC2 EC2-Instances mounten, die einer anderen AWS-Konto gehören. Weitere Informationen zur Einrichtung einer gemeinsam genutzten VPC finden Sie unter [Teilen Sie Ihre VPC mit anderen Konten](#) im Amazon VPC Peering Guide.

Nachdem Sie die VPC-Freigabe eingerichtet haben, können die EC2-Instances das EFS-Dateisystem mit Domain Name System (DNS)-Namensauflösung oder dem EFS-Hilfsprogramm mounten. Wir empfehlen die EFS-Mountinghilfe zum Mounting von EFS-Dateisystemen.

Mounten von EFS-Dateisystemen von einer anderen VPC

Wenn Sie eine VPC-Peering-Verbindung oder ein Transit-Gateway verwenden, um VPCs zu verbinden, können Amazon EC2 EC2-Instances, die sich in einer VPC befinden, auf EFS-Dateisysteme in einer anderen VPC zugreifen, auch wenn sie zu unterschiedlichen Konten gehören. VPCs

In einer anderen VPC kann keine DNS-Namensauflösung für EFS-Mountingpunkte verwendet werden. Verwenden Sie die IP-Adresse der Mountingpunkte in der entsprechenden Availability Zone, um Ihr EFS-Dateisystem zu mounten.

Alternativ können Sie Amazon Route 53 als Ihren DNS-Service verwenden. In Route 53 können Sie die EFS-Mount-Ziel-IP-Adressen von einer anderen VPC auflösen, indem Sie eine private gehostete Zone und einen Ressourcendatensatz erstellen. Weitere Informationen dazu finden Sie unter [Arbeiten mit privat gehosteten Zonen](#) im Amazon Route 53-Entwicklerhandbuch.

Voraussetzungen

Führen Sie die folgenden Schritte aus, bevor Sie das folgende Verfahren anwenden:

- Installieren Sie den Amazon-EFS-Client, der Teil der amazon-efs-utils-Dienstprogramme ist, auf der Datenverarbeitungs-Instance, auf der Sie das EFS-Dateisystem mounten. Sie verwenden die EFS-Mountinghilfe, die in amazon-efs-utils enthalten ist, um das Dateisystem zu mounten. Anweisungen zur Installation von amazon-efs-utils finden Sie unter [Den Amazon EFS-Client installieren](#).
- Lassen Sie die Aktion `ec2:DescribeAvailabilityZones` in der IAM-Richtlinie für die IAM-Rolle zu, die Sie der Instance zugewiesen haben. Wir empfehlen, dass Sie die AWS verwaltete Richtlinie `AmazonElasticFileSystemsUtils` an eine IAM-Entität anhängen, um die erforderlichen Berechtigungen für die Entität bereitzustellen.
- Wenn Sie von einem anderen System aus mounten AWS-Konto, aktualisieren Sie die Dateisystem-Ressourcenrichtlinie, um die `elasticfilesystem:DescribeMountTarget` Aktion für den Haupt-ARN eines anderen zuzulassen AWS-Konto. Beispiel:

```
{
  "Id": "access-point-example03",
  "Statement": [
    {
      "Sid": "access-point-statement-example03",
      "Effect": "Allow",
```

```
    "Principal": {"AWS": "arn:aws:iam::555555555555:root"},
    "Action": "elasticfilesystem:DescribeMountTargets",
    "Resource": "arn:aws:elasticfilesystem:us-east-2:111122223333:file-
system/fs-12345678"
  }
]
```

Weitere Informationen zu Ressourcenrichtlinien für EFS-Dateisysteme finden Sie unter [Ressourcenbasierte Richtlinien in Amazon EFS](#).

- Installieren Sie Botocore. Der EFS-Client verwendet Botocore, um die IP-Adresse des Mountingziels abzurufen, wenn der DNS-Name des Dateisystems beim Mounting eines Dateisystems in einer anderen VPC nicht aufgelöst werden kann. Weitere Informationen finden Sie in der README-Datei `amazon-efs-utils` unter [Installieren von Botocore](#).
- Richten Sie entweder eine VPC-Peering-Verbindung oder ein VPC-Transit-Gateway ein.

Sie verbinden die VPC des Clients mit der VPC Ihres EFS-Dateisystems über eine VPC-Peering-Verbindung oder ein VPC-Transit-Gateway. Wenn Sie eine VPC-Peering-Verbindung oder ein Transit-Gateway verwenden, um VPCs zu verbinden, können Amazon EC2 EC2-Instances, die sich in einer VPC befinden, auf EFS-Dateisysteme in einer anderen VPC zugreifen, auch wenn sie zu unterschiedlichen Konten gehören. VPCs

Ein Transit-Gateway ist ein Netzwerk-Transit-Hub, über den Sie Ihre Netzwerke und Ihre lokalen Netzwerke miteinander verbinden können. VPCs Weitere Informationen zur Verwendung von VPC-Transit-Gateways finden Sie unter [Erste Schritte mit Transit-Gateways](#) im Amazon VPC-Gateways-Handbuch.

Eine VPC-Peering-Verbindung ist eine Netzwerkverbindung zwischen zwei VPCs Dieser Verbindungstyp ermöglicht es Ihnen, den Verkehr zwischen ihnen mithilfe von privaten Internetprotokolladressen der Version 4 (IPv4) oder der Internetprotokoll-Version 6 (IPv6) weiterzuleiten. Sie können VPC-Peering verwenden, um VPCs innerhalb derselben AWS-Region oder zwischen ihnen eine Verbindung herzustellen. AWS-Regionen Weitere Informationen zu VPC-Peering finden Sie unter [Was ist VPC-Peering?](#) im Amazon VPC Peering Guide.

Um eine hohe Verfügbarkeit Ihres Dateisystems sicherzustellen, empfehlen wir, immer eine IP-Adresse eines EFS-Mountingziels zu verwenden, die sich in derselben Availability Zone (AZ) wie der NFS-Client befindet. Wenn Sie ein EFS-Dateisystem mounten, das sich in einem anderen Konto befindet, stellen Sie sicher, dass sich der NFS-Client und das EFS-Mountingziel in derselben

Availability-Zone-ID befinden. Diese Anforderung gilt, da AZ-Namen zwischen Konten unterschiedlich sein können.

So mounten Sie ein EFS-Dateisystem mithilfe von IAM oder einem Zugangspunkt in einer anderen VPC

1. Stellen Sie eine Verbindung zu Ihrer EC2- Instance her. Weitere Informationen finden Sie unter [Mit Ihrer EC2-Instance verbinden](#) im Amazon-EC2-Benutzerhandbuch.
2. Erstellen Sie mit dem folgenden Befehl ein Verzeichnis zum Mounten des Dateisystems.

```
$ sudo mkdir /mnt/efs
```

3. Verwenden Sie den folgenden Befehl, um das Dateisystem mit der IAM-Autorisierung zu mounten:

```
$ sudo mount -t efs -o tls,iam file-system-dns-name /mnt/efs/
```

Weitere Hinweise zur Verwendung von IAM-Autorisierung mit EFS finden Sie unter [Verwendung von IAM zur Steuerung des Zugriffs auf Dateisysteme](#).

Verwenden Sie den folgenden Befehl, um das Dateisystem mithilfe eines EFS-Zugangspunkts zu mounten:

```
$ sudo mount -t efs -o tls,accesspoint=access-point-id file-system-dns-name /mnt/efs/
```

Weitere Hinweise zu EFS-Zugangspunkten finden Sie unter [Arbeiten mit Zugriffspunkten](#).

Mounten von EFS-Dateisystemen von einem anderen AWS-Region

Wenn Sie Ihr EFS-Dateisystem von einer anderen VPC aus mounten, die sich in einem anderen Dateisystem AWS-Region als dem Dateisystem befindet, müssen Sie die `efs-utils.conf` Datei bearbeiten. Suchen Sie in `/dist/efs-utils.conf` die folgenden Zeilen:

```
#region = us-east-1
```

Entfernen Sie das Kommentarzeichen für die Zeile und ersetzen Sie den Wert für die ID der Region, in der sich das Dateisystem befindet, falls es sich nicht in der Region `us-east-1` befindet.

Verwenden des Netzwerkdateisystems zum Mounten von EFS-Dateisystemen

Im Folgenden erfahren Sie, wie Sie den Network File System (NFS) -Client installieren und wie Sie Ihr Amazon EFS-Dateisystem auf einer Amazon EC2 EC2-Instance mounten. Dazu finden Sie eine Erläuterung des mount-Befehls und der verfügbaren Optionen zur Angabe des DNS-Namens Ihres Dateisystems im mount-Befehl. Dazu erfahren Sie, wie Sie mit der Datei `fstab` Ihr Dateisystem nach Systemneustarts automatisch erneut mounten.

Note

In diesem Abschnitt erfahren Sie, wie Sie Ihr Amazon EFS-Dateisystem ohne das `amazon-efs-utils` Paket mounten. Um eine Verschlüsselung von Daten während der Übertragung mit Ihrem Dateisystem zu verwenden, müssen Sie Ihr Dateisystem mit Transport Layer Security (TLS) mounten. Zu diesem Zweck empfehlen wir, das `amazon-efs-utils` Paket zu verwenden. Weitere Informationen finden Sie unter [Den Amazon EFS-Client installieren](#).

Themen

- [Voraussetzungen](#)
- [NFS-Support](#)
- [Installieren des NFS-Clients](#)
- [Empfohlene NFS-Mount-Einstellungen](#)
- [Mounting auf Amazon EC2 mit einem DNS-Namen](#)
- [Mounting mit einer IP-Adresse](#)

Voraussetzungen

Bevor Sie ein Dateisystem mounten können, stellen Sie sicher, dass Sie die folgenden Anforderungen erfüllen:

- Erstellen, konfigurieren und starten Sie Ihre zugehörigen AWS Ressourcen. Detaillierte Anweisungen finden Sie unter [Erste Schritte mit Amazon EFS](#).

- Erstellen Sie VPC-Sicherheitsgruppen für Ihre Amazon EC2 EC2-Instances und mounten Sie Ziele mit dem erforderlichen eingehenden und ausgehenden Zugriff. Weitere Informationen finden Sie unter [Verwenden von VPC-Sicherheitsgruppen](#).

NFS-Support

Amazon EFS unterstützt beim Mounten Ihrer Dateisysteme auf Amazon-EC2-Instances die Network File System-Versionen 4.0 und 4.1 (NFSv4) und NFSv4.0-Protokolle. Obwohl NFSv4.0 unterstützt wird, empfehlen wir Ihnen, NFSv4.1 zu verwenden. Zum Mounten Ihres Amazon-EFS-Dateisystems auf Ihrer Amazon-EC2-Instance ist auch ein NFS-Client erforderlich, der Ihr gewähltes NFSv4-Protokoll unterstützt. Amazon-EC2-Mac-Instances, auf denen macOS Big Sur ausgeführt wird, unterstützen nur NFS v4.0.

Amazon EFS unterstützt die Mount-Option `nconnect` nicht.

Note

Für die Linux-Kernel-Versionen 5.4.* verwendet der Linux-NFS-Client einen `read_ahead_kb`-Standardwert von 128 KB. Wir empfehlen, diesen Wert auf 15 MB zu erhöhen. Weitere Informationen finden Sie unter [Optimierung der NFS-Größe von `read\_ahead\_kb`](#).

Um eine optimale Leistung zu erreichen und verschiedene bekannte NFS-Client-Bugs zu vermeiden, empfehlen wir, mit einem aktuellen Linux-Kernel zu arbeiten. Wenn Sie eine Linux-Unternehmensdistribution verwenden, empfehlen wir Folgendes:

- Amazon Linux 2
- Amazon Linux 2017.09 oder neuer
- Red Hat Enterprise Linux (und Derivate wie CentOS) Version 8 und neuer
- Ubuntu 16.04 LTS und höher
- SLES 12 Sp2 oder höher

Wenn Sie eine andere Verteilung oder einen benutzerdefinierten Kernel verwenden, empfehlen wir Kernel-Version 4.3 oder neuer. Zum Beheben von Problemen im Zusammenhang mit bestimmten AMI- oder Kernel-Versionen bei der Verwendung von Amazon EFS von einer EC2-Instance siehe [Beheben von AMI- und Kernel-Problemen](#).

 Note

Das Mounten EFS EFS-Dateisystemen mit Amazon EC2 EC2-Instances, auf denen Microsoft Windows ausgeführt wird, wird nicht unterstützt.

Installieren des NFS-Clients

Um Ihr EFS-Dateisystem auf Ihrer Amazon EC2 EC2-Instance zu mounten, müssen Sie zunächst einen NFS-Client installieren. Zum Herstellen einer Verbindung mit Ihrer EC2-Instance und zur Installation eines NFS-Clients benötigen Sie den öffentlichen DNS-Namen der EC2-Instance sowie einen Benutzernamen für die Anmeldung. Dieser Benutzername für Ihre Instance ist in der Regel `ec2-user`.

So stellen Sie eine Verbindung mit Ihrer EC2-Instance her und installieren den NFS-Client:

1. Stellen Sie eine Verbindung zu Ihrer EC2- Instance her. Weitere Informationen finden Sie unter [Mit Ihrer EC2-Instance verbinden](#) im Amazon-EC2-Benutzerhandbuch.

Die Schlüsseldatei darf für SSH nicht öffentlich anzeigbar sein. Sie können den Befehl `chmod 400 filename.pem` verwenden, um diese Berechtigungen einzurichten. Weitere Informationen finden Sie unter [Erstellen eines key pair für Ihre Amazon EC2 EC2-Instance](#).

2. (Optional) Rufen Sie Aktualisierungen ab, und führen Sie einen Neustart durch.

```
$ sudo yum -y update
$ sudo reboot
```

3. Stellen Sie nach dem Neustart erneut eine Verbindung mit Ihrer EC2-Instance her.
4. Installieren Sie den NFS-Client.

Wenn Sie ein Amazon Linux-AMI oder Red Hat Linux-AMI verwenden, installieren Sie den NFS-Client mit dem folgenden Befehl.

```
$ sudo yum -y install nfs-utils
```

Wenn Sie ein Ubuntu-Amazon-EC2-AMI verwenden, installieren Sie den NFS-Client mit dem folgenden Befehl.

```
$ sudo apt-get -y install nfs-common
```

5. Starten Sie den NFS-Service mit dem folgenden Befehl:

```
$ sudo service nfs-server start
```

6. Stellen Sie sicher, dass der NFS-Service wie folgt gestartet wurde.

```
$ sudo service nfs-server status
Redirecting to /bin/systemctl status nfs.service
# nfs-server.service - NFS server and services
   Loaded: loaded (/usr/lib/systemd/system/nfs-server.service; disabled; vendor
   preset: disabled)
   Active: active (exited) since Wed 2019-10-30 16:13:44 UTC; 5s ago
   Process: 29446 ExecStart=/usr/sbin/rpc.nfsd $RPCNFSDARGS (code=exited, status=0/
  SUCCESS)
   Process: 29441 ExecStartPre=/bin/sh -c /bin/kill -HUP `cat /run/gssproxy.pid`
  (code=exited, status=0/SUCCESS)
   Process: 29439 ExecStartPre=/usr/sbin/exportfs -r (code=exited, status=0/SUCCESS)
   Main PID: 29446 (code=exited, status=0/SUCCESS)
   CGroup: /system.slice/nfs-server.service
```

Wenn Sie einen benutzerdefinierten Kernel verwenden (d. h. wenn Sie ein benutzerdefiniertes AMI erstellen), müssen Sie mindestens das Client-Kernelmodul NFSv4 .1 und den richtigen NFS4 Userspace-Mount-Helper einbinden.

Note

Wenn Sie beim Starten Ihrer Amazon-EC2-Instance Amazon Linux AMI 2016.03.0 oder Amazon Linux AMI 2016.09.0 auswählen, müssen Sie `nfs-utils` nicht installieren, da es bereits standardmäßig in der AMI enthalten ist.

Dann: Mounten Sie Ihr Dateisystem

Verwenden Sie eines der folgenden Verfahren, um Ihr Dateisystem zu mounten.

- [Mounting auf Amazon EC2 mit einem DNS-Namen](#)
- [Mounting mit einer IP-Adresse](#)

- [Automatisches Mounten von EFS-Dateisystemen](#)

Empfohlene NFS-Mount-Einstellungen

Wir empfehlen die folgenden Werte für die Mountingoptionen unter Linux:

- `noresvport` – Teilt dem NFS-Client mit, einen neuen nicht privilegierten Quellanschluss für Transmission Control Protocol (TCP) zu verwenden, wenn erneut eine Netzwerkverbindung eingerichtet wird. Die NFS-Clientsoftware in älteren Versionen des Linux-Kernels (Version v5.4 und darunter) enthält ein Verhalten, das NFS-Klienten dazu veranlasst, nach einer Trennung der Verbindung zu versuchen, sich über denselben TCP-Quellport erneut zu verbinden. Dieses Verhalten entspricht nicht dem TCP RFC und kann diese Clients daran hindern, die Verbindung zu einem EFS-Dateisystem schnell wiederherzustellen.

Mit der Option `noresvport` können Sie sicherstellen, dass NFS-Clients bei einer erneuten Verbindung nach einem Netzwerkwiederherstellungsereignis transparent eine erneute Verbindung zu Ihrem EFS-Dateisystem herstellen und ununterbrochene Verfügbarkeit sicherstellen.

Important

Wir empfehlen dringend, die `noresvport`-Mounting-Option zu verwenden, um sicherzustellen, dass das EFS-Dateisystem nach einer erneuten Verbindung oder einem Netzwerkwiederherstellungsereignis ununterbrochen verfügbar ist.

Überlegen Sie, Ihr Dateisystem mit der [EFS-Mountinghilfe](#) zu mounten. Die EFS-Mountinghilfe verwendet NFS-Mounting-Optionen, die für Amazon-EFS-Dateisysteme optimiert sind.

- `rsize=1048576` – Legt die maximale Byteanzahl der Daten fest, die der NFS-Client für jede Netzwerk-READ-Anforderung erhalten kann. Dieser Wert gilt beim Lesen von Daten aus einer Datei in einem EFS-Dateisystem. Zur Vermeidung von Leistungseinbußen empfehlen wir die Verwendung der maximal möglichen Größe (bis zu 1048576).
- `wsize=1048576` – Legt die maximale Byteanzahl der Daten fest, die der NFS-Client für jede Netzwerk-WRITE-Anforderung senden kann. Dieser Wert gilt beim Schreiben von Daten in eine Datei in einem EFS-Dateisystem. Zur Vermeidung von Leistungseinbußen empfehlen wir die Verwendung der maximal möglichen Größe (bis zu 1048576).
- `hard` – Legt das Wiederherstellungsverhalten des NFS-Clients nach dem Timeout einer NFS-Anforderung fest, damit NFS-Anforderungen so lange wiederholt werden, bis der Server antwortet.

Zur Sicherstellung der Datenintegrität wird die Verwendung der dauerhaften Mountingoption (`hard`) empfohlen. Wenn Sie ein `soft`-Mount verwenden, legen Sie den `timeo`-Parameter auf mindestens 150 Zehntelsekunden (15 Sekunden) fest. Dadurch wird das Risiko einer Datenbeschädigung verringert, die bei `Soft-Mounts` inhärent ist.

- `timeo=600` – Legt den Timeout-Wert, der angibt, wie lange der NFS-Client auf eine Antwort wartet, bis er eine NFS-Anforderung wiederholt, auf 600 Zehntelsekunden (60 Sekunden) fest. Wenn Sie den Timeout-Parameter (`timeo`) ändern müssen, empfehlen wir, dass Sie einen Wert von mindestens 150, entsprechend 15 Sekunden, verwenden. Dadurch wird eine verringerte Leistung vermieden.
- `retrans=2` – Legt die Anzahl der Anforderungswiederholungen eines NFS-Clients vor dem Versuch einer weiteren Wiederherstellungsaktion auf 2 fest.
- `_netdev` – Sofern in `/etc/fstab` vorhanden, wird der Client an dem Versuch gehindert, das EFS-Dateisystem zu mounten, bis das Netzwerk aktiviert wurde.
- `nofail` – Wenn Ihre EC2-Instance unabhängig vom Status des gemounteten EFS-Dateisystems starten muss, fügen Sie die Option `nofail` zum Eintrag Ihres Dateisystems in Ihrer Datei `/etc/fstab` hinzu.

Wenn Sie die vorgenannten Standardwerte nicht verwenden, achten Sie auf Folgendes:

- Vermeiden Sie es generell, jegliche anderen Mounting-Optionen zu verwenden, die sich von den Standardoptionen unterscheiden, denn dies kann zu Leistungseinbußen und anderen Problemen führen. Beispielsweise können Änderungen der Puffergröße für Lese- oder Schreibvorgänge oder Deaktivierung der Attributzwischenspeicherung zu einer Leistungsverringerung führen.
- Amazon EFS ignoriert Quellports. Wenn Sie Amazon-EFS-Quellports ändern, hat dies keinerlei Auswirkungen.
- Amazon EFS unterstützt die Mount-Option `nconnect` nicht.
- Amazon EFS unterstützt keine der Kerberos-Sicherheitsvarianten. Beispielsweise führt der folgende Mounting-Befehl zu einem Fehler.

```
$ mount -t nfs4 -o krb5p <DNS_NAME>:/ /efs/
```

- Mounten Sie Ihr System möglichst mit dessen DNS-Namen. Dieser Name löst zur IP-Adresse des Amazon-EFS-Mountingziels in der gleichen Availability Zone wie Ihre Amazon-EC2-Instance auf. Wenn Sie ein Mountingziel in einer Availability Zone mounten, die sich von der Amazon-EC2-Instance unterscheidet, werden Standard-EC2-Gebühren für Daten erhoben, die zwischen

Availability Zones übertragen werden. Sie bemerken bei Dateisystemvorgängen möglicherweise auch erhöhte Latenzen.

- Weitere Einhängeloptionen und ausführliche Erläuterungen der Standardwerte finden Sie in der Linux-Dokumentation.

Mounting auf Amazon EC2 mit einem DNS-Namen

Note

Bevor Sie Ihr Dateisystem mounten, müssen Sie der Mount-Zielsicherheitsgruppe eine Regel hinzufügen, die eingehenden NFS-Zugriff auf Port 2049 von der EC2-Sicherheitsgruppe aus ermöglicht. Weitere Informationen finden Sie unter [Verwenden von VPC-Sicherheitsgruppen](#).

- Dateisystem-DNS-Name – Die einfachste Mountingoption ist die Verwendung des DNS-Namens des Dateisystems. Der DNS-Name des Dateisystems wird automatisch zur IP-Adresse des Mountingziels in der Availability Zone der die Verbindung herstellenden Amazon-EC2-Instance aufgelöst. Sie erhalten den DNS-Namen von der Konsole; wenn Sie die Dateisystem-ID haben, können Sie ihn aber auch gemäß der folgenden Konvention konstruieren.

```
file-system-id.efs.aws-region.amazonaws.com
```

Note

Die DNS-Auflösung für die DNS-Namen des Dateisystems erfordert, dass das Amazon-EFS-Dateisystem über ein Mountingziel in derselben Availability Zone wie die Client-Instance verfügt.

- Mit dem DNS-Namen des Dateisystems können Sie ein Dateisystem auf Ihrer Amazon-EC2-Linux-Instance mit dem folgenden Befehl mounten.

```
sudo mount -t nfs -o  
nfsvers=4.1,rsize=1048576,wsiz=1048576,hard,timeo=600,retrans=2,noresvport file-  
system-id.efs.aws-region.amazonaws.com:/ /efs-mount-point
```


- Mithilfe des DNS-Namens des Dateisystems können Sie mit dem folgenden Befehl ein Dateisystem auf Ihrer Amazon-EC2-Mac-Instance mounten, auf der eine unterstützte macOS-Version (Big Sur, Monterey, Ventura) ausgeführt wird.

```
sudo mount -t nfs -o  
nfsvers=4.0,rsz=65536,wsz=65536,hard,timeo=600,retrans=2,noresvport,mountport=2049 file-system-id.efs.aws-region.amazonaws.com:/ /efs
```

Important

Sie müssen `mountport=2049` verwenden, um beim Mounten auf EC2-Mac-Instances, auf denen unterstützte macOS-Versionen ausgeführt werden, erfolgreich eine Verbindung zum EFS-Dateisystem herzustellen.

- DNS-Name des Mountingziels – Im Dezember 2016 haben wir Dateisystem-DNS-Namen eingeführt. Wir stellen weiterhin einen DNS-Namen für jede Availability Zone bereit, um die Abwärtskompatibilität zu gewährleisten. Die allgemeine Form des DNS-Namens eines Mountingziels ist wie folgt.

```
availability-zone.file-system-id.efs.aws-region.amazonaws.com
```

Note

Die Mounten der Ziel-DNS-Namensauflösung in allen Availability Zones wird unterstützt.

In einigen Fällen könnten Sie ein Mountingziel löschen und dann ein neues in derselben Availability Zone erstellen. In einem solchen Fall ist der DNS-Name für dieses neue Mountingziel in der Availability Zone derselbe wie der DNS-Name für das alte Mountingziel.

Sie können die entsprechenden Befehle zum Mounting des Dateisystems im Dialogfeld Anhängen einsehen und kopieren.

Gehen Sie wie folgt vor, um die Mount-Befehle für Ihr Dateisystem anzuzeigen:

1. Wählen Sie in der Amazon-EFS-Konsole das Dateisystem aus, das Sie mounten möchten, um dessen Detailseite anzuzeigen.

2. Um die für dieses Dateisystem zu verwendenden Mountingbefehle anzuzeigen, wählen Sie oben rechts die Option Anhängen aus.

Auf dem Bildschirm Anhängen werden die entsprechenden Befehle angezeigt, die zum Mounting des Dateisystems verwendet werden können:

3. In der Standardansicht Über DNS mounten wird der Befehl zum Mounten des Dateisystems unter Verwendung des DNS-Namens des Dateisystems angezeigt, wenn das Mounten mit der EFS-Mountinghilfe oder einem NFS-Client erfolgt.

Eine Liste der AWS-Regionen, die Amazon EFS unterstützen, finden Sie unter [Amazon Elastic File System](#) in der Allgemeine AWS-Referenz.

Damit ein DNS-Name im mount-Befehl verwendet werden kann, muss folgendes gelten:

- Die EC2-Instance muss sich in einer VPC befinden und so konfiguriert sein, dass Sie den von Amazon bereitgestellten DNS-Server verwendet. Informationen zum Amazon DNS-Server finden Sie unter [DHCP-Optionssätze in Amazon VPC](#) im Amazon VPC-Benutzerhandbuch.
- Die VPC der EC2-Instance, die eine Verbindung herstellt, muss über DNS Resolution (DNS-Auflösung) und DNS Hostnames (DNS-Hostnamen) verfügen. Weitere Informationen finden Sie unter [DNS-Attribute für Ihre VPC](#) im Amazon VPC-Benutzerhandbuch.
- Die EC2-Instance, die eine Verbindung herstellt, muss sich in der gleichen VPC wie das EFS-Dateisystem befinden. Weitere Informationen zum Zugriff auf ein Dateisystem und dessen Mounting von einem anderen Standort oder einer anderen VPC finden Sie unter [Voraussetzungen](#) und [Tutorial: Mounten Sie ein Dateisystem von einer anderen VPC](#).

Note

Wir empfehlen, dass Sie nach dem Erstellen eines Mountingziels 90 Sekunden warten, bevor Sie das Dateisystem mounten. Durch diese Wartezeit können sich die DNS-Einträge vollständig dort verbreiten, AWS-Region wo sich das Dateisystem befindet.

Mounting mit einer IP-Adresse

Alternativ zum Mounting Ihres Amazon-EFS-Dateisystems mit dem DNS-Namen, können Amazon-EC2-Instances ein Dateisystem mithilfe der IP-Adresse eines Mountingziels mounten. Das Mounten

nach IP-Adresse funktioniert in Umgebungen, in denen DNS deaktiviert ist, z. B. VPCs wenn DNS-Hostnamen deaktiviert sind.

Sie können das Mounten eines Dateisystems auch mithilfe der IP-Adresse des Mountingziels als Fallback-Option für Anwendungen konfigurieren, die so konfiguriert sind, dass sie das Dateisystem standardmäßig unter Verwendung seines DNS-Namens mounten. Beim Herstellen einer Verbindung mit der IP-Adresse eines Mountingziels sollten EC2-Instances das Mounting unter Verwendung der IP-Adresse des Mountingziels in derselben Availability Zone wie die verbindende Instance durchführen.

Sie können die entsprechenden Befehle zum Mounting des Dateisystems im Dialogfeld Anhängen einsehen und kopieren.

 Note

Bevor Sie Ihr Dateisystem mounten, müssen Sie eine Regel für die Mount-Zielsicherheitsgruppe hinzufügen, um eingehenden NFS-Zugriff auf Port 2049 von der EC2-Sicherheitsgruppe aus zuzulassen. Weitere Informationen finden Sie unter [Verwenden von VPC-Sicherheitsgruppen](#).

Gehen Sie folgt vor, um die genauen Befehle zum Mounten Ihres EFS-Dateisystems mithilfe der Mount-Ziel-IP-Adresse anzuzeigen und zu kopieren:

1. Öffnen Sie die Amazon Elastic File System-Konsole unter <https://console.aws.amazon.com/efs/>.
2. Wählen Sie in der Amazon-EFS-Konsole das Dateisystem aus, das Sie mounten möchten, um dessen Detailseite anzuzeigen.
3. Um die für dieses Dateisystem zu verwendenden Mountingbefehle anzuzeigen, wählen Sie oben rechts die Option Anhängen aus.
4. Auf dem Bildschirm Anhängen werden die entsprechenden Befehle angezeigt, die zum Mounting des Dateisystems verwendet werden können:

Wählen Sie Über IP mounten aus, um den Befehl zum Mounten des Dateisystems unter Verwendung der IP-Adresse des Mountingziels in der ausgewählten Availability Zone mit einem NFS-Client.

- Mithilfe der IP-Adresse eines Mountingziels im mount-Befehl können Sie mit dem folgenden Befehl ein Dateisystem auf Ihrer Amazon-EC2-Linux-Instance mounten.

```
sudo mount -t nfs -o  
nfsvers=4.1,rsiz=1048576,wsiz=1048576,hard,timeo=600,retrans=2,noresvport mount-  
target-IP:/ /efs
```

- Mithilfe der IP-Adresse eines Mountingziels im mount-Befehl können Sie mit dem folgenden Befehl ein Dateisystem auf Ihrer Amazon-EC2-Mac-Instance, auf der macOS Big Sur ausgeführt wird, mounten.

```
sudo mount -t nfs -o  
nfsvers=4.0,rsiz=65536,wsiz=65536,hard,timeo=600,retrans=2,noresvport,mountport=2049 mount-  
target-IP:/ /efs
```

Important

Sie müssen `mountport=2049` verwenden, um beim Mounten auf EC2-Mac-Instances, auf denen macOS Big Sur ausgeführt wird, erfolgreich eine Verbindung zum EFS-Dateisystem herzustellen.

Montage mit einer IP-Adresse in AWS CloudFormation

Sie können Ihr Dateisystem auch mithilfe einer IP-Adresse in einer CloudFormation Vorlage mounten. Weitere Informationen finden Sie unter [storage-efs-mountfilesystem-ip-addr.config](#) im `awsdocs/`-Repository für von der Community bereitgestellte `elastic-beanstalk-samples` Konfigurationsdateien. [GitHub](#)

Automatisches Mounten von EFS-Dateisystemen

Sie können den EFS-Mount-Helper oder NFS verwenden, um eine Amazon EC2 EC2-Instance so zu konfigurieren, dass beim Start der Instance automatisch ein EFS-Dateisystem bereitgestellt wird.

- Verwenden der EFS-Mountinghilfe:
 - Hängen Sie ein EFS-Dateisystem an, wenn Sie mit dem EC2-Launch-Instance-Assistenten eine neue EC2-Linux-Instance erstellen.
 - Aktualisieren Sie die `/etc/fstab`-EC2-Datei mit einem Eintrag für das EFS-Dateisystem.
- Verwenden von [NFS ohne den EFS-Mount-Helper](#) zur Aktualisierung der `/etc/fstab` EC2-Datei für EC2-Linux- und Mac-Instances.

 Note

Der EFS-Mount-Helper unterstützt kein automatisches Mounten auf EC2-Mac-Instances, auf denen macOS Big Sur oder Monterey ausgeführt wird. Stattdessen können Sie [NFS verwenden, um die Datei /etc/fstab auf einer EC2-Mac-Instance so zu konfigurieren](#), dass ein EFS-Dateisystem automatisch gemountet wird.

Themen

- [Aktivierung der automatischen Installation auf neuen EC2-Linux-Instances](#)
- [Aktivierung der automatischen Installation auf vorhandenen EC2-Linux-Instances](#)
- [Aktivierung des automatischen Mounten auf EC2-Linux- oder Mac-Instances mithilfe von NFS](#)

Aktivierung der automatischen Installation auf neuen EC2-Linux-Instances

Wenn Sie mit dem Amazon EC2-Launch-Instance-Assistenten eine neue EC2-Linux-Instance erstellen, können Sie sie so konfigurieren, dass Ihr Amazon EFS-Dateisystem automatisch bereitgestellt wird. Die EC2-Instance mountet das Dateisystem automatisch bei der zuerst gestarteten Instance und auch bei jedem Neustart.

Bei dieser Methode wird die EFS-Mountinghilfe verwendet, um das Dateisystemupdate der /etc/fstab-Datei auf der EC2-Instance zu mounten. Die Mountinghilfe ist Teil der [amazon-efs-utils](#)-Tools.

 Note

EFS-Dateisysteme unterstützen das Mounten auf EC2-Mac-Instances, auf denen macOS Big Sur oder Monterey ausgeführt wird, beim Instance-Start nicht.

 Note

Sie können Amazon EFS nicht mit Microsoft Windows-basierten EC2-Instances verwenden.

Bevor Sie eine EC2-Instance starten und eine Verbindung zu ihr herstellen können, müssen Sie ein key pair erstellen. Weitere Informationen finden Sie unter [Amazon EC2 EC2-Schlüsselpaare und Amazon EC2 EC2-Instances](#) im Amazon EC2 EC2-Benutzerhandbuch zum Erstellen eines key pair.

So konfigurieren Sie die EC2-Instance zum automatischen Mounting eines EFS-Dateisystems beim Start

1. Öffnen Sie die Amazon-EC2-Konsole unter <https://console.aws.amazon.com/ec2/>.
2. Wählen Sie Launch Instance aus.
3. Suchen Sie unter Schritt 1: Auswählen eines Amazon-Systemabbilds (AMI) ein Amazon Linux-AMI oben in der Liste und klicken Sie auf Select (Auswählen).
4. Klicken Sie unter Schritt 2: Auswählen eines Instance-Typs auf Next: Configure Instance Details (Weiter: Instance-Details konfigurieren).
5. Geben Sie die folgenden Informationen ein unter Step 3: Configure Instance Details (Schritt 3: Konfigurieren von Instance-Details):
 - Wählen Sie unter Netzwerk den Eintrag für dieselbe VPC aus, in der sich das EFS-Dateisystem befindet, in dem Sie mounten.
 - Wählen Sie für Subnet (Subnetz) ein Standardsubnetz in einer beliebigen Availability Zone aus.
 - Wählen Sie unter Dateisysteme das EFS-Dateisystem aus, das Sie mounten möchten. Der Pfad neben der Dateisystem-ID ist der Mounting-Punkt, den die EC2-Instance verwendet, die Sie ändern können.
 - Unter Erweiterte Details werden die User data (Benutzerdaten) automatisch generiert und enthalten die Befehle, die zum Mounten der unter Dateisysteme angegebenen EFS-Dateisysteme erforderlich sind.
6. Wählen Sie Next: Add Storage aus.
7. Wählen Sie Next: Add Tags (Weiter: Tags hinzufügen) aus.
8. Geben Sie der Instance einen Namen und klicken Sie auf Next: Configure Security Group (Weiter: Sicherheitsgruppe konfigurieren).
9. Stellen Sie in Step 6: Configure Security Group (Schritt 6: Sicherheitsgruppe konfigurieren) für Assign a security group (Eine Sicherheitsgruppe zuweisen) Select an existing security group (Eine vorhandene Sicherheitsgruppe auswählen) ein. Wählen Sie die Standardsicherheitsgruppe aus, um sicherzustellen, dass sie auf das EFS-Dateisystem zugreifen kann.

Sie können mit dieser Sicherheitsgruppe nicht über Secure Shell (SSH) auf Ihre EC2-Instance zugreifen. Für den Zugriff über SSH können Sie später die Standardsicherheit bearbeiten und eine Regel hinzufügen, um SSH oder eine neue Sicherheitsgruppe zuzulassen, die SSH zulässt. Sie können die folgenden Einstellungen verwenden:

- Typ: SSH
- Protocol (Protokoll): TCP
- Port-Bereich: 22
- Quelle: Anywhere 0.0.0.0/0

10. Klicken Sie auf Review and Launch.

11. Wählen Sie Launch (Starten) aus.

12. Aktivieren Sie das Kontrollkästchen für das Schlüsselpaar, das Sie erstellt haben, und klicken Sie dann auf Launch Instances (Instances starten).

Ihre EC2-Instance ist jetzt so konfiguriert, dass das EFS-Dateisystem beim Start und bei jedem Neustart gemountet wird.

Aktivierung der automatischen Installation auf vorhandenen EC2-Linux-Instances

Die `/etc/fstab`-Datei enthält Informationen zu Dateisystemen. Mit dem Befehl `mount -a`, der während des Instance-Starts ausgeführt wird, werden die in `/etc/fstab` aufgeführten Dateisysteme gemountet. In diesem Verfahren aktualisieren Sie die `/etc/fstab` auf einer Amazon EC2 EC2-Linux-Instance manuell, sodass die Instance den EFS-Mount-Helper verwendet, um ein EFS-Dateisystem automatisch neu zu mounten, wenn die Instance neu gestartet wird.

Note

EFS-Dateisysteme unterstützen kein automatisches Mounten `/etc/fstab` mit dem EFS-Mount-Helper auf EC2-Mac-Instances, auf denen macOS Big Sur oder Monterey ausgeführt wird. Stattdessen können Sie [NFS mit /etc/fstab](#) verwenden, um das Dateisystem automatisch auf EC2-Mac-Instances zu mounten, auf denen macOS Big Sur und Monterey ausgeführt werden.

Bei dieser Methode wird die EFS-Mountinghilfe verwendet, um das Dateisystem zu mounten. Die Mountinghilfe ist Teil der `amazon-efs-utils`-Tools.

Die `amazon-efs-utils` Tools sind für die Installation auf Amazon Linux und Amazon Linux 2 Amazon Machine Images (AMIs) verfügbar. Mehr über `amazon-efs-utils` erfahren Sie unter [Den](#)

[Amazon EFS-Client installieren](#). Wenn Sie eine andere Linux-Verteilung wie Red Hat Enterprise Linux (RHEL) verwenden, erstellen und installieren Sie sie `amazon-efs-utils` manuell. Weitere Informationen finden Sie unter [Installation des Amazon-EFS-Clients auf anderen Linux-Distributionen](#).

Voraussetzungen

Die folgenden Anforderungen müssen erfüllt sein, bevor Sie dieses Verfahren erfolgreich implementieren können:

- Sie haben bereits das EFS-Dateisystem erstellt, das automatisch erneut bereitgestellt werden soll. Weitere Informationen finden Sie unter [Schnelles Erstellen mit der Konsole](#).
- Sie haben bereits die EC2-Linux-Instance erstellt, die Sie für das automatische erneute Mounting eines EFS-Dateisystems konfigurieren möchten.
- Die EFS-Mountinghilfe ist auf der EC2-Linux-Instance installiert. Weitere Informationen finden Sie unter [Den Amazon EFS-Client installieren](#).

Aktualisieren Sie die Datei `/etc/fstab`

Führen Sie die folgenden Schritte aus, um `/etc/fstab` auf einer EC2-Linux-Instance zu aktualisieren, sodass die Instance den EFS-Mount-Helper verwendet, um ein EFS-Dateisystem beim Neustart der Instanz automatisch neu zu mounten.

So aktualisieren Sie die `/etc/fstab`-Datei in Ihrer EC2-Instance

1. Stellen Sie eine Verbindung zu Ihrer EC2-Instance her. Weitere Informationen finden Sie unter [Mit Ihrer EC2-Instance verbinden](#) im Amazon-EC2-Benutzerhandbuch.
2. Öffnen Sie die Datei `/etc/fstab` in einem Editor.
3. Automatisches Mounting des EFS-Dateisystems mithilfe der IAM-Autorisierung oder eines EFS-Zugangspunkts:
 - Um eine EC2-Instance, die über ein Instance-Profil verfügt, automatisch mit IAM-Autorisierung zu mounten, fügen Sie der Datei die folgende Zeile hinzu. `/etc/fstab`

```
file-system-id:/ efs-mount-point efs _netdev,noresvport,tls,iam 0 0
```

- Für ein automatisches Mounting mit IAM-Autorisierung zu einer Linux-Instance mithilfe einer Anmeldeinformationendatei fügen Sie der `/etc/fstab`-Datei die folgende Zeile hinzu.


```
file-system-id:/ efs-mount-point efs  
_netdev,noresvport,tls,iam,awsprofile=namedprofile 0 0
```

- Wenn Sie ein Dateisystem mithilfe eines EFS-Zugangspunkts automatisch mounten möchten, fügen Sie der `/etc/fstab`-Datei die folgende Zeile hinzu.

```
file-system-id:/ efs-mount-point efs _netdev,noresvport,tls,accesspoint=access-  
point-id 0 0
```

Warning

Verwenden Sie beim automatischen Mounting Ihres Dateisystems die Option `_netdev`, um es als Netzwerkdateisystem zu identifizieren. Wenn `_netdev` fehlt, reagiert die EC2-Instance möglicherweise nicht mehr. Der Grund dafür ist, dass zuerst das Netzwerk auf der Datenverarbeitungs-Instance gestartet worden sein muss. Die Netzwerkdateisysteme müssen danach initialisiert werden. Weitere Informationen finden Sie unter [Automatisches Mounting schlägt fehl und die Instance reagiert nicht](#).

Weitere Informationen erhalten Sie unter [Mounting mit IAM-Autorisierung](#) und [Mounting mit EFS-Zugangspunkten](#).

4. Speichern Sie die Änderungen an der Datei.

Note

In einigen Fällen muss Ihre EC2-Instance möglicherweise unabhängig vom Status Ihres bereitgestellten EFS-Dateisystems gestartet werden. Fügen Sie in solchen Fällen die `nofail`-Option zum Eintrag Ihres Dateisystems in Ihrer `/etc/fstab`-Datei hinzu.

Die Codezeile, die Sie der Datei `/etc/fstab` hinzugefügt haben, führt Folgendes aus.

Feld	Description
<i>file-system-id</i> :/	Die ID für Ihr EFS-Dateisystem. Sie können diese ID von der Konsole oder programmgesteuert von der CLI oder einem AWS SDK abrufen.
<i>efs-mount-point</i>	Der Mountingpunkt für das EFS-Dateisystem auf Ihrer EC2-Instance.
efs	Der Typ des Dateisystems. Wenn Sie die Mountinghilfe verwenden, ist dieser Typ immer efs.
mount options	Mountingoptionen für das Dateisystem. Dies ist eine durch Kommata getrennte Liste der folgenden Optionen: • <code>_netdev</code> – Diese Option teilt dem Betriebssystem mit, dass das Dateisystem sich auf einem Gerät befindet, das Netzwerkzugriff erfordert. Diese Option verhindert, dass die Instance das Dateisystem mountet, bis das Netzwerk auf dem Client aktiviert wurde. • <code>noresvport</code> – Teilt dem NFS-Client mit, einen neuen Quellanschluss für Transmission Control Protocol (TCP) zu verwenden, wenn erneut eine Netzwerkverbindung eingerichtet wird. Dadurch wird der ununterbrochene Zugriff des EFS-Dateisystems nach einem Netzwerkwiderherstellungsereignis sichergestellt. • <code>tls</code> – Ermöglicht die Verschlüsselung von Daten während der Übertragung. • <code>iam</code>— Verwenden Sie diese Option, um mit IAM-Autorisierung eine EC2-Instance zu mounten, die über ein Instanzprofil verfügt. Die Verwendung der Mounting-Option <code>iam</code> erfordert auch die Verwendung der <code>tls</code>-Option. Weitere Informationen finden Sie unter Verwendung von IAM zur Steuerung des Zugriffs auf Dateisysteme. • <code>awsprofile= <i>namedprofile</i></code> – Verwenden Sie diese Option mit den Optionen <code>iam</code> und <code>tls</code>, um mit IAM-Autorisierung für eine Linux-Instance mithilfe einer Anmeldeinformationendatei zu mounten. Weitere Hinweise zu EFS-Zugangspunkten finden Sie unter Verwendung von IAM zur Steuerung des Zugriffs auf Dateisysteme.

Feld	Description
	<code>accesspoint= <i>access-point-id</i></code> – Verwenden Sie diese Option mit der <code>tls</code>-Option zum Mounten über einen EFS-Zugangspunkt. Weitere Hinweise zu EFS-Zugangspunkten finden Sie unter Arbeiten mit Zugriffspunkten.
0	Ein Wert ungleich Null gibt an, dass das Dateisystem von dump gesichert werden soll. Für EFS muss dieser Wert 0 sein.
0	Die Reihenfolge, in der fsck die Dateisysteme beim Systemstart prüft. Bei EFS-Dateisystemen sollte dieser Wert 0 lauten; dieser gibt an, dass fsck beim Start nicht ausgeführt werden soll.

Aktivierung des automatischen Mounten auf EC2-Linux- oder Mac-Instances mithilfe von NFS

Verwenden von NFS ohne den EFS-Mount-Helper zur Aktualisierung der Amazon EC2 `/etc/fstab` EC2-Datei für EC2-Linux- und Mac-Instances.

So aktualisieren Sie die `/etc/fstab`-Datei in der EC2-Instance

1. Stellen Sie eine Verbindung zu Ihrer EC2- Instance her. Weitere Informationen finden Sie unter [Mit Ihrer EC2-Instance verbinden](#) im Amazon-EC2-Benutzerhandbuch.
2. Öffnen Sie die Datei `/etc/fstab` in einem Editor.
3. Wenn Sie ein Dateisystem mithilfe von NFS anstelle der EFS-Mountinghilfe mounten möchten, fügen Sie der `/etc/fstab`-Datei die folgende Zeile hinzu.
 - `file_system_id` Ersetzen Sie es durch die ID des Dateisystems, das Sie mounten.
 - `aws-region` Ersetzen Sie durch die AWS-Region , in der sich das Dateisystem befindet, z. `us-east-1` B.
 - Ersetzen Sie `mount_point` durch den Mountingpunkt des Dateisystems.

```
file_system_id.efs.aws-region.amazonaws.com:/ mount_point nfs4
nfsvers=4.1,rsize=1048576,wsiz=1048576,hard,timeo=600,retrans=2,noresvport,_netdev
0 0
```

Die Codezeile, die Sie der Datei `/etc/fstab` hinzugefügt haben, führt Folgendes aus.

Feld	Description
<i>file-system-id</i> :/	Die ID für Ihr EFS-Dateisystem. Sie können diese ID von der Konsole oder programmgesteuert von der CLI oder einem AWS SDK abrufen.
<i>efs-mount-point</i>	Der Mountingpunkt für das EFS-Dateisystem auf Ihrer EC2-Instance.
<code>nfs4</code>	Gibt den Dateisystemtyp an.
<code>mount options</code>	Die kommagetrennte Liste der Mountingoptionen für das Dateisystem: • <code>nfsvers=4.1</code> – Gibt die Verwendung von NFS v4.1 an. • <code>rsiz=1048576</code> – Legt zur Verbesserung der Leistung die maximale Anzahl von Datenbytes fest, die der NFS-Client für jede Netzwerk-READ-Anforderung empfangen kann, wenn er Daten aus einer Datei in einem EFS-Dateisystem liest. Die größtmögliche Anzahl lautet 1048576. • <code>wsiz=1048576</code> – Legt zur Verbesserung der Leistung die maximale Anzahl von Datenbytes fest, die der NFS-Client für jede Netzwerk-WRITE-Anforderung senden kann, wenn er Daten in eine Datei in einem EFS-Dateisystem schreibt. Die größtmögliche Anzahl lautet 1048576. • <code>hard</code> – Legt das Wiederherstellungsverhalten des NFS-Clients nach dem Timeout einer NFS-Anforderung fest, damit NFS-Anforderungen so lange wiederholt werden, bis der Server antwortet. Zur Sicherstellung der Datenintegrität wird die Verwendung der dauerhaften Mountingoption (<code>hard</code>) empfohlen. Wenn Sie ein <code>soft</code>-Mount verwenden, legen Sie den <code>timeo</code>-Parameter auf mindestens 150 Zehntelsekunden (15 Sekunden) fest. Dadurch wird das Risiko einer Datenbeschädigung verringert, die bei <code>Soft</code>-Mounts inhärent ist. • <code>timeo=600</code> – Legt den Timeout-Wert, der angibt, wie lange der NFS-Client auf eine Antwort wartet, bis er eine Anforderung wiederholt, auf 600 Zehntelsekunden (60 Sekunden) fest. Wenn Sie den Timeout-Parameter (<code>timeo</code>) ändern müssen, empfehlen

Feld	Description
	wir, dass Sie einen Wert von mindestens 150, entsprechend 15 Sekunden, verwenden. Dadurch wird eine verringerte Leistung vermieden. • <code>retrans=2</code> – Legt die Anzahl der Anforderungswiederholungen eines NFS-Clients vor dem Versuch einer weiteren Wiederherstellungsaktion auf 2 fest. • <code>noresvport</code> – Teilt dem NFS-Client mit, einen neuen Quellanschluss für Transmission Control Protocol (TCP) zu verwenden, wenn erneut eine Netzwerkverbindung eingerichtet wird. Dadurch wird der ununterbrochene Zugriff des EFS-Dateisystems nach einem Netzwerkwiderherstellungsereignis sichergestellt. • <code>_netdev</code> – Hindert den Client an dem Versuch, das EFS-Dateisystem zu mounten, bis das Netzwerk aktiviert wurde.
0	Gibt den dump Wert an; 0 weist das Dienstprogramm dump an, das Dateisystem nicht zu sichern.
0	Weist das Dienstprogramm fsck an, beim Start nicht ausgeführt zu werden.

Aufheben des Mountings von Dateisystemen

Bevor Sie ein Dateisystem löschen, empfehlen wir, dass Sie sein Mounting auf allen Amazon-EC2-Instances aufheben, mit denen es verbunden ist. Sie können das Mounting eines Dateisystems auf der Amazon-EC2-Instance aufheben, indem Sie den Befehl `umount` auf der Instance selbst ausführen. Sie können ein EFS-Dateisystem nicht über das AWS CLI, das oder über eines der AWS-Managementkonsole AWS SDKs unmounten. Um ein EFS-Dateisystem auszuhängen, das mit einer EC2-Instance unter Linux verbunden ist, verwenden Sie den `umount` Befehl wie folgt:

```
umount /mnt/efs
```

Wir empfehlen, dass Sie keine anderen `umount`-Optionen angeben. Vermeiden Sie die Einstellung anderer `umount`-Optionen, die sich von den Standardwerten unterscheiden.

Sie können überprüfen, ob Ihr EFS-Dateisystem nicht bereitgestellt wurde, indem Sie den `df` Befehl ausführen. Mit diesem Befehl werden die Datenträgnutzungsstatistiken für die Dateisysteme angezeigt, die derzeit auf der Linux-basierten Amazon-EC2-Instance gemountet werden. Wenn das EFS-Dateisystem, das Sie unmounten möchten, nicht in der `df` Befehlsausgabe aufgeführt ist, bedeutet dies, dass das Dateisystem nicht bereitgestellt wurde.

Example— Identifizieren Sie den Mount-Status eines EFS-Dateisystems und hängen Sie es aus

```
$ df -T
Filesystem Type 1K-blocks Used Available Use% Mounted on
/dev/sda1 ext4 8123812 1138920 6884644 15% /
availability-zone.file-system-id.efs.aws-region.amazonaws.com :/ nfs4 9007199254740992
0 9007199254740992 0% /mnt/efs
```

```
$ umount /mnt/efs
```

```
$ df -T
```

```
Filesystem Type 1K-blocks Used Available Use% Mounted on
/dev/sda1 ext4 8123812 1138920 6884644 15% /
```

Tutorial: Erstellen Sie ein EFS-Dateisystem und mounten Sie es auf einer EC2-Instance mit dem AWS CLI

Erstellen Sie ein verschlüsseltes EFS-Dateisystem, mounten Sie es auf einer EC2-Instance in Ihrer VPC und testen Sie das Setup mit dem AWS CLI

Note

In der [Erste Schritte](#) Anleitung verwenden Sie die Konsole, um Amazon EC2- und EFS-Ressourcen zu erstellen. In diesem Tutorial verwenden Sie die AWS CLI um dasselbe zu tun — hauptsächlich, um sich mit der Amazon EFS-API vertraut zu machen.

In diesem Tutorial erstellen Sie die folgenden AWS Ressourcen in Ihrem Konto:

- Amazon EC2-Ressourcen:

- Zwei Sicherheitsgruppen (für Ihre EC2-Instance und Ihr EFS-Dateisystem).

Sie fügen diesen Sicherheitsgruppen Regeln hinzu, um entsprechenden ein-/ausgehenden Zugriff zu genehmigen. Auf diese Weise können Sie über das Mount-Ziel eine Verbindung EC2instance zum Dateisystem herstellen, indem Sie einen standardmäßigen NFSv4 .1 TCP-Port verwenden.

- Eine EC2-Instance in Ihrer VPC.
- Amazon-EFS-Ressourcen:
 - Ein Dateisystem.
 - Ein Mounting-Ziel für Ihr Dateisystem.

Um Ihr Dateisystem auf einer EC2-Instance bereitzustellen, müssen Sie ein Mounting-Ziel in Ihrer VPC erstellen. Sie können ein Mounting-Ziel in jeder Availability Zone in Ihrer VPC erstellen. Weitere Informationen finden Sie unter [So funktioniert Amazon EFS](#).

Anschließend testen Sie das Dateisystem auf Ihrer EC2-Instance. Der Bereinigungsschritt am Ende des Tutorials enthält Informationen zum Entfernen dieser Ressourcen.

Das Tutorial erstellt all diese Ressourcen in der Region USA West (Oregon) (us-west-2). Was auch immer AWS-Region Sie verwenden, achten Sie darauf, es konsistent zu verwenden. Alle Ihre Ressourcen — Ihre VPC-, EC2-Ressourcen und EFS-Ressourcen — müssen sich im selben System befinden. AWS-Region

Themen

- [Voraussetzungen](#)
- [Einrichtung des AWS CLI](#)
- [Schritt 1: EC2-Ressourcen erstellen](#)
- [Schritt 2: EFS-Ressourcen erstellen](#)
- [Schritt 3: Mounting des Dateisystems auf der EC2 Instance und Test](#)
- [Schritt 4: Bereinigen](#)

Voraussetzungen

- Sie können sich mit Ihren Stammdaten an der Konsole anmelden und AWS-Konto die Übung „Erste Schritte“ ausprobieren. AWS Identity and Access Management (IAM) empfiehlt jedoch, nicht

die Root-Anmeldeinformationen Ihres AWS-Konto zu verwenden. Erstellen Sie stattdessen einen Administrator-Benutzer in Ihrem Konto, und verwenden Sie dessen Anmeldeinformationen für die Verwaltung von Ressourcen in Ihrem Konto. Weitere Informationen finden Sie unter [Zuweisen von AWS-Konto Zugriff für einen IAM Identity Center-Benutzer](#) im AWS IAM Identity Center Benutzerhandbuch.

- Sie können eine Standard-VPC oder eine benutzerdefinierte VPC, die Sie in Ihrem Konto erstellt haben, verwenden. Für diese Anleitung funktioniert die Standard-VPC-Konfiguration. Wenn Sie jedoch eine benutzerdefinierte VPC verwenden, überprüfen Sie Folgendes:
 - DNS-Hostnamen sind aktiviert. Weitere Informationen finden Sie unter [DNS-Attribute in Ihrer VPC](#) im Amazon VPC-Benutzerhandbuch.
 - Das Internet-Gateway ist mit Ihrer VPC verbunden. Weitere Informationen finden Sie unter [Verbinden mit dem Internet über ein Internet-Gateway](#) im Amazon-VPC-Benutzerhandbuch.
 - Die VPC-Subnetze sind so konfiguriert, dass sie öffentliche IP-Adressen für Instances anfordern, die in den VPC-Subnetzen gestartet wurden. Weitere Informationen finden Sie unter [IP-Adressierung für Ihre VPCs und Subnetze](#) im Amazon VPC-Benutzerhandbuch.
 - Die VPC-Routing-Tabelle enthält eine Regel zum Senden des gesamten Internet-Datenverkehrs an das Internet-Gateway.
- Sie müssen das Admin-Benutzerprofil einrichten AWS CLI und hinzufügen.

Einrichtung des AWS CLI

Verwenden Sie die folgenden Anweisungen, um das AWS CLI Benutzerprofil einzurichten.

Um das einzurichten AWS CLI

1. Herunterladen und Konfigurieren von AWS CLI. Anweisungen finden Sie unter [Erste Schritte mit dem AWS CLI](#) im AWS Command Line Interface Benutzerhandbuch.
2. Richten Sie Profile ein.

Sie speichern Benutzeranmeldeinformationen in der AWS CLI config Datei. Die CLI-Beispielfehle in diesem Tutorial spezifizieren das Adminuser-Profil. Erstellen Sie das adminuser-Profil in der Datei config. Sie können auch das Administrator-Benutzerprofil als Standard in der Datei config festlegen, wie nachfolgend dargestellt.

```
[profile adminuser]
aws_access_key_id = admin user access key ID
```



```
aws_secret_access_key = admin user secret access key
region = us-west-2

[default]
aws_access_key_id = admin user access key ID
aws_secret_access_key = admin user secret access key
region = us-west-2
```

Das vorherige Profil legt auch den Standard AWS-Region fest. Wenn Sie im CLI-Befehl keine Region angeben, wird die Region us-west-2 angenommen.

3. Überprüfen Sie die Einrichtung, indem Sie den folgenden Befehl in die Befehlszeile eingeben. Beide Befehle stellen nicht explizit Anmeldeinformationen bereit, daher werden die Anmeldeinformationen des Standardprofils verwendet.

- Probieren Sie den Hilfebefehl aus.

Sie können auch das Benutzerprofil explizit angeben, indem Sie den Parameter `--profile` hinzufügen.

```
aws help
```

```
aws help \
--profile adminuser
```

Schritt 1: EC2-Ressourcen erstellen

In diesem Schritt führen Sie folgende Aufgaben aus:

- Erstellen von zwei Sicherheitsgruppen
- Hinzufügen von Regeln zu den Sicherheitsgruppen, um weiteren Zugriff zu autorisieren.
- Starten einer EC2-Instance. Im nächsten Schritt erstellen und mounten Sie ein EFS-Dateisystem auf dieser Instanz.

Schritt 1.1: Erstellen von zwei Sicherheitsgruppen

In diesem Abschnitt erstellen Sie Sicherheitsgruppen in Ihrer VPC für Ihre EC2-Instance und Ihr EFS-Mount-Ziel. Später im Tutorial weisen Sie diese Sicherheitsgruppen einer EC2-Instance und

einem EFS-Mount-Ziel zu. Informationen zu Sicherheitsgruppen finden Sie unter [Amazon EC2-Sicherheitsgruppen für Linux-Instances](#).

So erstellen Sie Sicherheitsgruppen:

1. Erstellen Sie mithilfe des CLI-Befehls `create-security-group` zwei Sicherheitsgruppen:
 - a. Erstellen Sie eine Sicherheitsgruppe (`efs-walkthrough1-ec2-sg`) für Ihre EC2-Instance, und geben Sie Ihre VPC-ID an.

```
$ aws ec2 create-security-group \  
--region us-west-2 \  
--group-name efs-walkthrough1-ec2-sg \  
--description "Amazon EFS walkthrough 1, SG for EC2 instance" \  
--vpc-id vpc-id-in-us-west-2 \  
--profile adminuser
```

Notieren Sie sich die Sicherheitsgruppen-ID. Nachfolgend finden Sie eine Beispielantwort.

```
{  
  "GroupId": "sg-aexample"  
}
```

Sie können die VPC-ID mithilfe des folgenden Befehls suchen.

```
$ aws ec2 describe-vpcs
```

- b. Erstellen Sie eine Sicherheitsgruppe (`efs-walkthrough1-mt-sg`) für Ihr EFS-Mount-Ziel. Sie müssen Ihre VPC-ID bereitstellen.

```
$ aws ec2 create-security-group \  
--region us-west-2 \  
--group-name efs-walkthrough1-mt-sg \  
--description "Amazon EFS walkthrough 1, SG for mount target" \  
--vpc-id vpc-id-in-us-west-2 \  
--profile adminuser
```

Notieren Sie sich die Sicherheitsgruppen-ID. Nachfolgend finden Sie eine Beispielantwort.

```
{
```

```
"GroupId": "sg-aexample"  
}
```

2. Überprüfen Sie die Sicherheitsgruppen.

```
aws ec2 describe-security-groups \  
--group-ids list of security group IDs separated by space \  
--profile adminuser \  
--region us-west-2
```

Beide sollten nur eine Regel für ausgehenden Datenverkehr aufweisen, die das Ausgehen des gesamten Datenverkehrs zulässt.

Im nächsten Abschnitt autorisieren Sie weiteren Zugriff, mit dem Folgendes möglich ist:

- Herstellen einer Verbindung mit Ihrer EC2-Instance.
- Aktivieren Sie den Datenverkehr zwischen einer EC2-Instance und einem EFS-Mount-Ziel (dem Sie diese Sicherheitsgruppen später in diesem Tutorial zuordnen).

Schritt 1.2: Fügen Sie den Sicherheitsgruppen Regeln hinzu, um den Zugriff zu autorisieren inbound/outbound

In diesem Schritt fügen Sie Regeln zu den Sicherheitsgruppen zur Genehmigung von ein-/ausgehendem Zugriff hinzu.

So fügen Sie Regeln hinzu:

1. Autorisieren Sie eingehende Secure Shell (SSH)-Verbindungen in der Sicherheitsgruppe für Ihre EC2-Instance (*efs-walkthrough1-ec2-sg*), sodass Sie mithilfe von SSH von einem beliebigen Host eine Verbindung mit Ihrer EC2-Instance herstellen können.

```
$ aws ec2 authorize-security-group-ingress \  
--group-id id of the security group created for EC2 instance \  
--protocol tcp \  
--port 22 \  
--cidr 0.0.0.0/0 \  
--profile adminuser \  
--region us-west-2
```

Überprüfen Sie, ob der Sicherheitsgruppe die Regel für ein- und ausgehenden Datenverkehr hinzugefügt wurde.

```
aws ec2 describe-security-groups \  
--region us-west-2 \  
--profile adminuser \  
--group-id security-group-id
```

2. Autorisieren Sie den eingehenden Zugriff auf die Sicherheitsgruppe für das EFS-Mount-Ziel (efs-walkthrough1-mt-sg).

Führen Sie in der Befehlszeile den folgenden AWS CLI `authorize-security-group-ingress` Befehl mit dem Administratorprofil aus, um die Regel für eingehenden Datenverkehr hinzuzufügen.

```
$ aws ec2 authorize-security-group-ingress \  
--group-id ID of the security group created for Amazon EFS mount target \  
--protocol tcp \  
--port 2049 \  
--source-group ID of the security group created for EC2 instance \  
--profile adminuser \  
--region us-west-2
```

3. Überprüfen Sie, ob beide Sicherheitsgruppen jetzt Zugriff auf eingehenden Datenverkehr autorisieren.

```
aws ec2 describe-security-groups \  
--group-names efs-walkthrough1-ec2-sg efs-walkthrough1-mt-sg \  
--profile adminuser \  
--region us-west-2
```

Schritt 1.3: Starten einer EC2-Instance

In diesem Schritt starten Sie eine EC2-Instance.

Starten Sie EC2-Instances wie folgt:

1. Tragen Sie die folgenden Informationen zusammen, die Sie beim Starten einer EC2-Instance bereitstellen müssen:

- Schlüsselpaar-Name. Anweisungen zum Erstellen eines key pair finden Sie unter [Erstellen eines key pair für Ihre Amazon EC2 EC2-Instance](#) im Amazon EC2 EC2-Benutzerhandbuch.
- Die ID des Amazon Machine Image (AMI), das Sie starten möchten.

Der AWS CLI Befehl, mit dem Sie eine EC2-Instance starten, erfordert die ID des Amazon Machine Image (AMI), das Sie bereitstellen möchten, als Parameter. In dieser Übung wird das Amazon Linux HVM AMI verwendet.

 Note

Sie können die meisten Linux-basierten AMIs Anwendungen verwenden. Wenn Sie ein anderes Linux-AMI verwenden, stellen Sie sicher, dass Sie den Paketmanager Ihrer Verteilung für die Installation des NFS-Clients auf der Instance verwenden. Außerdem müssen Sie bei Bedarf möglicherweise Softwarepakete hinzufügen.

Für das Amazon Linux HVM AMI finden Sie die neuesten Informationen IDs unter [Amazon Linux AMI](#). Sie wählen den ID-Wert aus der Amazon IDs Linux-AMI-Tabelle wie folgt aus:

- Wählen Sie die Region US West Oregon (USA West (Oregon)) aus. In dieser Anleitung wird davon ausgegangen, dass Sie alle Ressourcen in der Region USA West (Oregon) (us-west-2) erstellen.
- Wählen Sie den Typ EBS-backed HVM 64-bit (da Sie im CLI-Befehl den Instance-Typ `t2.micro` angeben, der den Instance-Speicher nicht unterstützt).
- ID der Sicherheitsgruppe, die Sie für eine EC2-Instance erstellt haben.
- AWS-Region. Diese Anleitung verwendet die Region `us-west-2`.
- Die ID Ihres VPC-Subnetzes, in dem Sie die Instance starten möchten. Mit dem Befehl `describe-subnets` erhalten Sie eine Liste der Subnetze.

```
$ aws ec2 describe-subnets \
--region us-west-2 \
--filters "Name=vpc-id,Values=vpc-id" \
--profile adminuser
```

Nachdem Sie die Subnetz-ID ausgewählt haben, notieren Sie sich die folgenden Werte aus dem `describe-subnets`-Ergebnis:

- Subnetz-ID – Sie benötigen diesen Wert, wenn Sie ein Mounting-Ziel erstellen. In dieser Übung erstellen Sie ein Mounting-Ziel im selben Subnetz, in dem Sie eine EC2-Instance starten.
 - Availability Zone des Subnetzes – Sie benötigen diesen Wert, um Ihren DNS-Namen für das Mounting-Ziel zu konstruieren, den Sie zum Mounten eines Dateisystems in der EC2-Instance verwenden.
2. Führen Sie den folgenden AWS CLI `run-instances` Befehl aus, um eine EC2-Instance zu starten.

```
$ aws ec2 run-instances \  
--image-id AMI ID \  
--count 1 \  
--instance-type t2.micro \  
--associate-public-ip-address \  
--key-name key-pair-name \  
--security-group-ids ID of the security group created for EC2 instance \  
--subnet-id VPC subnet ID \  
--region us-west-2 \  
--profile adminuser
```

3. Notieren Sie die Instance-ID an, die vom `run-instances`-Befehl ausgegeben wird.
4. Die EC2-Instance, die Sie erstellt haben, muss einen öffentlichen DNS-Namen haben, den Sie für die Verbindung mit der EC2-Instance und zum Mounting des Dateisystems verwenden. Der öffentliche DNS-Name hat die Form:

```
ec2-xx-xx-xx-xxx.compute-1.amazonaws.com
```

Führen Sie den folgenden CLI-Befehl aus, und notieren Sie sich den öffentlichen DNS-Namen.

```
aws ec2 describe-instances \  
--instance-ids EC2 instance ID \  
--region us-west-2 \  
--profile adminuser
```

Wenn Sie den öffentlichen DNS-Namen nicht finden, überprüfen Sie die Konfiguration der VPC, in der Sie die EC2-Instance gestartet haben. Weitere Informationen finden Sie unter [Voraussetzungen](#).

5. (Optional) Weisen Sie der erstellten EC2-Instance einen Namen zu. Fügen Sie dazu einen Tag mit dem Schlüsselnamen und -wert dem Namen hinzu, den Sie der Instance zuweisen möchten. Dazu führen Sie den folgenden AWS CLI `create-tags` Befehl aus.

```
$ aws ec2 create-tags \  
--resources EC2-instance-ID \  
--tags Key=Name,Value=Provide-instance-name \  
--region us-west-2 \  
--profile adminuser
```

Schritt 2: EFS-Ressourcen erstellen

In diesem Schritt führen Sie folgende Aufgaben aus:

- Erstellen Sie ein verschlüsseltes EFS-Dateisystem.
- Aktivieren Sie die Lebenszyklusverwaltung.
- Erstellen Sie ein Mount-Ziel in der Availability Zone, in der Sie Ihre EFS-Instanz gestartet haben.

Schritt 2.1: Erstellen Sie ein EFS-Dateisystem

In diesem Schritt erstellen Sie ein EFS-Dateisystem. Notieren Sie sich die `FileSystemId`, die Sie später brauchen, wenn Sie im nächsten Schritt Mounting-Ziele für das Dateisystem erstellen.

Erstellen Sie ein Dateisystem wie folgt:

- Erstellen Sie ein Dateisystem mit dem optionalen Name-Tag.
 - a. Führen Sie an der Befehlszeile den folgenden AWS `create-file-system` CLI-Befehl aus.

```
$ aws efs create-file-system \  
--encrypted \  
--creation-token FileSystemForWalkthrough1 \  
--tags Key=Name,Value=SomeExampleNameValue \  
--region us-west-2 \  
--profile adminuser
```

Sie erhalten die folgende Antwort.

```
{
  "OwnerId": "111122223333",
  "CreationToken": "FileSystemForWalkthrough1",
  "FileSystemId": "fs-c657c8bf",
  "CreationTime": 1548950706.0,
  "LifecycleState": "creating",
  "NumberOfMountTargets": 0,
  "SizeInBytes": {
    "Value": 0,
    "ValueInIA": 0,
    "ValueInStandard": 0
  },
  "PerformanceMode": "generalPurpose",
  "Encrypted": true,
  "KmsKeyId": "arn:aws:kms:us-west-2:111122223333:a5c11222-7a99-43c8-9dcc-
  abcdef123456",
  "ThroughputMode": "bursting",
  "Tags": [
    {
      "Key": "Name",
      "Value": "SomeExampleNameValue"
    }
  ]
}
```

- b. Notieren Sie sich den `FileSystemId`-Wert. Sie benötigen diesen Wert, wenn Sie unter [Schritt 2.3: Erstellen Sie ein Mounting-Ziel](#) ein Mounting-Ziel für dieses Dateisystem erstellen.

Schritt 2.2: Aktivieren Sie das Lebenszyklusmanagement

In diesem Schritt aktivieren Sie das Lebenszyklusmanagement auf Ihrem Dateisystem, um die EFS-Speicherklasse Infrequent Access (IA) zu verwenden. Weitere Informationen hierzu finden Sie unter [Verwaltung des Speicherlebenszyklus](#) und [EFS-Speicherklassen](#).

So aktivieren Sie das Lebenszyklusmanagement

- Führen Sie in der Befehlszeile den folgenden AWS CLI `put-lifecycle-configuration` Befehl aus.

```
$ aws efs put-lifecycle-configuration \
```



```
--file-system-id fs-c657c8bf \  
--lifecycle-policies TransitionToIA=AFTER_30_DAYS \  
--region us-west-2 \  
--profile adminuser
```

Sie erhalten die folgende Antwort.

```
{  
  "LifecyclePolicies": [  
    {  
      "TransitionToIA": "AFTER_30_DAYS"  
    }  
  ]  
}
```

Schritt 2.3: Erstellen Sie ein Mounting-Ziel

In diesem Schritt erstellen Sie ein Mounting-Ziel für Ihr Dateisystem in der Availability Zone, in der Sie Ihre EC2-Instance gestartet haben.

1. Stellen Sie sicher, dass Sie die folgenden Informationen haben:

- ID des Dateisystems (zum Beispiel `fs-example`), für das Sie das Mounting-Ziel erstellen.
- VPC-Subnetz-ID, in der Sie die EC2-Instance gestartet haben. [Schritt 1: EC2-Ressourcen erstellen](#)

Für dieses Tutorial erstellen Sie das Mount-Ziel in demselben Subnetz, in dem Sie die EC2-Instance gestartet haben. Daher benötigen Sie die Subnetz-ID (z. B.). `subnet-example`

- ID der Sicherheitsgruppe, die Sie im vorhergehenden Schritt für das Mounting-Ziel erstellt haben.
2. Führen Sie in der Befehlszeile den folgenden Befehl aus. AWS CLI `create-mount-target`

```
$ aws efs create-mount-target \  
--file-system-id file-system-id \  
--subnet-id subnet-id \  
--security-group ID-of-the security-group-created-for-mount-target \  
--region us-west-2 \  

```

```
--profile adminuser
```

Sie erhalten die folgende Antwort.

```
{
  "MountTargetId": "fsmt-example",
  "NetworkInterfaceId": "eni-example",
  "FileSystemId": "fs-example",
  "PerformanceMode": "generalPurpose",
  "LifecycleState": "available",
  "SubnetId": "fs-subnet-example",
  "OwnerId": "account-id",
  "IpAddress": "xxx.xx.xx.xxx"
}
```

3. Sie können auch den `describe-mount-targets`-Befehl verwenden, um Beschreibungen der Mounting-Ziele zu erhalten, die Sie auf einem Dateisystem erstellt haben.

```
$ aws efs describe-mount-targets \
--file-system-id file-system-id \
--region us-west-2 \
--profile adminuser
```

Schritt 3: Mounting des Dateisystems auf der EC2 Instance und Test

In diesem Schritt führen Sie folgende Aufgaben aus:

- Sammeln Sie die erforderlichen Informationen.
- Installieren Sie den NFS-Client auf Ihrer EC2-Instance.
- Mounten Sie das Dateisystem auf Ihrer EC2-Instance und testen Sie es.

Themen

- [Schritt 3.1: Sammeln Sie Informationen](#)
- [Schritt 3.2: Installieren Sie den NFS-Client auf Ihrer EC2-Instance](#)
- [Schritt 3.3: Mounten Sie das Dateisystem auf Ihrer EC2-Instance und testen Sie es](#)

Schritt 3.1: Sammeln Sie Informationen

Stellen Sie sicher, dass Sie die folgenden Informationen haben, wenn Sie die Schritte in diesem Abschnitt ausführen:

- Öffentlicher DNS-Name Ihrer EC2-Instance in folgendem Format:

```
ec2-xx-xxx-xxx-xx.aws-region.compute.amazonaws.com
```

- DNS-Name Ihres Dateisystems. Sie können diesen DNS-Namen mit dem folgenden allgemeinen Format konstruieren:

```
file-system-id.efs.aws-region.amazonaws.com
```

Die EC2-Instance, in der Sie das Dateisystem mithilfe des Mounting-Ziels mounten, kann den DNS-Namen des Dateisystems zu der IP-Adresse des Mounting-Ziels auflösen.

Note

Amazon EFS setzt nicht voraus, dass Ihre EC2-Instance entweder eine öffentliche IP-Adresse oder einen öffentlichen DNS-Namen hat. Die oben aufgeführten Anforderungen gelten nur für dieses Beispiel, um sicherzustellen, dass Sie eine Verbindung mit der Instance per SSH von außerhalb der VPC herstellen können.

Schritt 3.2: Installieren Sie den NFS-Client auf Ihrer EC2-Instance

Sie können eine Verbindung mit Ihrer EC2-Instance von Windows oder von einem Computer mit Linux oder macOS X oder einer beliebigen anderen Unix-Variante aus herstellen.

So installieren Sie einen NFS-Client

1. Stellen Sie eine Verbindung zu Ihrer EC2- Instance her. Weitere Informationen finden Sie unter [Mit Ihrer EC2-Instance verbinden](#) im Amazon-EC2-Benutzerhandbuch.
2. Führen Sie die folgenden Befehle auf der EC2-Instance mithilfe der SSH-Sitzung durch:
 - a. (Optional) Rufen Sie Aktualisierungen ab, und führen Sie einen Neustart durch.

```
$ sudo yum -y update
```

```
$ sudo reboot
```

Stellen Sie nach dem Neustart erneut eine Verbindung mit Ihrer EC2-Instance her.

- b. Installieren Sie den NFS-Client.

```
$ sudo yum -y install nfs-utils
```

 Note

Wenn Sie beim Starten Ihrer EC2-Instance das Amazon Linux AMI 2016.03.0 wählen, müssen Sie es nicht installieren, `nfs-utils` da es standardmäßig bereits im AMI enthalten ist.

Schritt 3.3: Mounten Sie das Dateisystem auf Ihrer EC2-Instance und testen Sie es

Jetzt mounten Sie das Dateisystem auf Ihrer EC2-Instance.

1. Erstellen Sie ein Verzeichnis („`./`“). `efs-mount-point`

```
$ mkdir ~/efs-mount-point
```

2. Hängen Sie das EFS-Dateisystem ein.

```
$ sudo mount -t nfs -o  
nfsvers=4.1,rsiz=1048576,wsiz=1048576,hard,timeo=600,retrans=2,noresvport mount-  
target-DNS:/ ~/efs-mount-point
```

Die EC2-Instance kann den DNS-Namen des Mounting-Ziels zur IP-Adresse auflösen. Optional können Sie die IP-Adresse des Mounting-Ziels auch direkt angeben.

```
$ sudo mount -t nfs -o  
nfsvers=4.1,rsiz=1048576,wsiz=1048576,hard,timeo=600,retrans=2,noresvport mount-  
target-ip:/ ~/efs-mount-point
```

3. Nachdem Sie das EFS-Dateisystem auf Ihrer EC2-Instance bereitgestellt haben, können Sie Dateien erstellen.

- a. Ändern Sie das Verzeichnis.

```
$ cd ~/efs-mount-point
```

- b. Listen Sie die Inhalte des Verzeichnisses auf.

```
$ ls -al
```

Es sollte leer sein.

```
drwxr-xr-x 2 root      root      4096 Dec 29 22:33 .  
drwx----- 4 ec2-user ec2-user 4096 Dec 29 22:54 ..
```

- c. Der Eigentümer und Inhaber der Schreibrechte eines Dateisystems ist bei dessen Erstellung der Root-Benutzer, Sie müssen daher die Berechtigungen zum Hinzufügen von Dateien ändern.

```
$ sudo chmod go+rw .
```

Wenn Sie jetzt den Befehl `ls -al` ausprobieren, sehen Sie, dass die Berechtigungen geändert wurden.

```
drwxrwxrwx 2 root      root      4096 Dec 29 22:33 .  
drwx----- 4 ec2-user ec2-user 4096 Dec 29 22:54 ..
```

- d. Erstellen Sie eine Textdatei.

```
$ touch test-file.txt
```

- e. Listen Sie den Inhalt des Verzeichnisses auf.

```
$ ls -l
```

Sie haben jetzt erfolgreich ein EFS-Dateisystem auf Ihrer EC2-Instance in Ihrer VPC erstellt und bereitgestellt.

Das Dateisystem, das Sie gemountet haben, bleibt bei Neustarts nicht erhalten. Für ein automatisches erneutes Mounting des Verzeichnisses können Sie die Datei `fstab` verwenden. Wenn

Sie eine Auto-Scaling-Gruppe zum Starten von EC2-Instances verwenden, können Sie auch Skripts in einer Startkonfiguration einrichten.

Schritt 4: Bereinigen

Wenn Sie die erstellten Ressourcen nicht mehr benötigen, sollten Sie sie entfernen. Sie können dies mit der CLI tun.

- Entfernen Sie EC2-Ressourcen (die EC2-Instance und die beiden Sicherheitsgruppen). Amazon EFS löscht die Netzwerkschnittstelle, wenn Sie das Mounting-Ziel löschen.
- Entfernen Sie die EFS-Ressourcen (Dateisystem, Mount-Ziel).

Um AWS Ressourcen zu löschen, die in dieser exemplarischen Vorgehensweise erstellt wurden

1. Beenden Sie die EC2-Instanz, die Sie für dieses Tutorial erstellt haben.

```
$ aws ec2 terminate-instances \  
--instance-ids instance-id \  
--profile adminuser
```

Sie können EC2-Ressourcen auch über die Konsole löschen. Anweisungen finden Sie unter [Beenden einer Instance](#).

2. Löschen Sie das Mounting-Ziel.

Sie müssen die für das Dateisystem erstellten Mounting-Ziele löschen, bevor Sie das Dateisystem löschen können. Mit dem CLI-Befehl `describe-mount-targets` erhalten Sie eine Liste der Mounting-Ziele.

```
$ aws efs describe-mount-targets \  
--file-system-id file-system-ID \  
--profile adminuser \  
--region aws-region
```

Löschen Sie dann das Mounting-Ziel mit dem CLI-Befehl `delete-mount-target`.

```
$ aws efs delete-mount-target \  
--mount-target-id ID-of-mount-target-to-delete \  
--profile adminuser
```

```
--region aws-region
```

3. (Optional) Löschen Sie die zwei Sicherheitsgruppen, die Sie erstellt haben. Die Erstellung von Sicherheitsgruppen ist kostenlos.

Sie müssen zunächst die Sicherheitsgruppe des Mounting-Ziels löschen, bevor Sie die Sicherheitsgruppe der EC2-Instance löschen können. Die Sicherheitsgruppe des Mounting-Ziels hat eine Regel, die auf die EC2-Sicherheitsgruppe verweist. Daher können Sie die Sicherheitsgruppe der EC2-Instance nicht zuerst löschen.

Anweisungen finden Sie unter [Löschen Ihrer Sicherheitsgruppe](#) im Amazon EC2 EC2-Benutzerhandbuch.

4. Löschen Sie das Dateisystem mithilfe des CLI-Befehls `delete-file-system`. Mit dem CLI-Befehl `describe-file-systems` erhalten Sie eine Liste Ihrer Dateisysteme. Sie können die Dateisystem-ID aus der Antwort ableiten.

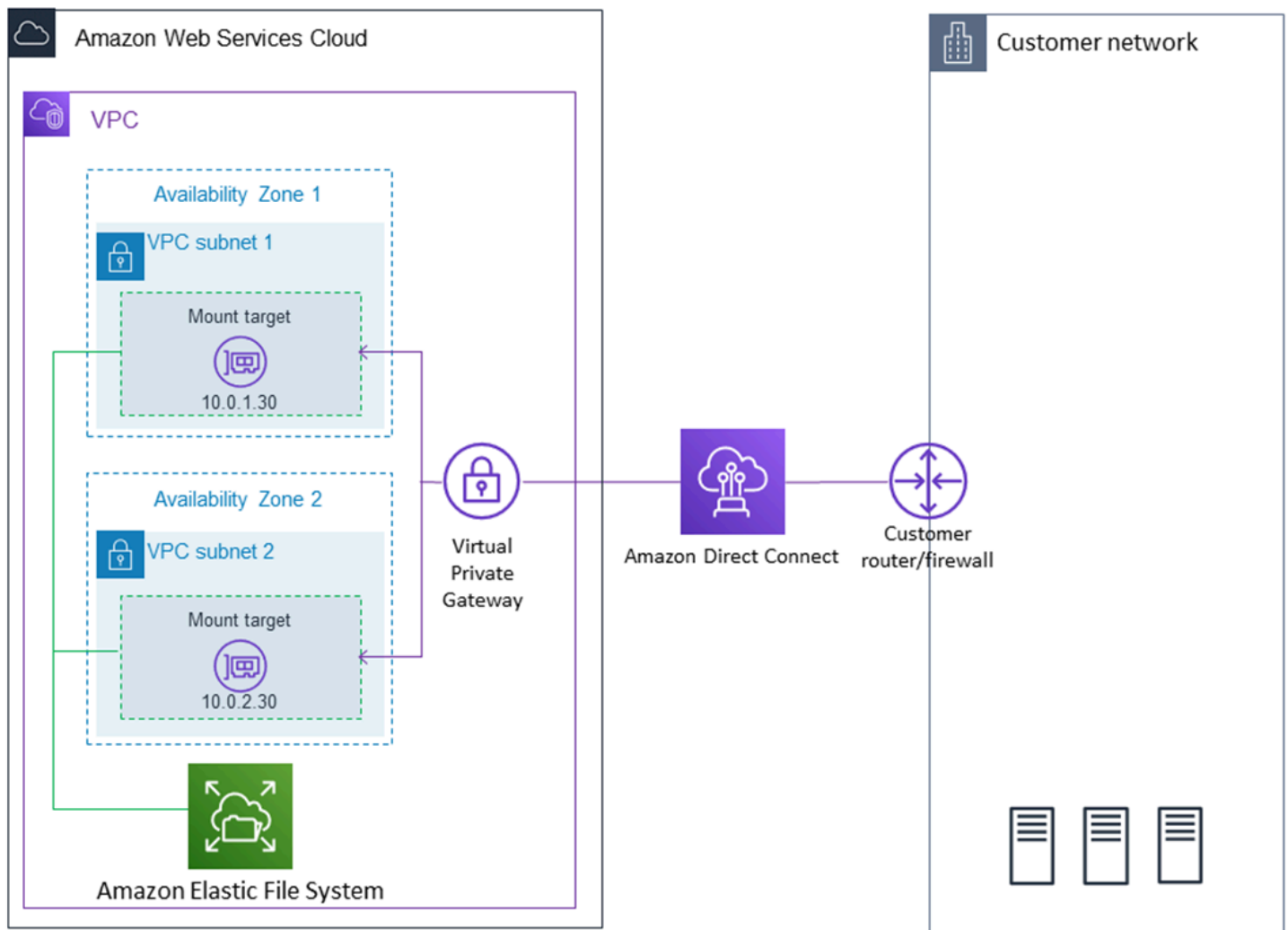
```
aws efs describe-file-systems \  
--profile adminuser \  
--region aws-region
```

Löschen Sie das Dateisystem, indem Sie die Dateisystem-ID angeben.

```
$ aws efs delete-file-system \  
--file-system-id ID-of-file-system-to-delete \  
--region aws-region \  
--profile adminuser
```

Tutorial: Mounten mit lokalen Linux-Clients

Sie können Ihre EFS-Dateisysteme auf Ihren lokalen Rechenzentrumsservern mounten, wenn Sie mit Direct Connect oder VPN mit Ihrer Amazon VPC verbunden sind. Die folgende Grafik zeigt eine allgemeine schematische Darstellung der Anforderungen beim AWS-Services Mounten von EFS-Dateisystemen von vor Ort aus.



Note

Die Verwendung von Amazon EFS mit Microsoft Windows-basierten Clients wird nicht unterstützt.

Themen

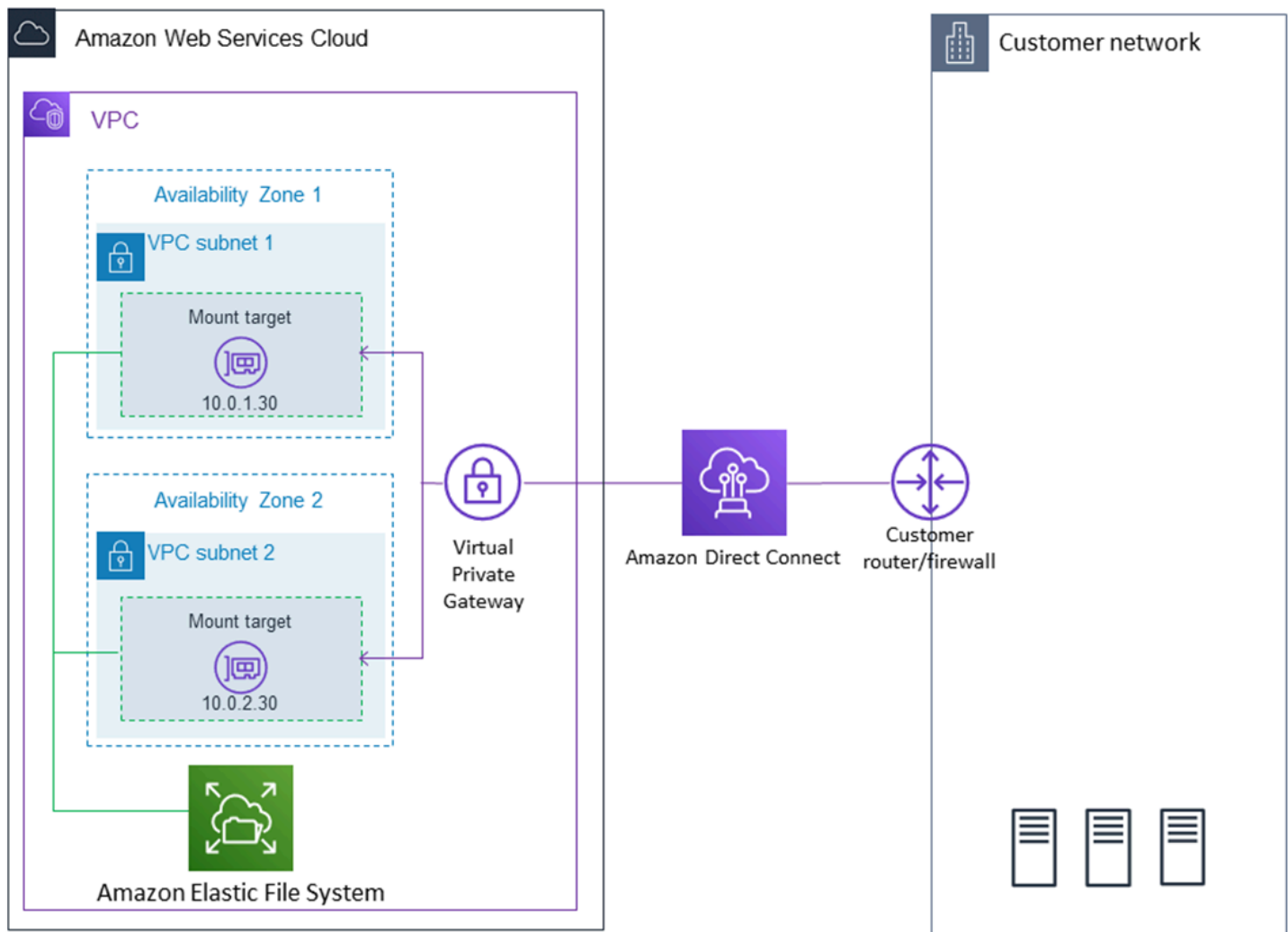
- [Voraussetzungen](#)
- [Schritt 1: Erstellen Sie Ihre EFS-Ressourcen](#)
- [Schritt 2: Installieren des NFS-Clients](#)
- [Schritt 3: Mounten Sie das EFS-Dateisystem auf Ihrem lokalen Client](#)
- [Schritt 4: Bereinigen Sie Ressourcen und schützen Sie Ihr AWS Konto](#)
- [Optional: Datenverschlüsselung während der Übertragung](#)

Voraussetzungen

Stellen Sie sicher, dass Sie bereits über eine Direct Connect oder VPN-Verbindung verfügen. Weitere Informationen zu Direct Connect finden Sie im [Direct Connect -Benutzerhandbuch](#). Weitere Informationen zum Einrichten einer VPN-Verbindung finden Sie unter [Connect Ihrer VPC mit Remote-Netzwerken AWS Virtual Private Network](#) im Amazon VPC-Benutzerhandbuch.

Nachdem Sie eine Direct Connect oder VPN-Verbindung hergestellt haben, erstellen Sie ein EFS-Dateisystem und ein Mount-Ziel in Ihrer Amazon VPC. Danach laden Sie die amazon-efs-utils Tools herunter und installieren sie. Sie testen das Dateisystem von Ihrem On-Premises-Client aus. Der Bereinigungsschritt am Ende der Anleitung stellt Informationen zum Entfernen dieser Ressourcen bereit.

Die Komplettlösung erstellt alle diese Ressourcen in der Region USA West (Oregon) (us-west-2). Was auch immer AWS-Region Sie verwenden, achten Sie darauf, es konsistent zu verwenden. Alle Ihre Ressourcen — Ihre VPC, Ihr Mount-Ziel und Ihr EFS-Dateisystem — müssen sich im selben System befinden AWS-Region, wie in der folgenden Abbildung dargestellt.



Note

In einigen Fällen muss Ihre lokale Anwendung möglicherweise wissen, ob das EFS-Dateisystem verfügbar ist. In diesen Fällen sollte Ihre Anwendung in der Lage sein, auf eine andere IP-Adresse des Mounting-Punkts zu verweisen, wenn der erste Mounting-Punkt vorübergehend nicht verfügbar ist. In diesem Szenario empfehlen wir, dass Sie zwei lokale Clients verwenden, die über verschiedene Availability Zones (AZs) mit Ihrem Dateisystem verbunden sind, um eine höhere Verfügbarkeit zu gewährleisten.

Sie können Ihre Root-Anmeldeinformationen verwenden AWS-Konto, um sich an der Konsole anzumelden und diese Übung auszuprobieren. In den bewährten Methoden von AWS Identity and Access Management (IAM) wird jedoch empfohlen, nicht die Root-Anmeldeinformationen Ihres AWS-Konto zu verwenden. Erstellen Sie stattdessen einen Administrator-Benutzer in Ihrem Konto, und

verwenden Sie dessen Anmeldeinformationen für die Verwaltung von Ressourcen in Ihrem Konto. Weitere Informationen finden Sie unter [Single Sign-On-Zugriff auf AWS-Konten](#) im AWS IAM Identity Center Benutzerhandbuch.

Sie können eine Standard-VPC oder eine benutzerdefinierte VPC, die Sie in Ihrem Konto erstellt haben, verwenden. Für diese Anleitung funktioniert die Standard-VPC-Konfiguration. Wenn Sie jedoch eine benutzerdefinierte VPC verwenden, überprüfen Sie Folgendes:

- Das Internet-Gateway ist mit Ihrer VPC verbunden. Weitere Informationen finden Sie unter [Aktivieren des Internetzugangs für eine VPC mithilfe eines Internet-Gateways](#) im Amazon VPC-Benutzerhandbuch.
- Die VPC-Routing-Tabelle enthält eine Regel zum Senden des gesamten Internet-Datenverkehrs an das Internet-Gateway.

Schritt 1: Erstellen Sie Ihre EFS-Ressourcen

In diesem Schritt erstellen Sie Ihr EFS-Dateisystem und mounten Ziele.

So erstellen Sie Ihr EFS-Dateisystem

1. Öffnen Sie die Amazon Elastic File System-Konsole unter <https://console.aws.amazon.com/efs/>.
2. Klicken Sie auf Create File System (Dateisystem erstellen).
3. Wählen Sie Ihre Standard-VPC aus der VPC-Liste aus.
4. Wählen Sie die Kontrollkästchen für alle Availability Zones aus. Stellen Sie sicher, dass alle über Standard-Subnetze, automatische IP-Adressen und die gewählten Standardsicherheitsgruppen verfügen. Diese sind Ihre Mounting-Ziele. Weitere Informationen finden Sie unter [Verwalten der Mountingziele](#).
5. Wählen Sie Next Step (Weiter) aus.
6. Benennen Sie Ihr Dateisystem, lassen Sie allgemeiner Zweck als Standardleistungsmodus ausgewählt, und klicken Sie auf Nächster Schritt.
7. Klicken Sie auf Create File System (Dateisystem erstellen).
8. Wählen Sie Ihr Dateisystem aus der Liste aus und notieren Sie sich den Wert der Sicherheitsgruppe. Sie benötigen diesen Wert im nächsten Schritt.

Das Dateisystem, das Sie gerade erstellt haben, verfügt über Mounting-Ziele. Jedem Mounting-Ziel ist eine Sicherheitsgruppe zugeordnet. Die Sicherheitsgruppe fungiert als virtuelle Firewall

zur Steuerung des Netzwerkverkehrs. Wenn Sie bei der Erstellung eines Mounting-Ziels keine Sicherheitsgruppe angegeben haben, verknüpft Amazon EFS die Standardsicherheitsgruppe der VPC mit dem Ziel. Wenn Sie die vorherigen Schritte genau befolgt haben, verwenden die Mounting-Ziele die Standardsicherheitsgruppe.

Anschließend fügen Sie der Sicherheitsgruppe des Mounting-Ziels eine Regel hinzu, die eingehenden Datenverkehr zum Network File System (NFS)-Port (2049) zulässt. Sie können die Regel verwenden AWS-Managementkonsole , um die Regel zu den Sicherheitsgruppen Ihres Mount-Ziels in Ihrer VPC hinzuzufügen.

So ermöglichen Sie eingehenden Datenverkehr zum NFS-Port:

1. Melden Sie sich bei der an AWS-Managementkonsole und öffnen Sie die Amazon EC2 EC2-Konsole unter <https://console.aws.amazon.com/ec2/>.
2. Wählen Sie unter NETZWERK UND SICHERHEIT die Option Sicherheitsgruppen aus.
3. Wählen Sie die Sicherheitsgruppe für Ihr Dateisystem. Sie haben sich diese am Ende von [Schritt 1: Erstellen Sie Ihre EFS-Ressourcen](#) notiert.
4. Wählen Sie im Registerkartenbereich unter der Liste der Sicherheitsgruppen die Registerkarte Eingehend aus.
5. Wählen Sie Bearbeiten aus.
6. Klicken Sie auf Regel hinzufügen und wählen Sie eine Regel des folgenden Typs aus:
 - Typ – NFS
 - Quelle – Beliebig

Wir empfehlen, dass Sie für Tests nur die Quelle Beliebig verwenden. Sie können eine benutzerdefinierte Quelle erstellen, die auf die IP-Adresse des Clients vor Ort festgelegt ist. Oder Sie verwenden die Konsole vom Client selbst und wählen Meine IP aus.

Note

Sie müssen keine ausgehende Regel hinzufügen, da die Standardausgangsregel jeden Datenverkehr nach außen zulässt. Wenn Sie nicht über diese Standardregel für ausgehenden Datenverkehr verfügen, fügen Sie eine Regel für ausgehenden Datenverkehr hinzu, um eine TCP-Verbindung auf dem NFS-Port 2049 zu öffnen und dabei die Mount-Zielsicherheitsgruppe als Ziel zu identifizieren.

Schritt 2: Installieren des NFS-Clients

In diesem Schritt installieren Sie den NFS-Client.

So installieren Sie den NFS-Client auf Ihrem On-Premises-Server

Note

Wenn Sie eine Verschlüsselung der Daten während der Übertragung benötigen, verwenden Sie die Amazon-EFS-Mountinghilfe, `amazon-efs-utils`, anstelle des NFS-Clients. Informationen zur Installation `amazon-efs-utils` finden Sie im Abschnitt **Optional: Verschlüsselung von Daten bei der Übertragung**.

1. Öffnen Sie das Terminal für den Client vor Ort.
2. Installieren Sie NFS.

Wenn Sie Red Hat Linux verwenden, installieren Sie NFS mit dem folgenden Befehl.

```
$ sudo yum -y install nfs-utils
```

Wenn Sie Ubuntu verwenden, installieren Sie NFS mit dem folgenden Befehl.

```
$ sudo apt-get -y install nfs-common
```

Schritt 3: Mounten Sie das EFS-Dateisystem auf Ihrem lokalen Client

So erstellen Sie ein Mount-Verzeichnis

1. Erstellen Sie ein Verzeichnis für den Mountingpunkt mit dem folgenden Befehl.

Example

```
mkdir ~/efs
```

2. Wählen Sie die bevorzugte IP-Adresse des Mounting-Ziels in der Availability Zone aus. Sie können die Latenz auf den Linux-Clients vor Ort messen. Geben Sie dafür die jeweilige IP-

Adresse der EC2-Instances in verschiedenen Availability Zones in ein terminalbasiertes Tool wie `ping` ein, um die Instance mit der niedrigsten Latenz zu bestimmen.

- Führen Sie den `mount`-Befehl aus, um das Dateisystem mit der IP-Adresse des Mounting-Ziels zu mounten.

```
$ sudo mount -t nfs -o  
nfsvers=4.1,rsize=1048576,wsiz=1048576,hard,timeo=600,retrans=2,noresvport mount-  
target-IP:/ ~/efs
```

Nachdem Sie Ihr EFS-Dateisystem bereitgestellt haben, können Sie es mit dem folgenden Verfahren testen.

So testen Sie die EFS-Dateisystemverbindung

1. Wechseln Sie mit dem folgenden Befehl zum neuen Verzeichnis, das Sie erstellt haben.

```
$ cd ~/efs
```

2. Erstellen Sie ein Unterverzeichnis, und ändern Sie dessen Eigentümerschaft zu Ihrem EC2-Instance-Benutzer. Navigieren Sie dann mit den folgenden Befehlen zu diesem neuen Verzeichnis.

```
$ sudo mkdir getting-started  
$ sudo chown ec2-user getting-started  
$ cd getting-started
```

3. Erstellen Sie eine Textdatei mit dem folgenden Befehl.

```
$ touch test-file.txt
```

4. Listen Sie mit dem folgenden Befehl den Inhalt des Verzeichnisses auf.

```
$ ls -al
```

Als Ergebnis wird die folgende Datei erstellt.

```
-rw-rw-r-- 1 username username 0 Nov 15 15:32 test-file.txt
```

Warning

Verwenden Sie beim automatischen Mounting Ihres Dateisystems die Option `_netdev`, um es als Netzwerkdateisystem zu identifizieren. Wenn `_netdev` fehlt, reagiert die EC2-Instance möglicherweise nicht mehr. Der Grund dafür ist, dass zuerst das Netzwerk auf der Datenverarbeitungs-Instance gestartet worden sein muss. Die Netzwerkdateisysteme müssen danach initialisiert werden. Weitere Informationen finden Sie unter [Automatisches Mounting schlägt fehl und die Instance reagiert nicht](#).

Schritt 4: Bereinigen Sie Ressourcen und schützen Sie Ihr AWS Konto

Nachdem Sie diese Komplettlösung beendet haben, oder wenn Sie die Komplettlösungen nicht erforschen möchten, sollten Sie diese Schritte befolgen, um Ihre Ressourcen zu bereinigen und Ihr AWS -Konto zu schützen.

Um Ressourcen zu bereinigen und Ihr AWS-Konto

1. Heben Sie das Mounting des EFS-Dateisystems mit dem folgenden Befehl auf.

```
$ sudo umount ~/efs
```

2. Öffnen Sie die Amazon EFS-Konsole unter [Amazon EFS Console](#).
3. Wählen Sie das EFS-Dateisystem, das Sie löschen möchten, aus der Liste der Dateisysteme aus.
4. Klicken Sie bei Aktionen auf Dateisystem löschen.
5. Geben Sie im Dialogfeld Dateisystem dauerhaft löschen die Dateisystem-ID für das EFS-Dateisystem ein, das Sie löschen möchten, und klicken Sie auf Dateisystem löschen.
6. Öffnen Sie die Amazon-EC2-Konsole unter <https://console.aws.amazon.com/ec2/>.
7. Wählen Sie im Navigationsbereich Security Groups (Sicherheitsgruppen) aus.
8. Wählen Sie den Namen der Sicherheitsgruppe, der Sie in dieser Übung die Regel hinzugefügt haben.

 Warning

Löschen Sie nicht die Standardsicherheitsgruppe für Ihre VPC.

9. Wählen Sie unter Actions (Aktionen) die Option Edit inbound rules (Eingangsregeln bearbeiten) aus.
10. Klicken Sie auf das X am Ende der von Ihnen hinzugefügten Eingangsregel und wählen Sie Save (Speichern) aus.

Optional: Datenverschlüsselung während der Übertragung

Um Daten während der Übertragung zu verschlüsseln, verwenden Sie den Amazon EFS-Mount-Helper anstelle des NFS-Clients. `amazon-efs-utils`

Das Paket „amazon-efs-utils“ ist eine Open-Source-Sammlung von Amazon-EFS-Tools. Die `amazon-efs-utils` Sammlung enthält einen Mount-Helfer und Tools, die es einfacher machen, Daten während der Übertragung für Amazon EFS zu verschlüsseln. Weitere Informationen zu diesem Paket finden Sie unter [Den Amazon EFS-Client installieren](#). Dieses Paket ist als kostenloser Download erhältlich. Sie können es herunterladen GitHub, indem Sie das Projektarchiv des Pakets klonen.

Zum Klonen `amazon-efs-utils` von GitHub

1. Öffnen Sie das Terminal für den Client vor Ort.
2. Klonen Sie das `amazon-efs-utils` Tool vom GitHub Terminal aus mit dem folgenden Befehl in ein Verzeichnis Ihrer Wahl.

```
git clone https://github.com/aws/efs-utils
```

Nachdem Sie das Paket jetzt erhalten haben, können Sie es installieren. Diese Installation erfolgt unterschiedlich, abhängig von der Linux-Distribution auf dem On-Premises-Client. Folgende Distributionen werden unterstützt:

- Amazon Linux 2
- Amazon Linux
- Red Hat Enterprise Linux (und Derivate wie CentOS) Version 8 und neuer
- Ubuntu 16.04 LTS und höher

Um es amazon-efs-utils als RPM-Paket zu erstellen und zu installieren

1. Öffnen Sie ein Terminal auf Ihrem Client und navigieren Sie zu dem Verzeichnis, aus dem das geklonte amazon-efs-utils Paket stammt GitHub.
2. Erstellen Sie das Paket mit dem folgenden Befehl.

```
make rpm
```

Note

Falls noch nicht erfolgt, installieren Sie das Paket „rpm-builder“ mit dem folgenden Befehl.

```
sudo yum -y install rpm-build
```

3. Installieren Sie das Paket mit dem folgenden Befehl:

```
sudo yum -y install build/amazon-efs-utils*rpm
```

Um es amazon-efs-utils als Deb-Paket zu bauen und zu installieren

1. Öffnen Sie ein Terminal auf Ihrem Client und navigieren Sie zu dem Verzeichnis, aus dem das geklonte amazon-efs-utils Paket stammt GitHub.
2. Erstellen Sie das Paket mit dem folgenden Befehl.

```
./build-deb.sh
```

3. Installieren Sie das Paket mit dem folgenden Befehl:

```
sudo apt-get install build/amazon-efs-utils*deb
```

Nachdem das Paket installiert ist, konfigurieren Sie es amazon-efs-utils für die Verwendung in Ihrem AWS-Region WLAN Direct Connect oder VPN.

Zur Konfiguration amazon-efs-utils für die Verwendung in Ihrem AWS-Region

1. Greifen Sie über Secure Shell (SSH) auf das Terminal für die EC2-Instance zu und melden Sie sich mit dem entsprechenden Benutzernamen an. Weitere Informationen finden Sie unter [Mit Ihrer EC2-Instance verbinden](#) im Amazon-EC2-Benutzerhandbuch.
2. Öffnen Sie mit einem Texteditor Ihrer Wahl die Datei `/etc/amazon/efs/efs-utils.conf`.
3. Suchen Sie die Zeile `dns_name_format = {fs_id}.efs.{region}.amazonaws.com`.
4. Ändern Sie `{region}` durch die ID für Ihre AWS Region, zum Beispiel `us-west-2`.

Zum Mounten des EFS-Dateisystems auf dem Client vor Ort öffnen Sie zuerst ein Terminal auf Ihrem Linux-Client vor Ort. Um das System einzuhängen, benötigen Sie die Dateisystem-ID, die IP-Adresse des Mounting-Ziels für eines Ihrer Mounting-Ziele und das AWS-Region des Dateisystems. Wenn Sie mehrere Mounting-Ziele für Ihr Dateisystem erstellt haben, können Sie eines davon auswählen.

Wenn Sie über diese Informationen verfügen, können Sie das Dateisystem in drei Schritten mounten:

So erstellen Sie ein Mount-Verzeichnis

1. Erstellen Sie ein Verzeichnis für den Mountingpunkt mit dem folgenden Befehl.

Example

```
mkdir ~/efs
```

2. Wählen Sie die bevorzugte IP-Adresse des Mounting-Ziels in der Availability Zone aus. Sie können die Latenz auf den Linux-Clients vor Ort messen. Geben Sie dafür die jeweilige IP-Adresse der EC2-Instances in verschiedenen Availability Zones in ein terminalbasiertes Tool wie `ping` ein, um die Instance mit der niedrigsten Latenz zu bestimmen.

So aktualisieren Sie **/etc/hosts**

- Fügen Sie der lokalen Datei `/etc/hosts` einen Eintrag mit der Dateisystem-ID und der Mounting-Ziel-IP-Adresse im folgenden Format hinzu.

```
mount-target-IP-Address file-system-ID.efs.region.amazonaws.com
```

Example

```
192.0.2.0 fs-12345678.efs.us-west-2.amazonaws.com
```

So erstellen Sie ein Mount-Verzeichnis

1. Erstellen Sie ein Verzeichnis für den Mountingpunkt mit dem folgenden Befehl.

Example

```
mkdir ~/efs
```

2. Führen Sie den Mounting-Befehl zum Mounten des Dateisystems aus.

Example

```
sudo mount -t efs fs-12345678 ~/efs
```

Wenn Sie die Verschlüsselung von Daten bei der Übertragung verwenden möchten, sieht der Mounting-Befehl in etwa folgendermaßen aus.

Example

```
sudo mount -t efs -o tls fs-12345678 ~/efs
```

Tutorial: Mounten Sie ein Dateisystem von einer anderen VPC

In diesem Tutorial richten Sie eine EC2-Instanz ein, um ein EFS-Dateisystem zu mounten, das sich in einer anderen Virtual Private Cloud (VPC) befindet. Dies ist mit der EFS-Mountinghilfe möglich. Die Mountinghilfe ist Teil der `amazon-efs-utils`-Tools. Mehr über `amazon-efs-utils` erfahren Sie unter [Den Amazon EFS-Client installieren](#).

Die VPC des Clients und die VPC des EFS-Dateisystems müssen entweder über eine VPC-Peering-Verbindung oder ein VPC-Transit-Gateway verbunden sein. Wenn Sie eine VPC-Peering-Verbindung oder ein Transit-Gateway verwenden, um VPCs zu verbinden, können EC2-Instances, die sich in einer VPC befinden, auf EFS-Dateisysteme in einer anderen VPC zugreifen, auch wenn sie zu unterschiedlichen Konten gehören. VPCs

 Note

Die Verwendung von Amazon EFS mit Microsoft Windows-basierten Clients wird nicht unterstützt.

Themen

- [Voraussetzungen](#)
- [Schritt 1: Ermitteln Sie die ID der Availability Zone des Mount-Ziels](#)
- [Schritt 2: Ermitteln Sie die Mount-Ziel-IP-Adresse](#)
- [Schritt 3: Fügen Sie einen Hosteintrag für das Mount-Ziel hinzu](#)
- [Schritt 4: Mounten Sie Ihr Dateisystem mit dem EFS-Mount-Helper](#)
- [Schritt 5: Säubern Sie Ressourcen und schützen Sie Ihre AWS-Konto](#)

Voraussetzungen

Um dieses Tutorial abschließen zu können, benötigen Sie folgende Voraussetzungen:

- Die `amazon-efs-utils`-Tools werden auf der EC2-Instance installiert, bevor Sie dieses Verfahren verwenden. Anweisungen zur Installation von `amazon-efs-utils` finden Sie unter [Den Amazon EFS-Client installieren](#).
- Eine der beiden folgenden Komponenten:
 - Eine VPC-Peering-Verbindung zwischen der VPC, in der sich das EFS-Dateisystem befindet, und der VPC, in der sich die EC2-Instance befindet. Eine VPC-Peering-Verbindung ist eine Netzwerkverbindung zwischen zwei VPCs. Dieser Verbindungstyp ermöglicht es Ihnen, den Verkehr zwischen ihnen mithilfe von privaten Internetprotokolladressen der Version 4 (IPv4) oder der Internetprotokoll-Version 6 (IPv6) weiterzuleiten. Sie können VPC-Peering verwenden, um VPCs innerhalb derselben AWS-Region oder zwischen ihnen eine Verbindung herzustellen. AWS-Regionen Weitere Informationen finden Sie unter [Erstellen und Akzeptieren einer VPC-Peering-Verbindung](#) in der Amazon VPC Peering-Anleitung.
 - Ein Transit Gateway verbindet die VPC mit dem EFS-Dateisystem und die VPC mit der EC2-Instance. Ein Transit-Gateway ist ein Netzwerk-Transit-Hub, über den Sie Ihre Netzwerke VPCs und Ihre lokalen Netzwerke miteinander verbinden können. Weitere Informationen finden Sie unter [Erste Schritte mit der Nutzung von Amazon VPC Transit Gateways](#) im Amazon VPC Transit Gateways-Handbuch.

Schritt 1: Ermitteln Sie die ID der Availability Zone des Mount-Ziels

Um eine hohe Verfügbarkeit Ihres Dateisystems zu gewährleisten, empfehlen wir, dass Sie immer eine EC2-Mount-Ziel-IP-Adresse verwenden, die sich in derselben Availability Zone wie Ihr NFS-Client befindet. Wenn Sie ein EFS-Dateisystem einhängen, das sich in einem anderen Konto befindet, stellen Sie sicher, dass sich der NFS-Client und das EFS-Mounting-Ziel in derselben Availability Zone ID befinden. Diese Anforderung gilt, weil die Namen der Availability Zones von Konto zu Konto unterschiedlich sein können.

So ermitteln Sie die Availability Zone ID der EC2-Instance

1. Stellen Sie eine Verbindung zu Ihrer EC2-Instance her. Weitere Informationen finden Sie unter [Mit Ihrer EC2-Instance verbinden](#) im Amazon-EC2-Benutzerhandbuch.
2. Ermitteln Sie die Availability Zone ID, in der sich die EC2-Instance befindet, mit dem CLI-Befehl `describe-availability-zones` wie folgt.

```
[ec2-user@ip-10.0.0.1] $ aws ec2 describe-availability-zones --zone-name
{
  "AvailabilityZones": [
    {
      "State": "available",
      "ZoneName": "us-east-2b",
      "Messages": [],
      "ZoneId": "use2-az2",
      "RegionName": "us-east-2"
    }
  ]
}
```

Die ID der Availability Zone wird in der Eigenschaft `use2-az2`, `ZoneId` zurückgegeben.

Schritt 2: Ermitteln Sie die Mount-Ziel-IP-Adresse

Da Sie nun die Availability Zone ID der EC2-Instance kennen, können Sie nun die IP-Adresse des Mounting-Ziels abrufen, das sich in derselben Availability Zone ID befindet.

So ermitteln Sie die IP-Adresse des Mounting-Ziels in derselben Availability Zone ID

- Rufen Sie die IP-Adresse des Mounting-Ziels für Ihr Dateisystem in der use2-az2-AZ-ID wie folgt mithilfe des `describe-mount-targets`-CLI-Befehls ab.

```
$ aws efs describe-mount-targets --file-system-id file_system_id
{
  "MountTargets": [
    {
      "OwnerId": "111122223333",
      "MountTargetId": "fsmt-11223344",
      "AvailabilityZoneId": "use2-az2",
      "NetworkInterfaceId": "eni-048c09a306023eeec",
      "AvailabilityZoneName": "us-east-2b",
      "FileSystemId": "fs-01234567",
      "LifeCycleState": "available",
      "SubnetId": "subnet-06eb0da37ee82a64f",
      "OwnerId": "958322738406",
      "IpAddress": "10.0.2.153"
    },
    ...
    {
      "OwnerId": "111122223333",
      "MountTargetId": "fsmt-667788aa",
      "AvailabilityZoneId": "use2-az3",
      "NetworkInterfaceId": "eni-0edb579d21ed39261",
      "AvailabilityZoneName": "us-east-2c",
      "FileSystemId": "fs-01234567",
      "LifeCycleState": "available",
      "SubnetId": "subnet-0ee85556822c441af",
      "OwnerId": "958322738406",
      "IpAddress": "10.0.3.107"
    }
  ]
}
```

Das Einhängeziel in der use2-az2 Availability Zone ID hat eine IP-Adresse von 10.0.2.153.

Schritt 3: Fügen Sie einen Hosteintrag für das Mount-Ziel hinzu

Sie können nun einen Eintrag in der `/etc/hosts`-Datei auf der EC2-Instance vornehmen, der die IP-Adresse des Mounting-Ziels dem Hostnamen Ihres EFS-Dateisystems zuordnet.

So fügen Sie einen Hosteintrag für das Mounting-Ziel hinzu

1. Fügen Sie der `/etc/hosts`-Datei der EC2-Instance eine Zeile für die IP-Adresse des Mounting-Ziels hinzu. Der Eintrag verwendet das Format `mount-target-IP-Address file-system-ID.efs.region.amazonaws.com`. Verwenden Sie den folgenden Befehl, um die Zeile der Datei hinzuzufügen.

```
echo "10.0.2.153 fs-01234567.efs.us-east-2.amazonaws.com" | sudo tee -a /etc/hosts
```

2. Stellen Sie sicher, dass die VPC-Sicherheitsgruppen für die EC2-Instance und das Mount-Ziel über Regeln verfügen, die den Zugriff auf das EFS-Dateisystem nach Bedarf ermöglichen. Weitere Informationen finden Sie unter [Verwenden von VPC-Sicherheitsgruppen](#).

Schritt 4: Mounten Sie Ihr Dateisystem mit dem EFS-Mount-Helper

Um Ihr EFS-Dateisystem zu mounten, erstellen Sie zunächst ein Mounting-Verzeichnis auf der EC2-Instance. Anschließend können Sie mit dem EFS-Mount-Helper das Dateisystem entweder mit AWS Identity and Access Management (IAM-) Autorisierung oder mit einem EFS-Zugriffspunkt mounten. Weitere Informationen erhalten Sie unter [Verwendung von IAM zur Steuerung des Zugriffs auf Dateisysteme](#) und [Arbeiten mit Zugriffspunkten](#).

So erstellen Sie ein Mount-Verzeichnis

- Erstellen Sie mit dem folgenden Befehl ein Verzeichnis zum Mounten des Dateisystems.

```
$ sudo mkdir /mnt/efs/
```

So mounten Sie das Dateisystem mithilfe der IAM-Autorisierung

- Verwenden Sie den folgenden Befehl, um das Dateisystem mit IAM-Autorisierung zu mounten.

```
$ sudo mount -t efs -o tls,iam file-system-id /mnt/efs/
```

So mounten Sie das Dateisystem mithilfe eines EFS-Zugangspunkts

- Verwenden Sie den folgenden Befehl, um das Dateisystem mithilfe eines EFS-Zugangspunkts zu mounten.

```
$ sudo mount -t efs -o tls,accesspoint=access-point-id file-system-id /mnt/efs/
```

Nachdem Sie Ihr EFS-Dateisystem bereitgestellt haben, können Sie es mit dem folgenden Verfahren testen.

So testen Sie die EFS-Dateisystemverbindung

1. Wechseln Sie mit dem folgenden Befehl zum neuen Verzeichnis, das Sie erstellt haben.

```
$ cd ~/mnt/efs
```

2. Erstellen Sie ein Unterverzeichnis, und ändern Sie dessen Eigentümerschaft zu Ihrem EC2-Instance-Benutzer. Navigieren Sie dann mit den folgenden Befehlen zu diesem neuen Verzeichnis.

```
$ sudo mkdir getting-started  
$ sudo chown ec2-user getting-started  
$ cd getting-started
```

3. Erstellen Sie eine Textdatei mit dem folgenden Befehl.

```
$ touch test-file.txt
```

4. Listen Sie mit dem folgenden Befehl den Inhalt des Verzeichnisses auf.

```
$ ls -al
```

Als Ergebnis wird die folgende Datei erstellt.

```
-rw-rw-r-- 1 username username 0 Nov 15 15:32 test-file.txt
```


Sie können das Dateisystem auch automatisch mounten, indem Sie der Datei `/etc/fstab` einen Eintrag hinzufügen. Weitere Informationen finden Sie unter [Aktivierung der automatischen Installation auf vorhandenen EC2-Linux-Instances](#).

 Warning

Verwenden Sie beim automatischen Mounting Ihres Dateisystems die Option `_netdev`, um es als Netzwerkdateisystem zu identifizieren. Wenn `_netdev` fehlt, reagiert die EC2-Instance möglicherweise nicht mehr. Der Grund dafür ist, dass zuerst das Netzwerk auf der Datenverarbeitungs-Instance gestartet worden sein muss. Die Netzwerkdateisysteme müssen danach initialisiert werden. Weitere Informationen finden Sie unter [Automatisches Mounting schlägt fehl und die Instance reagiert nicht](#).

Schritt 5: Säubern Sie Ressourcen und schützen Sie Ihre AWS-Konto

Nachdem Sie dieses Tutorial abgeschlossen haben, führen Sie die folgenden Schritte aus, um Ihre Ressourcen zu bereinigen und Ihre zu schützen AWS-Konto.

Um Ressourcen zu bereinigen und Ihre zu schützen AWS-Konto

1. Heben Sie das Mounting des EFS-Dateisystems mit dem folgenden Befehl auf.

```
$ sudo umount ~/efs
```

2. Öffnen Sie die Amazon Elastic File System-Konsole unter <https://console.aws.amazon.com/efs/>.
3. Wählen Sie das EFS-Dateisystem, das Sie löschen möchten, aus der Liste der Dateisysteme aus.
4. Klicken Sie bei Aktionen auf Dateisystem löschen.
5. Geben Sie im Dialogfeld Dateisystem dauerhaft löschen die Dateisystem-ID für das EFS-Dateisystem ein, das Sie löschen möchten, und klicken Sie auf Dateisystem löschen.
6. Öffnen Sie die Amazon-EC2-Konsole unter <https://console.aws.amazon.com/ec2/>.
7. Wählen Sie im Navigationsbereich Security Groups (Sicherheitsgruppen) aus.
8. Wählen Sie den Namen der Sicherheitsgruppe aus, zu der Sie die Regel für dieses Tutorial hinzugefügt haben.

 Warning

Löschen Sie nicht die Standardsicherheitsgruppe für Ihre VPC.

9. Wählen Sie unter Actions (Aktionen) die Option Edit inbound rules (Eingangsregeln bearbeiten) aus.
10. Klicken Sie auf das X am Ende der von Ihnen hinzugefügten Eingangsregel und wählen Sie Save (Speichern) aus.

Beheben von Mountingproblemen

Im Folgenden finden Sie Informationen zur Behebung von Problemen beim Mounten des EFS-Dateisystems.

Das Dateisystem-Mounting auf der Windows Instance schlägt fehl

Ein Dateisystem-Mount auf einer Amazon EC2-Instance unter Microsoft Windows schlägt fehl.

Maßnahme

Verwenden Sie Amazon EFS nicht mit Windows EC2-Instance, da es nicht unterstützt wird.

Zugriff vom Server verweigert

Ein Dateisystem-Mount schlägt mit der folgenden Meldung fehl:

```
/efs mount.nfs4: access denied by server while mounting 127.0.0.1:/
```

Dieses Problem kann auftreten, wenn Ihr NFS-Client nicht über die Berechtigung zum Mounting des Dateisystems verfügt.

Maßnahme

Wenn Sie versuchen, das Dateisystem mithilfe von IAM zu mounten, stellen Sie sicher, dass Sie in Ihrem Mount-Befehl die Option `-o iam` oder `-o tls` verwenden. Dies weist die EFS-Mountinghilfe an, Ihre Anmeldeinformationen an das EFS-Mount-Ziel zu übergeben. Wenn Sie weiterhin keinen Zugriff haben, überprüfen Sie Ihre Dateisystemrichtlinie und Ihre Identitätsrichtlinie, um sicherzustellen,

dass keine DENY-Klauseln für Ihre Verbindung vorhanden sind, und dass mindestens eine ALLOW-Klausel für die Verbindung vorhanden ist. Weitere Informationen erhalten Sie unter [Verwendung von IAM zur Steuerung des Zugriffs auf Dateisysteme](#) und [Erstellen von Dateisystemrichtlinien](#).

Automatisches Mounting schlägt fehl und die Instance reagiert nicht

Dieses Problem kann auftreten, wenn das Dateisystem automatisch auf einer Instance gemountet wurde und die Option `_netdev` nicht deklariert wurde. Wenn `_netdev` fehlt, reagiert die EC2-Instance möglicherweise nicht mehr. Der Grund dafür ist, dass zuerst das Netzwerk auf der Datenverarbeitungs-Instance gestartet worden sein muss. Die Netzwerkdateisysteme müssen danach initialisiert werden.

Maßnahme

Wenn dieses Problem auftritt, wenden Sie sich an den AWS Support.

Mounting mehrerer Amazon-EFS-Dateisysteme in `/etc/fstab` schlägt fehl

Bei Instances, die das `systemd`-Init-System mit zwei oder mehr Amazon-EFS-Einträgen bei `/etc/fstab` verwenden, kann es vorkommen, dass einige oder alle dieser Einträge nicht gemountet sind. In diesem Fall zeigt die `dmesg`-Ausgabe eine oder mehrere Zeilen, die in etwa wie folgt aussehen.

```
NFS: nfs4_discover_server_trunking unhandled error -512. Exiting with error EIO
```

Maßnahme

In diesem Fall empfehlen wir, dass Sie eine neue `systemd`-Dienstdatei in `/etc/systemd/system/mount-nfs-sequentially.service` erstellen. Welchen Code Sie in die Datei aufnehmen müssen, hängt davon ab, ob Sie die Dateisysteme manuell mounten oder die Amazon-EFS-Mountinghilfe verwenden.

- Wenn Sie die Dateisysteme manuell mounten, muss der `ExecStart` Befehl auf Network File System (NFS4) zeigen. Fügen Sie den folgenden Code in die Datei ein:

```
[Unit]
Description=Workaround for mounting NFS file systems sequentially at boot time
After=remote-fs.target

[Service]
```

```
Type=oneshot
ExecStart=/bin/mount -avt nfs4
RemainAfterExit=yes
```

```
[Install]
WantedBy=multi-user.target
```

- Wenn Sie den Amazon EFS-Mount-Helper verwenden, muss der ExecStart Befehl auf EFS verweisen, anstatt Transport Layer Security (TLS) NFS4 zu verwenden. Fügen Sie den folgenden Code in die Datei ein:

```
[Unit]
Description=Workaround for mounting NFS file systems sequentially at boot time
After=remote-fs.target
```

```
[Service]
Type=oneshot
ExecStart=/bin/mount -avt efs
RemainAfterExit=yes
```

```
[Install]
WantedBy=multi-user.target
```

Nachdem Sie die Datei erstellt haben, führen Sie die folgenden beiden Befehle aus:

1. `sudo systemctl daemon-reload`
2. `sudo systemctl enable mount-nfs-sequentially.service`

Starten Sie dann Ihre Amazon EC2-Instance neu. Die Dateisysteme werden nach Bedarf gemountet, in der Regel innerhalb einer Sekunde.

Mounting-Befehl schlägt mit der Fehlermeldung „falscher fs-Typ“ fehl

Der Mountingbefehl schlägt mit der folgenden Fehlermeldung fehl.

```
mount: wrong fs type, bad option, bad superblock on 10.1.25.30:/,
missing codepage or helper program, or other error (for several filesystems
(e.g. nfs, cifs) you might need a /sbin/mount.<type> helper program)
In some cases useful info is found in syslog - try dmesg | tail or so.
```

Maßnahme

Wenn Sie diese Meldung erhalten, installieren Sie das Paket `nfs-utils` (oder `nfs-common` unter Ubuntu). Weitere Informationen finden Sie unter [Installieren des NFS-Clients](#).

Der Mounting-Befehl schlägt mit der Fehlermeldung „Inkorrekte Mounting-Option“ fehl

Der Mountingbefehl schlägt mit der folgenden Fehlermeldung fehl.

```
mount.nfs: an incorrect mount option was specified
```

Maßnahme

Diese Fehlermeldung bedeutet höchstwahrscheinlich, dass Ihre Linux-Distribution die Network File System-Versionen 4.0 und 4.1 (NFSv4) nicht unterstützt. Um dies zu prüfen, können Sie den folgenden Befehl ausführen.

```
$ grep CONFIG_NFS_V4_1 /boot/config*
```

Wenn dieser Befehl `# CONFIG_NFS_V4_1 is not set` ausgibt, wird NFSv4.1 auf Ihrer Linux-Verteilung nicht unterstützt. Eine Liste der Amazon Machine Images (AMIs) für Amazon Elastic Compute Cloud (Amazon EC2), die NFSv4 .1 unterstützen, finden Sie unter [NFS-Support](#).

Mounting mit Zugangspunkt schlägt fehl

Der Mounting-Befehl schlägt fehl, wenn das Mounting über einen Zugangspunkt erfolgt, und es wird die folgende Fehlermeldung angezeigt:

```
mount.nfs4: mounting access_point failed, reason given by server: No such file or directory
```

Maßnahme

Diese Fehlermeldung zeigt an, dass der angegebene EFS-Pfad nicht existiert. Vergewissern Sie sich, dass Sie die Eigentümerschaft und die Berechtigungen für das Stammverzeichnis des Zugangspunkts angeben. Ohne diese Informationen wird EFS das Stammverzeichnis nicht erstellen. Weitere Informationen finden Sie unter [Arbeiten mit Zugriffspunkten](#).

Wenn Sie keinen Besitz und keine Berechtigungen für das Stammverzeichnis angeben und das Stammverzeichnis noch nicht existiert, erstellt EFS das Stammverzeichnis nicht. In diesem Fall schlagen Versuche, das Dateisystem mithilfe des Zugangspunkts zu mounten, fehl.

Das Mounting des Dateisystems schlägt sofort nach der Erstellung des Dateisystems fehl

Nach der Erstellung eines Mounting-Ziels kann es bis zu 90 Sekunden dauern, bis die DNS-Einträge (Domain Name Service) in einer AWS-Region vollständig verbreitet sind.

Maßnahme

Wenn Sie Dateisysteme programmgesteuert erstellen und bereitstellen, z. B. mit einer CloudFormation Vorlage, empfehlen wir Ihnen, eine Wartebedingung zu implementieren.

Das Mounting des Dateisystems hängt und schlägt dann mit einem Timeout-Fehler fehl

Der Mounting-Befehl des Dateisystems hängt eine oder zwei Minuten lang und schlägt dann mit einem Timeout-Fehler fehl. Der folgende Code zeigt ein Beispiel dafür.

```
$ sudo mount -t nfs -o  
nfsvers=4.1,rsize=1048576,wsiz=1048576,hard,timeo=600,retrans=2,noresvport mount-  
target-ip:/ mnt  
  
[2+ minute wait here]  
mount.nfs: Connection timed out  
$
```

Maßnahme

Dieser Fehler kann auftreten, weil entweder die Amazon EC2-Instance oder die Sicherheitsgruppen der Mounting-Ziele nicht richtig konfiguriert sind. Stellen Sie sicher, dass die Mount-Zielsicherheitsgruppe über eine Regel für eingehenden Datenverkehr verfügt, die NFS-Zugriff auf Port 2049 von der EC2-Sicherheitsgruppe aus ermöglicht. Weitere Informationen finden Sie unter [Verwenden von VPC-Sicherheitsgruppen](#).

Überprüfen Sie, ob die angegebene IP-Adresse des Mounting-Ziels korrekt ist. Wenn Sie die falsche IP-Adresse angegeben haben und unter dieser IP-Adresse nichts vorliegt, das das Mounting ablehnen könnte, kann dieses Problem auftreten.

Mounting eines Dateisystems mit NFS unter Verwendung eines DNS-Namens schlägt fehl

Der Versuch, ein Dateisystem mit einem NFS-Client (nicht mit dem `amazon-efs-utils`-Client) unter Verwendung des DNS-Namens des Dateisystems mounten, schlägt fehl, wie im folgenden Beispiel gezeigt:

```
$ sudo mount -t nfs -o
nfsvers=4.1,rsize=1048576,wsiz=1048576,hard,timeo=600,retrans=2,noresvport file-
system-id.efs.aws-region.amazonaws.com:/ mnt
mount.nfs: Failed to resolve server file-system-id.efs.aws-region.amazonaws.com:
Name or service not known.

$
```

Maßnahme

Prüfen Sie Ihre VPC-Konfiguration. Wenn Sie eine benutzerdefinierte VPC verwenden, müssen Sie sicherstellen, dass die DNS-Einstellungen aktiviert sind. Weitere Informationen finden Sie unter [DNS-Attribute für Ihre VPC](#) im Amazon VPC-Benutzerhandbuch. Außerdem sind die DNS-Namen von Dateisystemen und Mounting-Zielen von außerhalb der VPC, in der sie existieren, nicht auflösbar.

Bevor Sie ein Dateisystem mithilfe seines DNS-Namens im `mount` Befehl mounten können, müssen Sie wie folgt vorgehen:

- Stellen Sie sicher, dass sich ein Amazon-EFS-Mounting-Ziel in der gleichen Availability Zone wie die Amazon EC2-Instance befindet.
- Stellen Sie sicher, dass sich ein Mounting-Ziel in der gleichen VPC wie die Amazon EC2-Instance befindet. Andernfalls können Sie die DNS-Namensauflösung für EFS-Mounting-Ziele, die sich in einer anderen VPC befinden, nicht verwenden. Weitere Informationen finden Sie unter [Mounten von EFS-Dateisystemen von einer anderen AWS-Konto oder VPC](#).
- Verbinden Sie Ihre Amazon EC2-Instance innerhalb einer Amazon VPC, die so konfiguriert ist, dass sie den von Amazon bereitgestellten DNS-Server verwendet. Weitere Informationen finden Sie unter [DHCP-Optionssätze in Amazon VPC](#) im Amazon VPC-Benutzerhandbuch.
- Stellen Sie sicher, dass in der Amazon VPC der verbindenden Amazon EC2-Instance DNS-Hostnamen aktiviert sind. Weitere Informationen finden Sie unter [DNS-Attribute in Ihrer VPC](#) im Amazon VPC-Benutzerhandbuch.

Das Mounting des Dateisystems schlägt mit der Fehlermeldung „nfs reagiert nicht“.

Das Mounting eines Amazon-EFS-Dateisystems schlägt bei einem TCP-Wiederverbindungsereignis mit "nfs: server_name still not responding".

Maßnahme

Verwenden Sie die `noresvport` Mounting-Option, um sicherzustellen, dass der NFS-Client einen neuen TCP-Quellport verwendet, wenn eine Netzwerkverbindung wiederhergestellt wird. Dadurch wird die ununterbrochene Verfügbarkeit nach einem Netzwerkwiederherstellungsereignis sichergestellt.

Der Lebenszyklusstatus des Mounting-Ziels hängt fest

Der Lebenszyklusstatus des Mounting-Ziels hängt im Status `Wird erstellt` oder `Wird gelöscht` fest.

Maßnahme

Wiederholen Sie den Aufruf `CreateMountTarget` oder `DeleteMountTarget`.

Der Lebenszyklusstatus des Mount-Ziels zeigt einen Fehler

Der Lebenszyklusstatus des Mounting-Ziels wird als Fehler angezeigt.

Maßnahme

Amazon EFS kann die erforderlichen DNS-Einträge (Domain Name System) für neue Dateisystem-Mounting-Ziele nicht erstellen, wenn die Virtual Private Cloud (VPC) über widersprüchliche gehostete Zonen verfügt. Amazon EFS kann keine neuen Datensätze in einer kundeneigenen gehosteten Zone erstellen. Wenn Sie eine gehostete Zone mit einem kollidierenden `efs.<region>.amazonaws.com` DNS-Bereich verwalten müssen, erstellen Sie die gehostete Zone in einer separaten VPC. Weitere Informationen zu DNS-Überlegungen für VPC finden Sie unter [DNS-Attribute für Ihre VPC](#).

Um dieses Problem zu beheben, löschen Sie den in Konflikt stehenden `efs.<region>.amazonaws.com` Host aus der VPC und erstellen Sie das Mounting-Ziel neu. Weitere Informationen zum Löschen des Mounting-Ziels finden Sie unter [Verwalten der Mountingziele](#).

Mounting reagiert nicht

Ein Amazon-EFS-Mount scheint nicht zu reagieren. Beispielsweise bleiben Befehle wie `ls` hängen.

Maßnahme

Dieser Fehler kann auftreten, wenn eine andere Anwendung große Datenmengen in das Dateisystem schreibt. Der Zugriff auf die Dateien, die geschrieben werden, kann blockiert sein, bis der Vorgang abgeschlossen ist. Allgemein gilt, dass alle Befehle oder Anwendungen, die versuchen, auf Dateien zuzugreifen, die gerade geschrieben werden, augenscheinlich hängenbleiben. Beispielsweise bleibt der Befehl `ls` möglicherweise hängen, wenn er zu einer Datei gelangt, die gerade geschrieben wird. Der Grund dafür ist, dass einige Linux-Distributionen ein Alias des `ls`-Befehls erstellen, sodass er zusätzlich zum Auflisten der Verzeichnisinhalte Dateiattribute abrufen.

Um dieses Problem zu lösen, stellen Sie sicher, dass eine andere Anwendung Dateien zum EFS-Mounting schreibt, und dass sich diese im Status `Uninterruptible sleep (D)` befindet, wie im folgenden Beispiel:

```
$ ps aux | grep large_io.py
root 33253 0.5 0.0 126652 5020 pts/3 D+ 18:22 0:00 python large_io.py /efs/large_file
```

Nachdem Sie überprüft haben, dass dies der Fall ist, können Sie das Problem lösen, indem Sie darauf warten, dass der andere Schreibvorgang abgeschlossen wird, oder indem Sie ein Workaround implementieren. Im Beispiel von `ls` können Sie den Befehl `/bin/ls` direkt (anstelle eines Alias) verwenden. So kann der Befehl fortgesetzt werden, ohne bei der Datei, die gerade geschrieben wird, hängen zu bleiben. Allgemein gilt: Wenn die Anwendung, die die Daten schreibt, einen periodischen Datenfluss erzwingen kann, etwa mithilfe von `fsync(2)`, kann dies die Reaktionen Ihres Dateisystems für andere Anwendungen verbessern. Diese Verbesserung geht jedoch möglicherweise auf Kosten der Leistung, wenn die Anwendung Daten schreibt.

Gemounteter Client wird nicht mehr verbunden

Ein Client, der an ein Amazon-EFS-Dateisystem angeschlossen ist, kann gelegentlich aufgrund einer Vielzahl von Ursachen getrennt werden. NFS-Clients sind so konzipiert, dass sie sich im Falle einer Unterbrechung automatisch wieder verbinden, um die Auswirkungen routinemäßiger Verbindungsunterbrechungen auf die Leistung und Verfügbarkeit von Anwendungen zu minimieren. In den meisten Fällen stellen die Clients die Verbindung innerhalb von Sekunden wieder her.

Die NFS-Clientsoftware in älteren Versionen des Linux-Kernels (Version v5.4 und darunter) enthält jedoch ein Verhalten, das NFS-Klienten dazu veranlasst, nach einer Trennung der Verbindung zu versuchen, sich über denselben TCP-Quellport erneut zu verbinden. Dieses Verhalten entspricht nicht dem TCP RFC und kann diese Clients daran hindern, die Verbindung zu ihrem NFS-Server (in diesem Fall ein EFS-Dateisystem) schnell wiederherzustellen.

Um dieses Problem zu beheben, empfehlen wir Ihnen dringend, die Amazon-EFS-Mountinghilfe zu verwenden, um Ihre EFS-Dateisysteme zu mounten. Die EFS-Mountinghilfe verwendet Mount-Einstellungen, die für Amazon-EFS-Dateisysteme optimiert sind. Weitere Informationen über den EFS-Client und die Mountinghilfe finden Sie unter [Den Amazon EFS-Client installieren](#).

Wenn Sie die EFS-Mountinghilfe nicht verwenden können, empfehlen wir dringend die Verwendung der NFS-Mounting-Optionen, die `noresvport` NFS-Clients anweist, Verbindungen unter Verwendung neuer TCP-Quellports wiederherzustellen, um dieses Problem zu vermeiden. Weitere Informationen finden Sie unter [Empfohlene NFS-Mount-Einstellungen](#).

Operationen auf einem neu gemounteten Dateisystem geben den Fehler „bad file handle“ zurück

Vorgänge auf einem neu gemounteten Dateisystem generieren den Fehler `bad file handle`.

Dieser Fehler kann auftreten, wenn eine Amazon EC2-Instance mit einem Dateisystem und einem Mounting-Ziel mit einer bestimmten IP-Adresse verbunden war und dieses Dateisystem und Mounting-Ziel dann gelöscht wurde. Wenn Sie ein neues Dateisystem und Mounting-Ziel erstellen, um eine Verbindung zu dieser Amazon EC2-Instance mit derselben IP-Adresse des Mounting-Ziels herzustellen, kann dieses Problem auftreten.

Maßnahme

Sie können diesen Fehler beheben, indem Sie das Dateisystem unmounten und dann das Dateisystem auf der Amazon EC2-Instance erneut mounten. Weitere Informationen zum Unmounten Ihres Amazon-EFS-Dateisystems finden Sie unter [Aufheben des Mountings von Dateisystemen](#).

Unmounten eines Dateisystems schlägt fehl

Wenn Ihr Dateisystem ausgelastet ist, können Sie es nicht unmounten.

Maßnahme

Sie können dieses Problem auf folgende Weise beheben:

- Verwenden Sie Lazy Unmount, `umount -l` wodurch das Dateisystem bei der Ausführung von der Dateisystemhierarchie getrennt wird. Anschließend werden alle Verweise auf das Dateisystem gelöscht, sobald es nicht mehr ausgelastet ist.
- Warten Sie, bis alle Lese- und Schreibvorgänge abgeschlossen sind, und versuchen Sie dann, den Befehl `umount` erneut auszuführen.
- Erzwingen Sie ein Unmounten mit dem Befehl `umount -f`.

 Warning

Das Erzwingen des Ausbindens unterbricht alle Datenlese- oder -schreibvorgänge, die derzeit für das Dateisystem durchgeführt werden. Weitere Informationen und Anleitungen zur Verwendung dieser Option finden Sie auf der Seite [Manuelles Unmounten](#).

Übertragung von Daten in und aus Amazon EFS

Sie können AWS DataSync und verwenden AWS Transfer Family , um Daten in und aus Ihren Amazon EFS-Dateisystemen zu übertragen. AWS DataSync ist ein Online-Datenübertragungsservice, der Daten zwischen Network File System (NFS), Server Message Block (SMB) -Dateiservern, selbstverwaltetem Objektspeicher und auch zwischen Diensten kopieren kann. AWS Weitere Informationen zur Verwendung DataSync mit Amazon EFS finden Sie unter [Wird AWS DataSync zur Übertragung von Daten verwendet](#).

AWS Transfer Family ist ein vollständig verwalteter AWS Service, mit dem Sie Dateien über das Secure File Transfer Protocol (SFTP), File Transfer Protocol (FTP) und FTP over Secure Sockets Layer (FTPS) -Protokoll in und aus Amazon EFS-Dateisystemen übertragen können. Mit Transfer Family können Sie Ihren Geschäftspartnern Zugriff auf Dateien gewähren, die in Ihren Amazon-EFS-Dateisystemen für Anwendungsfälle wie Datenverteilung, Lieferkette, Inhaltsmanagement und Web-Serving-Anwendungen gespeichert sind. Weitere Informationen zur Verwendung von Transfer Family mit Amazon EFS finden Sie unter [Zum AWS Transfer Family Übertragen von Daten verwenden](#).

Topics

- [Wird AWS DataSync zur Übertragung von Daten verwendet](#)
- [Zum AWS Transfer Family Übertragen von Daten verwenden](#)

Wird AWS DataSync zur Übertragung von Daten verwendet

AWS DataSync ist ein Online-Datenübertragungsdienst, der das Verschieben und Replizieren von Daten zwischen lokalen Speichersystemen und auch zwischen Speicherdiensten vereinfacht, automatisiert und beschleunigt. AWS DataSync kann Daten zwischen Network File System (NFS), Server Message Block (SMB) -Dateiservern, selbstverwaltetem Objektspeicher AWS Snowball Edge, Amazon S3 S3-Buckets, EFS-Dateisystemen und FSx Windows-Dateisystemen kopieren.

Sie können es auch verwenden DataSync , um Dateien zwischen zwei EFS-Dateisystemen zu übertragen, einschließlich Dateisystemen in verschiedenen AWS-Region s und Dateisystemen, die verschiedenen AWS-Konto s gehören. Mithilfe von DataSync Daten zwischen EFS-Dateisystemen können Sie einmalige Datenmigrationen und regelmäßige Dateneingaben für verteilte Workloads durchführen und die Replikation zum Schutz und zur Wiederherstellung von Daten automatisieren.

Weitere Informationen finden Sie im [Erste Schritte mit Amazon EFS](#) und dem [AWS DataSync - Benutzerhandbuch](#).

Zum AWS Transfer Family Übertragen von Daten verwenden

AWS Transfer Family ist ein vollständig verwalteter AWS Service, mit dem Sie Dateien über die folgenden Protokolle in und aus Amazon EFS-Dateisystemen übertragen können:

- Secure Shell (SSH) File Transfer Protocol (SFTP) (AWS Transfer for SFTP)
- File Transfer Protocol Secure (FTPS) (AWS Transfer for FTPS)
- File Transfer Protocol (FTP) (AWS Transfer for FTP)

Mit der Transfer Family können Sie Dritten, wie z. B. Ihren Lieferanten, Partnern oder Kunden, den Zugriff auf Ihre Dateien über die unterstützten Protokolle auf sichere Weise und weltweit ermöglichen, ohne dass Sie eine Infrastruktur verwalten müssen. Außerdem können Sie jetzt von Windows-, macOS- und Linux-Umgebungen aus mit SFTP-, FTPS- und FTP-Clients problemlos auf Ihre EFS-Dateisysteme zugreifen. So können Sie den Zugriff auf Ihre Daten über NFS-Clients und Zugangspunkte hinaus auf Benutzer in verschiedenen Umgebungen ausweiten.

Die Verwendung von Transfer Family zur Übertragung von Daten in Amazon-EFS-Dateisystemen wird auf die gleiche Weise abgerechnet wie die Verwendung anderer Clients. Weitere Informationen erhalten Sie unter [Durchsatzmodi](#) und [Amazon EFS-Kontingente](#).

Weitere Informationen AWS Transfer Family dazu finden Sie im [AWS Transfer Family Benutzerhandbuch](#).

Note

Die Verwendung von Transfer Family mit Amazon AWS-Konto EFS ist standardmäßig für Systeme deaktiviert, die über Amazon EFS-Dateisysteme mit Richtlinien verfügen, die öffentlichen Zugriff ermöglichen und die vor dem 6. Januar 2021 erstellt wurden. Um die Verwendung von Transfer Family für den Zugriff auf Ihr Dateisystem zu ermöglichen, wenden Sie sich an Support.

Topics

- [Voraussetzungen für die Verwendung AWS Transfer Family mit Amazon EFS](#)
- [Konfiguration Ihres EFS-Dateisystems für AWS Transfer Family](#)

Voraussetzungen für die Verwendung AWS Transfer Family mit Amazon EFS

Um Transfer Family für den Zugriff auf Dateien in Ihrem Amazon-EFS-Dateisystem zu verwenden, muss Ihre Konfiguration die folgenden Bedingungen erfüllen:

- Der Transfer Family Server und Ihr Amazon-EFS-Dateisystem befinden sich in der selben AWS-Region.
- Der von Transfer Family verwendeten IAM-Rolle müssen Amazon EFS-Berechtigungen (wie `elasticfilesystem:ClientMount` und `elasticfilesystem:ClientWrite`) zugewiesen sein. Weitere Informationen zu den Berechtigungen und Richtlinien finden Sie unter [Erstellen einer IAM-Rolle und -Richtlinie](#) im AWS Transfer Family Benutzerhandbuch.
- (Optional) Wenn der Transfer Family Server einem anderen Konto gehört, aktivieren Sie den kontoübergreifenden Zugriff.
 - Stellen Sie sicher, dass Ihre Dateisystemrichtlinie keinen öffentlichen Zugriff zulässt. Weitere Informationen finden Sie unter [Sperren des öffentlichen Zugriffs auf EFS-Dateisysteme](#).
 - Ändern Sie die Dateisystemrichtlinie, um den kontenübergreifenden Zugriff zu aktivieren. Weitere Informationen finden Sie unter [Konfiguration des kontoübergreifenden Zugriffs für Transfer Family](#).

Konfiguration Ihres EFS-Dateisystems für AWS Transfer Family

Die Konfiguration eines Amazon-EFS-Dateisystems für die Verwendung mit Transfer Family erfordert die folgenden Schritte:

- Schritt 1. Ruft die Liste der POSIX-Benutzer IDs ab, die den Benutzern der Transfer Family zugewiesen sind.
- Schritt 2. Stellen Sie sicher, dass die Transfer Family Family-Benutzer auf die Verzeichnisse Ihres Dateisystems zugreifen können, indem Sie das POSIX verwenden, das den Transfer Family Family-Benutzern IDs zugewiesen ist.
- Schritt 3. Konfigurieren Sie IAM, um den Zugriff auf die von Transfer Family verwendete IAM-Rolle zu ermöglichen.

Datei- und Verzeichnisberechtigungen für Transfer Family-Benutzer festlegen

Stellen Sie sicher, dass die Benutzer der Transfer Family-Zugriff auf die erforderlichen Dateien und Verzeichnisse in Ihrem EFS-Dateisystem haben. Weisen Sie dem Verzeichnis mithilfe der POSIX-Liste, die den Benutzern der Transfer Family IDs zugewiesen ist, Zugriffsberechtigungen zu. In diesem Beispiel erstellt ein Benutzer ein Verzeichnis, das `transferFam` unter dem Amazon EFS-Bereitstellungspunkt benannt ist. Das Erstellen eines Verzeichnisses ist optional, je nach Anwendungsfall. Bei Bedarf können Sie den Namen und den Speicherort im EFS-Dateisystem auswählen.

So weisen Sie POSIX-Benutzern Datei- und Verzeichnisberechtigungen für Transfer Family zu

1. Stellen Sie eine Verbindung zu Ihrer Amazon-EC2-Instance her. Amazon EFS unterstützt nur das Mounten durch Linux-basierte EC2-Instances.
2. Mounten Sie Ihr EFS-Dateisystem, falls es nicht bereits auf der EC2-Instance gemountet ist. Weitere Informationen finden Sie unter [Mounting von EFS-Dateisystemen](#).
3. Das folgende Beispiel erstellt das Verzeichnis auf dem EFS-Dateisystem und ändert seine Gruppe in die POSIX-Gruppen-ID für die Benutzer von Transfer Family, die in diesem Beispiel 1101 lautet.
 - a. Führen Sie den folgenden Befehl aus, um das Verzeichnis `efs/transferFam` zu erstellen: In der Praxis können Sie einen Namen und einen Speicherort im Dateisystem Ihrer Wahl verwenden.

```
[ec2-user@ip-192-0-2-0 ~]$ ls
efs  efs-mount-point  efs-mount-point2
[ec2-user@ip-192-0-2-0 ~]$ ls efs
[ec2-user@ip-192-0-2-0 ~]$ sudo mkdir efs/transferFam
[ec2-user@ip-192-0-2-0 ~]$ ls -l efs
total 0
drwxr-xr-x 2 root root 6 Jan  6 15:58 transferFam
```

- b. Verwenden Sie den folgenden Befehl, um die Gruppe `efs/transferFam` in die POSIX-GID zu ändern, die den Benutzern der Transfer Family zugewiesen wurde.

```
[ec2-user@ip-192-0-2-0 ~]$ sudo chown :1101 efs/transferFam/
```

- c. Bestätigen Sie die Änderung.

```
[ec2-user@ip-192-0-2-0 ~]$ ls -l efs
total 0
drwxr-xr-x 2 root 1101 6 Jan  6 15:58 transferFam
```

Aktivieren Sie den Zugriff auf die von Transfer Family verwendete IAM-Rolle.

In Transfer Family erstellen Sie eine ressourcenbasierte IAM-Richtlinie und eine IAM-Rolle, die den Benutzerzugriff auf das EFS-Dateisystem definieren. Weitere Informationen finden Sie unter [Erstellen einer IAM-Rolle und IAM-Richtlinie](#) im AWS Transfer Family -Benutzerhandbuch. Sie müssen dieser Transfer Family IAM-Rolle entweder über eine IAM-Identitätsrichtlinie oder eine Dateisystemrichtlinie Zugriff auf Ihr EFS-Dateisystem gewähren.

Im Folgenden finden Sie ein Beispiel für eine Dateisystemrichtlinie, die der IAM-Rolle EFS-role-for-transfer ClientMount (Lesen) und ClientWrite Zugriff gewährt.

JSON

```
{
  "Version": "2012-10-17",
  "Id": "efs-policy-wizard-8698b356-4212-4d30-901e-ad2030b57762",
  "Statement": [
    {
      "Sid": "Grant-transfer-role-access",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/EFS-role-for-transfer"
      },
      "Action": [
        "elasticfilesystem:ClientWrite",
        "elasticfilesystem:ClientMount"
      ],
      "Resource": "arn:aws:elasticfilesystem:us-east-1:111122223333:file-
system/*"
    }
  ]
}
```


Weitere Informationen zum Erstellen einer Dateisystemrichtlinie finden Sie unter [Erstellen von Dateisystemrichtlinien](#). Weitere Informationen über die Verwendung identitätsbasierter IAM-Richtlinien zur Verwaltung des Zugriffs auf EFS-Ressourcen finden Sie unter [Identitätsbasierte Richtlinien für Amazon EFS](#).

Konfiguration des kontoübergreifenden Zugriffs für Transfer Family

Wenn der Transfer Family Family-Server, der für den Zugriff auf Ihr Dateisystem verwendet wird AWS-Konto, zu einem anderen gehört, müssen Sie diesem Konto Zugriff auf Ihr Dateisystem gewähren. Außerdem muss Ihre Dateisystemrichtlinie nicht öffentlich sein. Weitere Informationen zum Sperren des öffentlichen Zugriffs auf Ihr Dateisystem finden Sie unter [Sperren des öffentlichen Zugriffs auf EFS-Dateisysteme](#).

In der Dateisystemrichtlinie können Sie einen anderen AWS-Konto Zugriff auf Ihr Dateisystem gewähren. Verwenden Sie in der Amazon EFS-Konsole den Abschnitt Zusätzliche Berechtigungen gewähren des Dateisystemrichtlinien-Editors, um anzugeben, welche AWS-Konto und welche Ebene des Dateisystemzugriffs Sie gewähren. Weitere Informationen zum Erstellen oder Bearbeiten einer Dateisystemrichtlinie finden Sie unter [Erstellen von Dateisystemrichtlinien](#).

Sie können das Konto über die Konto-ID oder den Amazon Resource Name (ARN) des Kontos angeben. Weitere Informationen zu ARNs finden Sie unter [IAM ARNs](#) im IAM-Benutzerhandbuch.

Das folgende Beispiel ist eine nicht-öffentliche Dateisystem-Richtlinie, die kontoübergreifenden Zugriff auf das Dateisystem gewährt. Es enthält die folgenden beiden Aussagen:

1. Die erste Anweisung, `NFS-client-read-write-via-fsmt`, gewährt NFS-Clients, die über ein Mounting-Ziel auf das Dateisystem zugreifen, Lese-, Schreib- und Root-Rechte.
2. Die zweite Anweisung, `Grant-cross-account-access`, gewährt dem AWS-Konto 111122223333, dem Konto, dem der Transfer Family Family-Server gehört, der Zugriff auf dieses EFS-Dateisystem in Ihrem Konto benötigt, nur Lese- und Schreibberechtigungen.

```
{
  "Statement": [
    {
      "Sid": "NFS-client-read-write-via-fsmt",
      "Effect": "Allow",
      "Action": [
        "elasticfilesystem:ClientRootAccess",
        "elasticfilesystem:ClientWrite",
```

```

        "elasticfilesystem:ClientMount"
    ],
    "Condition": {
        "Bool": {
            "elasticfilesystem:AccessedViaMountTarget": "true"
        }
    }
},
{
    "Sid": "Grant-cross-account-access",
    "Effect": "Allow",
    "Principal": {
        "AWS": "arn:aws:iam::111122223333:root"
    },
    "Action": [
        "elasticfilesystem:ClientWrite",
        "elasticfilesystem:ClientMount"
    ]
}
]
}

```

Die folgende Dateisystemrichtlinie fügt eine Anweisung hinzu, die Zugriff auf die von Transfer Family verwendete IAM-Rolle gewährt.

```

{
    "Statement": [
        {
            "Sid": "NFS-client-read-write-via-fsmt",
            "Effect": "Allow",
            "Action": [
                "elasticfilesystem:ClientRootAccess",
                "elasticfilesystem:ClientWrite",
                "elasticfilesystem:ClientMount"
            ],
            "Condition": {
                "Bool": {
                    "elasticfilesystem:AccessedViaMountTarget": "true"
                }
            }
        },
        {
            "Sid": "Grant-cross-account-access",

```

```
    "Effect": "Allow",
    "Principal": {
      "AWS": "arn:aws:iam::111122223333:root"
    },
    "Action": [
      "elasticfilesystem:ClientWrite",
      "elasticfilesystem:ClientMount"
    ]
  },
  {
    "Sid": "Grant-transfer-role-access",
    "Effect": "Allow",
    "Principal": {
      "AWS": "arn:aws:iam::111122223333:role/EFS-role-for-transfer"
    },
    "Action": [
      "elasticfilesystem:ClientWrite",
      "elasticfilesystem:ClientMount"
    ]
  }
]
```

Verwaltung von EFS-Dateisystemen

Zu den Aufgaben der Dateisystemverwaltung gehören die Verwaltung des Netzwerkzugriffs eines Dateisystems mit Mount-Zielen, die Änderung des Durchsatzmodus, die Aktualisierung der Lebenszyklusrichtlinien und die Verwaltung der Verschlüsselung.

Sie können diese Aufgaben zur Dateisystemverwaltung mithilfe der AWS-Managementkonsole API oder programmgesteuert mit der AWS Command Line Interface (AWS CLI) oder API ausführen, wie in den folgenden Abschnitten beschrieben.

Themen

- [Grundlegendes zum Dateisystemstatus](#)
- [Verwalten der Mountingziele](#)
- [Verwalten des Dateisystemdurchsatzes](#)
- [Verwaltung des Speicherlebenszyklus](#)

Grundlegendes zum Dateisystemstatus

Sie können den Status von Amazon-EFS-Dateisystemen mit der Amazon-EFS-Konsole oder der AWS CLI anzeigen. Ein Amazon-EFS-Dateisystem kann einen der in der folgenden Tabelle beschriebenen Statuswerte haben.

Status des Dateisystems	Description
VERFÜGBAR	Das Dateisystem befindet sich in einem fehlerfreien Zustand und ist erreichbar und kann verwendet werden.
WIRD ERSTELLT	Amazon EFS ist dabei, das neue Dateisystem zu erstellen.
WIRD GELÖSCHT	Amazon EFS löscht das Dateisystem als Antwort auf eine vom Benutzer initiierte Löschanforderung. Weitere Informationen finden Sie unter Löschen von EFS-Dateisystemen .

Status des Dateisystems	Description
GELÖSCHT	Amazon EFS hat das Dateisystem als Antwort auf eine vom Benutzer initiierte Löschanforderung gelöscht. Weitere Informationen finden Sie unter Löschen von EFS-Dateisystemen .
WIRD AKTUALISIERT	Das Dateisystem wird als Reaktion auf eine vom Benutzer initiierte Aktualisierungsanfrage aktualisiert.
ERROR	Gilt für One-Zone-Dateisysteme, einschließlich Dateisysteme in einer Replikationskonfiguration. Das Dateisystem befindet sich in einem fehlerhaften Zustand und kann nicht wiederhergestellt werden. Um auf die Dateisystemdaten zuzugreifen, stellen Sie eine Sicherungskopie dieses Dateisystems in einem neuen Dateisystem wieder her. Weitere Informationen finden Sie unter Daten in Amazon EFS sichern und replizieren.

Verwalten der Mountingziele

Sie mounten Ihr Dateisystem auf Amazon EC2 oder einer anderen AWS Recheninstanz in Ihrer Virtual Private Cloud (VPC) mithilfe eines oder mehrerer Mount-Ziele, die Sie für das Dateisystem erstellen. Sie können das Mount-Ziel erstellen, wenn Sie das Dateisystem erstellen oder nachdem Sie das Dateisystem erstellt haben.

Nachdem ein Einhängeziel für ein Dateisystem erstellt wurde, können Sie weitere Einhängeziele erstellen, Einhängeziele löschen und die Sicherheitsgruppen für Einhängeziele ändern. Wenn Sie die VPC für Mount-Ziele ändern möchten, müssen Sie zuerst die vorhandenen Mount-Ziele löschen.

Note

Die IP-Adresse eines vorhandenen Mountingziels kann nicht geändert werden. Um eine IP-Adresse zu ändern, müssen Sie das Mount-Ziel löschen und ein neues mit der neuen Adresse erstellen.

Topics

- [Mount-Ziele und Availability Zones](#)
- [Erstellen von Bereitstellungszielen](#)
- [Mount-Ziele werden gelöscht](#)
- [Änderung der Mount-Ziel-VPC](#)
- [Sicherheitsgruppen für Mount-Ziele ändern](#)

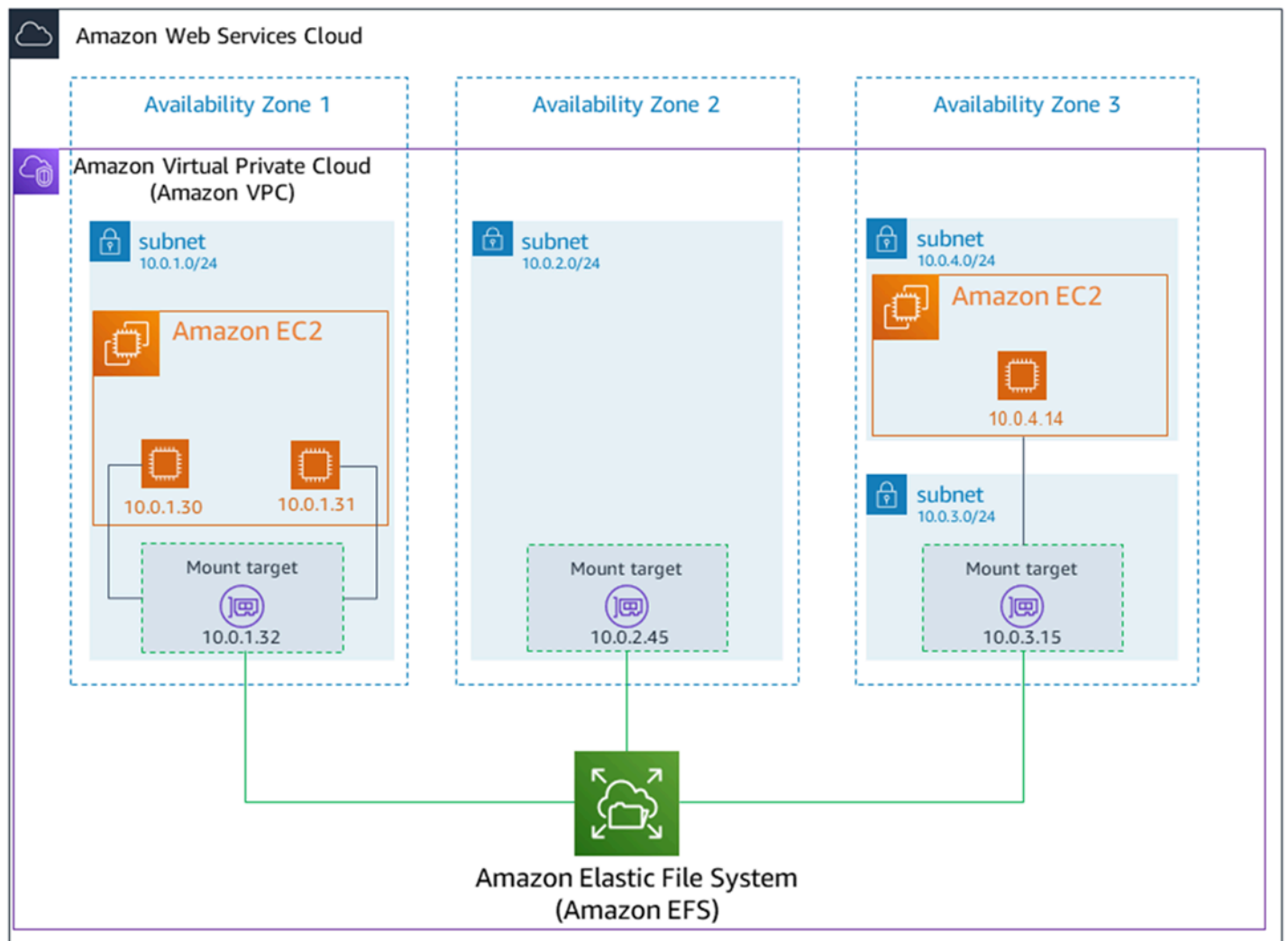
Mount-Ziele und Availability Zones

Für EFS-Dateisysteme, die regionale Speicherklassen verwenden, können Sie in jeder Availability Zone in einer ein Mount-Ziel erstellen AWS-Region.

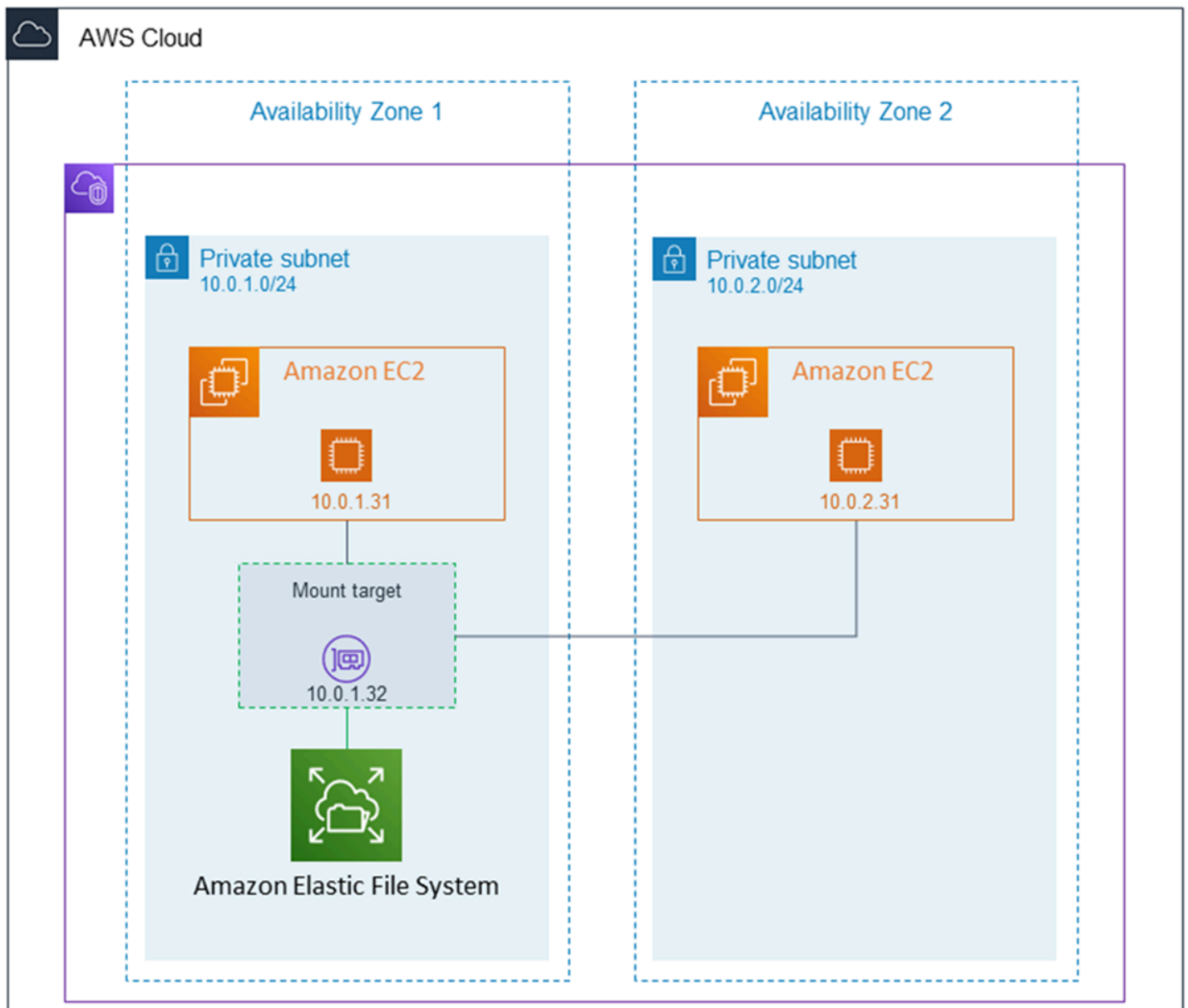
Für One-Zone-Dateisysteme können Sie nur ein einziges Mount-Ziel erstellen, das sich in der gleichen Availability Zone wie das Dateisystem befindet. Anschließend können Sie das Dateisystem auf Recheninstanzen wie Amazon EC2, Amazon ECS und AWS Lambda in Ihrer Virtual Private Cloud (VPC) mounten.

Das folgende Diagramm zeigt ein regionales Dateisystem mit Mount-Zielen, die in allen Availability Zones der VPC erstellt wurden. Die Abbildung zeigt drei EC2-Instances, die in verschiedenen VPC-Subnetzen gestartet wurden und auf ein EFS-Dateisystem zugreifen. Darüber hinaus ist in der Abbildung in jeder Availability Zone (unabhängig von der Anzahl der Subnetze in jeder Availability Zone) ein Mountingziel zu sehen.

Sie können nur ein Mountingziel pro Availability Zone erstellen. Wenn eine Availability Zone über mehrere Subnetze verfügt, wie in einer der Zones in der Abbildung dargestellt, erstellen Sie nur in einem Subnetz ein Mountingziel. Solange es in einer Availability Zone ein Mountingziel gibt, können die EC2-Instances, die in einem der Subnetze der Zone gestartet wurden, dasselbe Mountingziel gemeinsam verwenden.



Das folgende Diagramm zeigt ein One-Zone-Dateisystem mit einem einzigen Mount-Ziel, das sich in der gleichen Availability Zone wie das Dateisystem befindet. Für den Zugriff auf das Dateisystem mithilfe der EC2-Instance in der us-west2c Availability Zone fallen Datenzugriffsgebühren an, da sich die Instance in einer anderen Availability Zone als das Mount-Ziel befindet.



Die Sicherheitsgruppe für das Mountingziel fungiert als virtuelle Firewall, die den Datenverkehr steuert. So legt sie etwa fest, welche Clients auf das Dateisystem zugreifen können. In diesem Abschnitt wird Folgendes erklärt:

- Verwalten von Mount-Ziel-Sicherheitsgruppen und die Ermöglichung von Datenverkehr.
- Mounten des Dateisystems auf Ihren Clients.
- Überlegungen zu Berechtigungen auf NFS-Ebene.

Anfänglich hat nur der Root-Benutzer auf der Amazon EC2 EC2-Instance read-write-execute Berechtigungen für das Dateisystem. Dieses Thema erläutert die Berechtigungen auf NFS-Ebene

und zeigt Beispiele für die Gewährung von Berechtigungen in verbreiteten Szenarien. Weitere Informationen finden Sie unter [Benutzer, Gruppen und Berechtigungen auf NFS-Ebene \(Network File System\)](#).

Erstellen von Bereitstellungszielen

Um auf ein EFS-Dateisystem in einer VPC zuzugreifen, müssen Sie Mount-Ziele für das Dateisystem erstellen.

Für ein EFS-Dateisystem gilt Folgendes:

- Sie können Mount-Ziele für das Dateisystem jeweils in einer VPC erstellen. Wenn Sie von einer anderen VPC aus auf das Dateisystem zugreifen möchten, müssen Sie die Mount-Ziele aus der aktuellen VPC löschen und dann neue Mount-Ziele in der anderen VPC erstellen. Weitere Informationen finden Sie unter [Änderung der Mount-Ziel-VPC](#).
- Falls die VPC über mehrere Subnetze in einer Availability Zone verfügt, können Sie nur in einem dieser Subnetze ein Mountingziel erstellen. Alle EC2-Instances in der Availability Zone können dieses einzelne Mountingziel gemeinsam verwenden.
- Sie sollten mindestens in jeder Availability Zone, von der aus Sie auf das Dateisystem zugreifen möchten, ein Mount-Ziel erstellen.

Note

Es gibt Kostenüberlegungen für das Mounten eines Dateisystems auf einer EC2-Instance in einer Availability Zone über ein in einer anderen Availability Zone erstelltes Mountingziel. Weitere Informationen finden Sie unter [Amazon EFS-Preise](#). Wenn Sie immer ein für die Availability Zone der Instance lokales Mountingziel verwenden, vermeiden Sie darüber hinaus ein Teilausfallszenario. Wenn die Zone des Mountingziels ausfällt, können Sie über das betreffende Mountingziel nicht mehr auf Ihr Dateisystem zugreifen.

Sie können Mount-Ziele für ein Dateisystem mithilfe von AWS-Managementkonsole AWS CLI, oder programmgesteuert mit dem erstellen. AWS SDKs In der Konsole können Sie Bereitstellungsziele erstellen, wenn Sie das Dateisystem erstellen oder nachdem das Dateisystem erstellt wurde. Anweisungen zum Erstellen von Einhängezielen beim Erstellen eines Dateisystems finden Sie unter [Benutzerdefiniertes Erstellen mithilfe der Konsole](#).

Verwenden der Konsole

Gehen Sie wie folgt vor, um Einhängeziele zu einem vorhandenen EFS-Dateisystem hinzuzufügen.

So erstellen Sie ein Mount-Ziel auf einem EFS-Dateisystem

1. Öffnen Sie die Amazon Elastic File System-Konsole unter <https://console.aws.amazon.com/efs/>.
2. Wählen Sie im linken Navigationsbereich Dateisysteme und dann das Dateisystem aus, für das Sie die VPC ändern möchten.
3. Wählen Sie Netzwerk und anschließend Verwalten, um die Mount-Ziele für das Dateisystem anzuzeigen.
4. Wählen Sie das Dateisystem, für das Sie Einhängeziele hinzufügen möchten, indem Sie den Namen oder die Dateisystem-ID auswählen.

Note

Für One-Zone-Dateisysteme können Sie nur ein einziges Mount-Ziel erstellen, das sich in der gleichen Availability Zone wie das Dateisystem befindet.

5. Wählen Sie für Dateisysteme, die regionale EFS-Speicherklassen verwenden, für jedes Einhängeziel, das Sie für das Dateisystem erstellen möchten, die Option Einhängeziel hinzufügen aus.
6. Definieren Sie die Einstellungen für das Mount-Ziel:
 - a. Wählen Sie die Availability Zone und die Subnetz-ID für das Mount-Ziel.
 - b. Wählen Sie für den IP-Adresstyp aus, dass IPv4 nur IPv4 Adressen unterstützt werden sollen, IPv6 nur IPv6 Adressen unterstützt werden sollen, oder Dual-Stack, um beide IPv4 Adressen zu unterstützen. IPv6

Note

Der IP-Adresstyp muss mit dem IP-Typ des Subnetzes übereinstimmen. Darüber hinaus hat der IP-Adresstyp Vorrang vor dem IP-Adressierungsattribut Ihres Subnetzes. Wenn der IP-Adresstyp beispielsweise IPv4 -only ist und das IPv6 Adressierungsattribut für Ihr Subnetz aktiviert ist, erhalten die im Subnetz erstellten Netzwerkschnittstellen eine IPv4 Adresse aus dem Bereich des Subnetzes. Weitere

Informationen finden Sie unter [Ändern des öffentlichen IP-Adressierungsattributs Ihres Subnetzes](#).

- c. Wenn Sie die IP-Adresse kennen, an der Sie das Mount-Ziel platzieren möchten, geben Sie sie in das IP-Adressfeld ein, das dem IP-Adresstyp entspricht. Wenn Sie keinen Wert angeben, wählt Amazon EFS eine ungenutzte IP-Adresse aus dem angegebenen Subnetz aus.

 Note

Sie können die IP-Adresse eines Mount-Ziels nicht ändern, nachdem es erstellt wurde. Um eine IP-Adresse zu ändern, müssen Sie das Mount-Ziel löschen und ein neues mit der neuen Adresse erstellen.

7. Wählen Sie mindestens eine Sicherheitsgruppe aus, die dem Mount-Ziel zugeordnet werden soll. Sie können [die Sicherheitsgruppen später ändern](#).
8. Wählen Sie Speichern.

Mit dem AWS CLI

Dieser Abschnitt enthält Beispiele für die Erstellung eines Mount-Ziels AWS CLI mit dem `create-mount-target` Befehl. Der äquivalente API-Befehl lautet [CreateMountTarget](#).

- Wenn Sie keinen IP-Adresstyp für das Mount-Ziel angeben, wird IPv4 -only verwendet.
- Wenn Sie keine IP-Adresse für das Mount-Ziel angeben, weist Amazon EFS eine verfügbare Adresse im angegebenen Subnetz zu.
- Der IP-Adresstyp hat Vorrang vor dem IP-Adressierungsattribut Ihres Subnetzes. Wenn der IP-Adresstyp beispielsweise IPv4 -only ist und das IPv6 Adressierungsattribut für Ihr Subnetz aktiviert ist, erhalten die im Subnetz erstellten Netzwerkschnittstellen eine IPv4 Adresse aus dem Bereich des Subnetzes. Weitere Informationen finden Sie unter [Ändern des öffentlichen IP-Adressierungsattributs Ihres Subnetzes](#).

 Note

Für One-Zone-Dateisysteme können Sie nur ein einziges Mount-Ziel erstellen, das sich in der gleichen Availability Zone wie das Dateisystem befindet.

Beispiel: Erstellen Sie ein Mount-Ziel an einer verfügbaren Adresse in einem Subnetz IPv4

Der folgende Befehl gibt das Dateisystem, das Subnetz und die Sicherheitsgruppe für das Mount-Ziel an. Das Ziel wird an einer verfügbaren IPv4 Adresse im angegebenen Subnetz erstellt.

```
$ aws efs create-mount-target \  
--file-system-id file-system-id \  
--subnet-id subnet-id \  
--security-group ID-of-the-security-group-created-for-mount-target \  
--region aws-region \  

```

Das folgende Beispiel zeigt den Befehl mit Beispieldaten.

```
$ aws efs create-mount-target \  
--file-system-id fs-0123456789abcdef1 \  
--subnet-id subnet-b3983dc4 \  
--security-group sg-01234567 \  
--region us-east-2 \  

```

Nach der erfolgreichen Erstellung des Mountingziels gibt Amazon EFS die Beschreibung des Mountingziels als JSON wie im folgenden Beispiel gezeigt aus.

```
{  
  "OwnerId": "111122223333"  
  "MountTargetId": "fsmt-f9a14450",  
  "FileSystemId": "fs-0123456789abcdef1",  
  "SubnetId": "subnet-b3983dc4",  
  "LifecycleState": "available",  
  "IpAddress": "10.0.1.24",  
  "NetworkInterfaceId": "eni-3851ec4e",  
  "AvailabilityZoneId": "use2-az1",  
  "AvailabilityZoneName": "us-east-2a",  
  "VpcId": "vpc-3c39ef57"  
}
```

Beispiel: Erstellen Sie ein Mount-Ziel an einer bestimmten Adresse IPv4

Der folgende Befehl gibt das Dateisystem, das Subnetz, die Sicherheitsgruppe und die IPv4 Adresse an, die für das Mount-Ziel verwendet werden sollen. Das Ziel wird an der angegebenen IPv4 Adresse im angegebenen Subnetz erstellt.

```
$ aws efs create-mount-target \  
--file-system-id file-system-id \  
--subnet-id subnet-id \  
--security-group ID-of-the-security-group-created-for-mount-target \  
--ip-address IPv4-address \  
--region aws-region \  

```

Das folgende Beispiel zeigt den Befehl mit Beispieldaten.

```
$ aws efs create-mount-target \  
--file-system-id fs-0123456789abcdef1 \  
--subnet-id subnet-b3983dc4 \  
--security-group sg-01234567 \  
--ip-address 10.0.1.24 \  
--region us-east-2 \  

```

Nach der erfolgreichen Erstellung des Mountingziels gibt Amazon EFS die Beschreibung des Mountingziels als JSON wie im folgenden Beispiel gezeigt aus.

```
{  
  "OwnerId": "111122223333"  
  "MountTargetId": "fsmt-f9a14450",  
  "FileSystemId": "fs-0123456789abcdef1",  
  "SubnetId": "subnet-b3983dc4",  
  "LifeCycleState": "available",  
  "IpAddress": "10.0.1.24",  
  "NetworkInterfaceId": "eni-3851ec4e",  
  "AvailabilityZoneId": "use2-az1",  
  "AvailabilityZoneName": "us-east-2a",  
  "VpcId": "vpc-3c39ef57"  
}
```

Beispiel: Erstellen Sie ein Mount-Ziel an einer bestimmten Adresse IPv6

Der folgende Befehl gibt das Dateisystem, das Subnetz, die Sicherheitsgruppe und die IPv6 Adresse an, die für das Mount-Ziel verwendet werden sollen. Das Ziel wird an der angegebenen IPv6 Adresse im angegebenen Subnetz erstellt.

```
$ aws efs create-mount-target \  
--file-system-id file-system-id \  
--subnet-id subnet-id \  

```

```
--security-group ID-of-the-security-group-created-for-mount-target \  
--ip-address-type IP-address-type \  
--ipv6-address IPv6-address \  
--region aws-region \
```

Das folgende Beispiel zeigt den Befehl mit Beispieldaten.

```
$ aws efs create-mount-target \  
--file-system-id fs-0123456789abcdef1 \  
--subnet-id subnet-b3983dc4 \  
--security-group sg-01234567 \  
--ip-address-type IPV6_ONLY \  
--ipv6-address 2001:0db8:85a3:0000:0000:8a2e:0370:7334 \  
--region us-east-2 \
```

Nach der erfolgreichen Erstellung des Mountingziels gibt Amazon EFS die Beschreibung des Mountingziels als JSON wie im folgenden Beispiel gezeigt aus.

```
{  
  "OwnerId": "111122223333"  
  "MountTargetId": "fsmt-f9a14450",  
  "FileSystemId": "fs-0123456789abcdef1",  
  "SubnetId": "subnet-b3983dc4",  
  "LifecycleState": "available",  
  "Ipv6Address": "2001:0db8:85a3:0000:0000:8a2e:0370:7334",  
  "NetworkInterfaceId": "eni-3851ec4e",  
  "AvailabilityZoneId": "use2-az1",  
  "AvailabilityZoneName": "us-east-2a",  
  "VpcId": "vpc-3c39ef57"  
}
```

Beispiel: Erstellen Sie ein Mount-Ziel an einer verfügbaren IPv4 Adresse im Dual-Stack-Subnetz

Der Befehl gibt das Dateisystem, das Subnetz, die Sicherheitsgruppe, den Dual-Stack-IP-Adresstyp und die IPv6 Adresse für das Mount-Ziel an. Das Ziel wird an einer verfügbaren IPv4 Adresse und der angegebenen IPv6 Adresse im Dual-Stack-Subnetz erstellt.

```
$ aws efs create-mount-target \  
--file-system-id file-system-id \  
--subnet-id subnet-id \  
--security-group ID-of-the-security-group-created-for-mount-target \
```

```
--ip-address-type IP-address-type
--ipv6-address IPv6-address \
--region aws-region \
```

Das folgende Beispiel zeigt den Befehl mit Beispieldaten.

```
$ aws efs create-mount-target \
--file-system-id fs-0123456789abcdef1 \
--subnet-id subnet-b3983dc4 \
--security-group sg-01234567 \
--ip-address-type DUAL_STACK \
--ipv6-address 2001:0db8:85a3:0000:0000:8a2e:0370:7334 \
--region us-east-2 \
```

Nach der erfolgreichen Erstellung des Mountingziels gibt Amazon EFS die Beschreibung des Mountingziels als JSON wie im folgenden Beispiel gezeigt aus.

```
{
  "OwnerId": "111122223333"
  "MountTargetId": "fsmt-f9a14450",
  "FileSystemId": "fs-0123456789abcdef1",
  "SubnetId": "subnet-b3983dc4",
  "LifecycleState": "available",
  "IpAddress": "10.0.1.24",
  "Ipv6Address": "2001:0db8:85a3:0000:0000:8a2e:0370:7334",
  "NetworkInterfaceId": "eni-3851ec4e",
  "AvailabilityZoneId": "use2-az1",
  "AvailabilityZoneName": "us-east-2a",
  "VpcId": "vpc-3c39ef57"
}
```

Mount-Ziele werden gelöscht

Wenn Sie ein Mount-Ziel löschen, unterbricht der Vorgang gewaltsam alle Mounts des Dateisystems, wodurch Instanzen oder Anwendungen, die diese Mounts verwenden, gestört werden könnten. Um eine Anwendungsunterbrechung zu vermeiden, stoppen Sie die Anwendungen und heben Sie den Dateisystem-Mount auf, bevor Sie das Mountingziel löschen. Weitere Informationen finden Sie unter [Aufheben des Mountings von Dateisystemen](#).

Sie können Einhängeziele für ein Dateisystem mit der AWS-Managementkonsole, der AWS CLI, oder programmgesteuert mit den AWS SDKs löschen.

Verwenden der Konsole

Gehen Sie wie folgt vor, um Mount-Ziele für ein vorhandenes EFS-Dateisystem zu löschen.

So löschen Sie Mount-Ziele in einem EFS-Dateisystem

1. Hängen Sie das Dateisystem aus. Detaillierte Anweisungen finden Sie unter [Aufheben des Mountings von Dateisystemen](#).
2. Öffnen Sie die Amazon Elastic File System-Konsole unter <https://console.aws.amazon.com/efs/>.
3. Wählen Sie im linken Navigationsbereich Dateisysteme und dann das Dateisystem aus, für das Sie das Mount-Ziel löschen möchten.
4. Wählen Sie Netzwerk und anschließend Verwalten, um die Mount-Ziele für das Dateisystem anzuzeigen.
5. Wählen Sie für jedes Mount-Ziel, das Sie löschen möchten, die Option Entfernen aus.
6. Wählen Sie Speichern.

Verwenden Sie den AWS CLI

Um ein vorhandenes Mount-Ziel zu löschen, verwenden Sie den `delete-mount-target` AWS CLI Befehl (die entsprechende Operation ist [DeleteMountTarget](#)), wie im Folgenden gezeigt.

Note

Bevor Sie ein Mount-Ziel löschen, müssen Sie das Mounting des Dateisystems aufheben.

```
$ aws efs delete-mount-target \  
--mount-target-id mount-target-ID-to-delete \  
--region aws-region-where-mount-target-exists
```

Im Folgenden finden Sie Beispieldaten.

```
$ aws efs delete-mount-target \  
--mount-target-id fsmt-5751852e \  
--region us-east-2 \  

```


Änderung der Mount-Ziel-VPC

Sie können jeweils ein EFS-Dateisystem in einer VPC verwenden, die auf dem Amazon VPC-Service basiert. Sie erstellen also Mountingziele in einer VPC für Ihr Dateisystem und verwenden diese Mountingziele, um Zugriff auf das Dateisystem zu ermöglichen.

Sie können das EFS-Dateisystem von diesen Zielen aus mounten:

- Amazon-EC2-Instances in derselben VPC
- EC2-Instances in einer VPC, die durch VPC-Peering verbunden sind
- Lokale Server mithilfe von AWS Direct Connect
- Lokale Server über ein AWS virtuelles privates Netzwerk (VPN) mithilfe von Amazon VPC

Eine VPC-Peering-Verbindung ist eine Netzwerkverbindung zwischen zwei VPCs , mit der Sie den Verkehr zwischen ihnen weiterleiten können. Die Verbindung kann das private Internetprotokoll Version 4 (IPv4) oder Version 6 (IPv6) verwenden. Weitere Informationen dazu, wie Amazon EFS mit VPC-Peering funktioniert, finden Sie unter [Mounten von EFS-Dateisystemen von einer anderen AWS-Konto oder VPC](#).

So ändern Sie die VPC für ein Dateisystem

Die folgenden Schritte führen Sie aus, um die VPC für die Netzwerkkonfiguration eines EFS-Dateisystems zu ändern.

1. Löschen Sie jedes dem Dateisystem zugewiesene Mount-Ziel. Detaillierte Anweisungen finden Sie unter [Mount-Ziele werden gelöscht](#).
2. Wenn der Status der Mount-Ziele für jedes Mount-Ziel Gelöscht lautet, weisen Sie die neue VPC zu und erstellen Sie neue Mount-Ziele für das Dateisystem. Detaillierte Anweisungen finden Sie unter [Erstellen von Bereitstellungszielen](#).

Sicherheitsgruppen für Mount-Ziele ändern

Sicherheitsgruppen definieren den ein- und ausgehenden Zugriff. Wenn Sie einem Mountingziel zugeordnete Sicherheitsgruppen ändern, müssen Sie darauf achten, dass Sie den erforderlichen eingehenden/ausgehenden Zugriff zulassen. Dies ermöglicht es der EC2-Instance, mit dem Dateisystem zu kommunizieren. Weitere Informationen zu Sicherheitsgruppen finden Sie unter [Verwenden von VPC-Sicherheitsgruppen](#).

Sie können Sicherheitsgruppen für das Mount-Ziel eines Dateisystems hinzufügen oder entfernen, indem Sie den AWS-Managementkonsole, oder programmgesteuert verwenden AWS CLI, indem Sie den verwenden. AWS SDKs

Verwenden der Konsole

Um Sicherheitsgruppen für Mount-Ziele zu ändern

Gehen Sie wie folgt vor, um Mount-Zielsicherheitsgruppen für ein vorhandenes EFS-Dateisystem hinzuzufügen oder zu entfernen.

1. Öffnen Sie die Amazon Elastic File System-Konsole unter <https://console.aws.amazon.com/efs/>.
2. Wählen Sie im linken Navigationsbereich Dateisysteme und dann das Dateisystem aus, für das Sie Mount-Ziele verwalten möchten.
3. Wählen Sie Netzwerk und anschließend Verwalten, um die Mount-Ziele für das Dateisystem anzuzeigen.
4. Um eine Sicherheitsgruppe aus einem Mount-Ziel zu entfernen, wählen Sie X neben der Sicherheitsgruppen-ID aus.
5. Um einem Mount-Ziel eine Sicherheitsgruppe hinzuzufügen, wählen Sie die Sicherheit aus der Liste der Sicherheitsgruppen aus.
6. Wählen Sie Speichern.

Verwenden Sie den AWS CLI

Um Sicherheitsgruppen zu ändern, die für ein Mount-Ziel gültig sind, verwenden Sie den `modify-mount-target-security-group` AWS CLI Befehl (der entsprechende Vorgang ist [ModifyMountTargetSecurityGroups](#)), um alle vorhandenen Sicherheitsgruppen zu ersetzen, wie im Folgenden dargestellt.

```
$ aws efs modify-mount-target-security-groups \
--mount-target-id mount-target-ID-whose-configuration-to-update \
--security-groups security-group-ids-separated-by-space \
--region aws-region-where-mount-target-exists \
--profile adminuser
```

Im Folgenden finden Sie Beispieldaten.

```
$ aws efs modify-mount-target-security-groups \
```

```
--mount-target-id fsmt-5751852e \  
--security-groups sg-1004395a sg-1114433a \  
--region us-east-2
```

Verwalten des Dateisystemdurchsatzes

Elastischer Durchsatz ist der Standarddurchsatzmodus in der Amazon EFS-Konsole und wird für die meisten Anwendungsfälle empfohlen. Mit Elastic-Durchsatz wird die Leistung automatisch nach oben oder unten skaliert, um den Anforderungen Ihrer Workload-Aktivität gerecht zu werden. Wenn Sie jedoch die spezifischen Zugriffsmuster für Ihre Workloads kennen (einschließlich Durchsatz, Latenz und Speicherbedarf), können Sie den Durchsatzmodus ändern.

Note

Elastic Throughput ist zwar darauf ausgelegt, elastisch mit Ihrem Durchsatz zu skalieren, wir empfehlen jedoch, im Rahmen Ihrer betrieblichen Best Practices eine angemessene Steuerung durch die Überwachung von Metriken mit CloudWatch (gemessenen I/O Bytes) Daten und Nutzungswarnungen zu implementieren. Dies hilft Ihnen dabei, eine optimale Ressourcennutzung aufrechtzuerhalten und Ihre geplanten Betriebsparameter einzuhalten. Weitere Informationen finden Sie unter [Metriken mit Amazon überwachen CloudWatch](#).

Zu den anderen Durchsatzmodi, die Sie wählen können, gehören:

- Bereitgestellter Durchsatz – Sie geben einen Durchsatz an, den das Dateisystem unabhängig von der Größe oder dem Burst-Guthabensaldo des Dateisystems erreichen kann.
- Bursting-Durchsatz – Der Durchsatz ändert sich mit der Menge an Speicherplatz in Ihrem Dateisystem und unterstützt das Bursting auf höhere Levels für bis zu 12 Stunden pro Tag.

Weitere Informationen zu den Amazon EFS-Durchsatzmodi finden Sie unter [Durchsatzmodi](#).

Note

Sie können den Durchsatzmodus und die bereitgestellte Durchsatzmenge ändern, sobald das Dateisystem verfügbar ist. Das Ändern des Durchsatzmodus führt nicht zu Ausfallzeiten der Anwendung. Jedes Mal, wenn Sie das Dateisystem auf „Bereitgestellter Durchsatz“ ändern oder den bereitgestellten Durchsatz erhöhen, müssen Sie jedoch mindestens 24 Stunden

warten, bevor Sie den Durchsatzmodus erneut ändern oder den bereitgestellten Durchsatz verringern können.

Sie können den Durchsatzmodus des Dateisystems mithilfe der Amazon EFS-Konsole, der AWS Command Line Interface (AWS CLI) und der Amazon EFS-API verwalten.

Verwenden der Konsole

Um den Durchsatz des Dateisystems zu verwalten

1. Öffnen Sie die Amazon Elastic File System-Konsole unter <https://console.aws.amazon.com/efs/>.
2. Wählen Sie im linken Navigationsbereich Dateisysteme, um die Liste der EFS-Dateisysteme in Ihrem Konto anzuzeigen.
3. Wählen Sie das Dateisystem aus, für das Sie den Durchsatzmodus ändern möchten.
4. Wählen Sie auf der Seite mit den Dateisystemdetails im Abschnitt Allgemein die Option Bearbeiten aus. Die Seite Bearbeiten wird angezeigt.
5. Ändern Sie die Einstellung für den Durchsatzmodus.
 - Um „Elastic-Durchsatz“ oder „Bereitgestellter Durchsatz“ zu verwenden, wählen Sie Erweitert und dann Elastic oder Bereitgestellt.

Wenn Sie Bereitgestellt wählen, geben Sie im Feld Provisioned Throughput (MIB/s) die Menge des Durchsatzes ein, der für Dateisystemanforderungen bereitgestellt werden soll. Der Maximale Lesedurchsatz wird dreimal so hoch angezeigt wie der von Ihnen eingegebene Durchsatz. EFS-Dateisysteme messen Leseanforderungen mit einem Drittel der Rate anderer Anforderungen. Nachdem Sie den Durchsatz eingegeben haben, wird eine Schätzung der monatlichen Kosten für das Dateisystem angezeigt.

Note

Sie können den Durchsatzmodus und die bereitgestellte Durchsatzmenge ändern, sobald das Dateisystem verfügbar ist. Jedes Mal, wenn Sie den Dateisystemdurchsatz auf Bereitgestellt ändern oder den bereitgestellten Durchsatz erhöhen, müssen Sie jedoch mindestens 24 Stunden warten, bevor Sie den Durchsatzmodus erneut ändern oder die bereitgestellte Menge verringern können.

- Um den Bursting-Durchsatz zu verwenden, wählen Sie Bursting.

Weitere Informationen zur Auswahl des richtigen Durchsatzmodus für Ihre Leistungsanforderungen finden Sie unter [Durchsatzmodi](#).

6. Wählen Sie Änderungen speichern aus, um die Änderungen zu speichern.

Unter Verwendung des AWS CLI

Verwenden Sie den [update-file-system](#) CLI-Befehl oder die [UpdateFileSystem](#) API-Aktion, um den Durchsatzmodus eines Dateisystems zu ändern.

Verwaltung des Speicherlebenszyklus

Sie können Ihre Dateisysteme so verwalten, dass sie während ihres gesamten Lebenszyklus über kostengünstigen Speicher verfügen. Verwenden Sie das Lebenszyklusmanagement, um Daten automatisch zwischen Speicherklassen entsprechend der Lebenszykluskonfiguration für das Dateisystem zu übertragen. Die Lebenszykluskonfiguration besteht aus drei Lebenszyklusrichtlinien, die Sie für das Dateisystem festlegen.

Lebenszyklusrichtlinien weisen das Lebenszyklusmanagement an, wann Dateien in die Speicherklassen EFS Infrequent Access (IA) und EFS Archive übertragen und aus diesen herausgenommen werden müssen. Dies basiert darauf, wann in der Speicherklasse „Standard“ zuletzt auf die Dateien zugegriffen wurde. Um den Zeitpunkt des letzten Zugriffs in der Standard-Speicherklasse zu ermitteln, verfolgt ein interner Timer, wann zuletzt auf eine Datei zugegriffen wurde (dies sind nicht die POSIX-Dateisystemattribute, die öffentlich einsehbar sind). Immer wenn auf eine Datei in Standard zugegriffen wird, wird der Lebenszyklusmanagement-Timer zurückgesetzt.

Lebenszyklusrichtlinien gelten für das gesamte EFS-Dateisystem.

Die EFS-Lebenszyklusrichtlinien lauten:

- Umstellung auf IA — Weist dem Lebenszyklusmanagement an, wann Dateien in die Speicherkategorie Infrequent Access (IA) verschoben werden sollen. Diese Speicherkategorie ist kostenoptimiert für Daten, auf die nur ein paar Mal pro Quartal zugegriffen wird. Standardmäßig werden Dateien, auf die nicht innerhalb der Standardspeicherkategorie oder innerhalb von 30 Tagen zugegriffen wird, auf IA umgestellt.
- Übergang ins Archiv — Weist dem Lebenszyklusmanagement an, wann Dateien von der Standard- oder IA-Speicherkategorie in die Archivspeicherkategorie verschoben werden sollen. Diese

ist kostenoptimiert für Daten, auf die nur ein paar Mal pro Jahr oder weniger zugegriffen wird. Standardmäßig werden Dateien, auf die 90 Tage lang nicht in der Standard-Speicherklasse zugegriffen wurde, in die Speicherklasse Archiv umgestellt.

- **Übergang zur Standardspeicherklasse** — Weist dem Lebenszyklusmanagement an, ob Dateien aus der Speicherklasse IA oder Archiv wieder in die Speicherklasse Standard umgestellt werden sollen, wenn auf die Dateien in der IA- oder Archive-Speicherklasse zugegriffen wird. Standardmäßig werden Dateien nicht zurück in die Standard-Speicherklasse verschoben und sie verbleiben in der IA- oder Archive-Speicherklasse, wenn auf sie zugegriffen wird.

Für leistungsabhängige Anwendungsfälle, die die schnellste Latenzzeit erfordern (z. B. Anwendungen, die mit einer großen Menge kleiner Dateien arbeiten), sollten Sie Dateien beim ersten Zugriff in den Standardspeicher verschieben.

Weitere Informationen zur Konfiguration der Lebenszyklusrichtlinien für ein Dateisystem finden Sie unter [Konfiguration von Lebenszyklusrichtlinien](#).

Dateisystemoperationen für die Lebenszyklusverwaltung

Dateisystemoperationen für die Lebenszyklusverwaltung haben eine niedrigere Priorität als Operationen für EFS-Dateisystem-Workloads. Die Zeit, die für die Übertragung von Dateien in oder aus dem IA- und Archivspeicher benötigt wird, hängt von der Dateigröße und der Arbeitslast des Dateisystems ab. Beispielsweise kann die Übertragung von Millionen kleiner Dateien länger dauern als die Übertragung weniger großer Dateien mit derselben Gesamtspeichergröße.

Datei-Metadaten, einschließlich Dateinamen, Eigentümerinformationen und Dateisystem-Verzeichnisstruktur, werden immer im Standardspeicher gespeichert, um eine konsistente Metadaten-Leistung sicherzustellen.

Metadatenoperationen für Dateisysteme im IA- oder Archivspeicher, wie z. B. das Auflisten des Inhalts eines Verzeichnisses, zählen nicht als Dateizugriff. Während des Vorgangs zum Übertragen der Inhalte einer Datei in den IA- oder Archive-Speicher wird die Datei in der Standardspeicherklasse gespeichert und mit dem Standardspeichersatz abgerechnet.

Alle Schreibvorgänge in Dateien in den IA- oder Archive-Speicherklassen des Dateisystems werden zuerst in die Standard-Speicherklassen geschrieben und können dann nach 24 Stunden auf die entsprechende Speicherklasse umgestellt werden.

Konfiguration von Lebenszyklusrichtlinien

Wenn Sie mithilfe von ein EFS-Dateisystem mit den empfohlenen Einstellungen erstellen AWS-Managementkonsole, wird das Dateisystem automatisch mit der folgenden Standard-Lebenszykluskonfiguration konfiguriert:

- Übergang in IA ist auf 30 Tage seit dem letzten Zugriff festgelegt.
- Transition into Archive (Übertragung zu Archive) ist auf 90 Tage seit dem letzten Zugriff festgelegt.
- Übergang zum Standard ist auf Keine gesetzt.

Sie können die standardmäßigen Lebenszyklusrichtlinien ändern, wenn Sie ein Dateisystem mit benutzerdefinierten Einstellungen mithilfe von erstellen AWS-Managementkonsole oder wenn Sie ein Dateisystem mit dem erstellen AWS CLI. Sie können die Richtlinien auch nach der Erstellung des Dateisystems ändern, wie in den folgenden Verfahren beschrieben.

Verwenden der Konsole

Sie können den verwenden AWS-Managementkonsole , um die Lebenszyklusrichtlinien für ein vorhandenes Dateisystem festzulegen.

1. Melden Sie sich bei der an AWS-Managementkonsole und öffnen Sie die Amazon EFS-Konsole unter <https://console.aws.amazon.com/efs/>.
2. Wählen Sie Dateisysteme, um die Liste der Dateisysteme in Ihrem Konto anzuzeigen.
3. Wählen Sie das Dateisystem aus, für das Sie Lebenszyklusrichtlinien ändern möchten.
4. Wählen Sie auf der Seite mit den Dateisystemdetails im Abschnitt Allgemein die Option Bearbeiten aus. Die Seite Bearbeiten wird angezeigt.
5. Konfigurieren Sie für das Lebenszyklusmanagement die Lebenszyklusrichtlinien:
 - Stellen Sie „Übergang zu IA“ auf eine der verfügbaren Optionen ein. Um das Verschieben von Dateien in den IA-Speicher zu beenden, wählen Sie Keine.
 - Stellen Sie „Übergang ins Archiv“ auf eine der verfügbaren Optionen ein. Um das Verschieben von Dateien in den Archive-Speicher zu beenden, wählen Sie None (Keine) aus.
 - Stellen Sie Übergang zum Standard auf Beim ersten Zugriff, um Dateien, die sich im IA-Speicher befinden, in den Standardspeicher zu verschieben, wenn auf sie für Nicht-Metadaten-Operationen zugegriffen wird.

Um das Verschieben von Dateien vom IA- oder Archive-Speicher in den Standardspeicher beim ersten Zugriff zu beenden, wählen Sie Keine aus.

6. Wählen Sie Änderungen speichern aus, um Ihre Änderungen zu speichern.

Verwenden Sie den AWS CLI

Sie können den verwenden AWS CLI , um die Lebenszyklusrichtlinien eines Dateisystems festzulegen oder zu ändern.

- Führen Sie den [put-lifecycle-configuration](#) AWS CLI Befehl oder den [PutLifecycleConfiguration](#) API-Befehl aus und geben Sie dabei die Dateisystem-ID des Dateisystems an, für das Sie das Lebenszyklusmanagement verwalten.

```
$ aws efs put-lifecycle-configuration \
--file-system-id File-System-ID \
--lifecycle-policies "[{\"TransitionToIA\": \"AFTER_60_DAYS\"},
{\"TransitionToPrimaryStorageClass\": \"AFTER_1_ACCESS\"}, {\"TransitionToArchive\":
\"AFTER_90_DAYS\"}]" \
--region us-west-2 \
--profile adminuser
```

Sie erhalten die folgende Antwort.

```
{
  "LifecyclePolicies": [
    {
      "TransitionToIA": "AFTER_60_DAYS"
    },
    {
      "TransitionToPrimaryStorageClass": "AFTER_1_ACCESS"
    },
    {
      "TransitionToArchive": "AFTER_90_DAYS"
    }
  ]
}
```


So halten Sie die Lebenszyklusverwaltung für ein vorhandenes Dateisystem an (CLI)

- Führen Sie den Befehl `put-lifecycle-configuration` aus und geben Sie die ID des Dateisystems an, für das Sie die Lebenszyklusverwaltung anhalten möchten. Lassen Sie die Eigenschaft `--lifecycle-policies` leer.

```
$ aws efs put-lifecycle-configuration \  
--file-system-id File-System-ID \  
--lifecycle-policies \  
--region us-west-2 \  
--profile adminuser
```

Sie erhalten die folgende Antwort.

```
{  
  "LifecyclePolicies": []  
}
```

Überwachen von Amazon EFS

Die Überwachung ist ein wichtiger Bestandteil der Aufrechterhaltung der Zuverlässigkeit, Verfügbarkeit und Leistung von Amazon EFS und Ihren AWS Lösungen. Wir empfehlen Ihnen, Überwachungsdaten aus allen Teilen Ihrer AWS Lösung zu sammeln, damit Sie einen etwaigen Ausfall an mehreren Stellen leichter debuggen können. Bevor Sie mit der Überwachung von Amazon EFS beginnen, erstellen Sie einen Überwachungsplan, der Antworten auf folgende Fragen enthält:

- Was sind Ihre Ziele bei der Überwachung?
- Welche Ressourcen werden überwacht?
- Wie oft werden diese Ressourcen überwacht?
- Welche Überwachungstools werden verwendet?
- Wer soll die Überwachungsaufgaben ausführen?
- Wer soll benachrichtigt werden, wenn Fehler auftreten?

Im nächsten Schritt legen Sie eine Baseline für die normale Amazon-EFS-Leistung in Ihrer Umgebung fest, indem Sie die Leistung zu verschiedenen Zeiten und unter verschiedenen Lastbedingungen messen. Wenn Sie Amazon EFS überwachen, sollten Sie das Speichern historischer Überwachungsdaten in Betracht ziehen. Diese gespeicherten Daten bieten eine Basis für den Vergleich mit aktuellen Leistungsdaten, für die Identifikation normaler Leistungsmuster und Leistungsanomalien sowie für die Entwicklung von Verfahren für den Umgang mit Problemen.

Mit Amazon EFS können Sie beispielsweise den Netzwerkdurchsatz I/O für Lese-, Schreib- und Metadatenvorgänge, Client-Verbindungen und Burst-Guthabensalden für Ihre Dateisysteme überwachen. Wenn die Leistung außerhalb der festgelegten Baseline liegt, müssen Sie möglicherweise die Größe Ihres Dateisystems oder die Anzahl der verbundenen Clients modifizieren, um das Dateisystem für Ihren Workload zu optimieren.

Zur Festlegung eines Grundwertes sollten Sie mindestens die folgenden Elemente überwachen:

- Der Netzwerkdurchsatz Ihres Dateisystems.
- Die Anzahl von Client-Verbindungen mit einem Dateisystem.
- Die Bytezahl für jede Dateisystemoperation, einschließlich Datenlese-, Datenschreib- und Metadatenoperationen.

Themen

- [Tools zur Überwachung](#)
- [Wie Amazon EFS Dateisystem- und Objektgrößen meldet](#)
- [Anzeigen der Größe der Speicherklasse](#)
- [Metriken mit Amazon überwachen CloudWatch](#)
- [Protokollieren von Amazon EFS-API-Aufrufen mit AWS CloudTrail](#)

Tools zur Überwachung

AWS bietet verschiedene Tools, mit denen Sie Amazon EFS überwachen können. Sie können einige dieser Tools so konfigurieren, dass diese die Überwachung für Sie übernehmen, während bei anderen Tools ein manuelles Eingreifen nötig ist. Wir empfehlen, dass Sie die Überwachungsaufgaben möglichst automatisieren.

Automatisierte Überwachungstools

Sie können die folgenden automatisierten Tools zur Überwachung von Amazon EFS verwenden und auftretende Probleme melden:

- Amazon CloudWatch Alarms — Überwachen Sie eine einzelne Metrik über einen von Ihnen angegebenen Zeitraum und führen Sie eine oder mehrere Aktionen aus, die auf dem Wert der Metrik im Verhältnis zu einem bestimmten Schwellenwert über mehrere Zeiträume basieren. Die Aktion ist eine Benachrichtigung, die an ein Amazon Simple Notification Service (Amazon SNS) -Thema oder eine Amazon EC2 Auto Scaling Scaling-Richtlinie gesendet wird. CloudWatch Alarme lösen keine Aktionen aus, nur weil sie sich in einem bestimmten Status befinden. Der Status muss sich geändert haben und für eine bestimmte Anzahl von Zeiträumen beibehalten worden sein. Weitere Informationen finden Sie unter [Metriken mit Amazon überwachen CloudWatch](#).
- Amazon CloudWatch Logs — Überwachen, speichern und greifen Sie auf Ihre Protokolldateien aus AWS CloudTrail oder anderen Quellen zu. Weitere Informationen finden Sie unter [Was ist Amazon CloudWatch Logs?](#) im CloudWatch Amazon-Benutzerhandbuch.
- Amazon CloudWatch Events — Ordnen Sie Ereignisse zu und leiten Sie sie an eine oder mehrere Zielfunktionen oder Streams weiter, um Änderungen vorzunehmen, Statusinformationen zu erfassen und Korrekturmaßnahmen zu ergreifen. Weitere Informationen finden Sie unter [Was ist Amazon CloudWatch Events](#) im CloudWatch Amazon-Benutzerhandbuch.

- **AWS CloudTrail Protokollüberwachung** — Teilen Sie Protokolldateien zwischen Konten, überwachen CloudTrail Sie Protokolldateien in Echtzeit, indem Sie sie an CloudWatch Logs senden, schreiben Sie Protokollverarbeitungsanwendungen in Java und überprüfen Sie, ob sich Ihre Protokolldateien nach der Lieferung von nicht geändert haben CloudTrail. Weitere Informationen finden Sie unter [Arbeiten mit CloudTrail Protokolldateien](#) im AWS CloudTrail Benutzerhandbuch.

Manuelle Überwachungstools

Ein weiterer wichtiger Teil der Überwachung von Amazon EFS ist die manuelle Überwachung der Artikel, die von den CloudWatch Amazon-Alarmen nicht abgedeckt werden. Amazon EFS und andere AWS-Managementkonsole Dashboards bieten einen at-a-glance Überblick über den Zustand Ihrer AWS Umgebung. CloudWatch Wir empfehlen, dass Sie auch die Protokolldateien für die Dateisysteme überprüfen.

- In der Amazon-EFS-Konsole finden Sie die folgenden Elemente für Ihre Dateisysteme:
 - Die aktuelle gemessene Größe
 - Die Anzahl der Mountingziele
 - Lebenszyklusstatus
- CloudWatch Die Startseite zeigt:
 - Aktuelle Alarme und Status
 - Diagramme mit Alarmen und Ressourcen
 - Servicestatus

Darüber hinaus können CloudWatch Sie Folgendes verwenden:

- Erstellen Sie [benutzerdefinierte Dashboards](#) zur Überwachung der Services, die Sie verwenden.
- Aufzeichnen von Metrikdaten, um Probleme zu beheben und Trends zu erkennen.
- Suchen und durchsuchen Sie alle Ihre AWS Ressourcenmetriken.
- Erstellen und Bearbeiten von Alarmen, um über Probleme benachrichtigt zu werden.

Wie Amazon EFS Dateisystem- und Objektgrößen meldet

In den folgenden Abschnitten wird beschrieben, wie Amazon EFS Dateisystemgrößen, Objektgrößen innerhalb eines Dateisystems und Dateisystemdurchsatz meldet.

Messung von EFS-Dateisystemobjekten

Zu den Objekten, die Sie in einem EFS-Dateisystem anzeigen können, gehören reguläre Dateien, Verzeichnisse, symbolische Links und spezielle Dateien (FIFOs und Sockets). Jedes dieser Objekte wird auf 2 KiB (Kibibyte) Metadaten (für den Inode) und ein oder mehrere Schritte von 4 KiB Daten gemessen. In der folgenden Liste werden die gemessenen Datengrößen für verschiedene Arten von Dateisystemobjekten erläutert:

- **Normale Dateien:** Die gemessene Datengröße einer normalen Datei ist die logische Größe der Datei, gerundet auf den nächsten 4 KiB-Schritt. Bei Sparse-Dateien kann der Wert allerdings geringer sein.

Eine Sparse-Datei ist eine Datei, bei der nicht in alle Dateipositionen Daten geschrieben werden, bevor ihre logische Größe erreicht ist. Bei einer Sparse-Datei ist der tatsächlich verwendete Speicherplatz in einigen Fällen geringer als die auf den nächsten 4 KiB-Schritt gerundete logische Größe. In diesen Fällen meldet Amazon EFS den tatsächlich belegten Speicher als gemessene Datengröße.

- **Verzeichnisse** – Die gemessene Datengröße eines Verzeichnisses ist der für die Verzeichniseinträge und die Datenstruktur, in der diese enthalten sind, tatsächlich verwendete Speicher, gerundet auf den nächsten 4 KiB-Schritt. Die gemessene Datengröße beinhaltet nicht den tatsächlich von den Dateidaten verwendeten Speicherplatz.
- **Symbolische Verknüpfungen und besondere Dateien** – Die gemessene Datengröße für diese Objekte beträgt immer 4 KiB.

Wenn Amazon EFS den für ein Objekt belegten Speicherplatz über das NFSv4 .1 `space_used` -Attribut meldet, schließt es die aktuelle gemessene Datengröße des Objekts ein, nicht jedoch die Metadatengröße. Sie können zwei Dienstprogramme zum Messen der Datenträgerverwendung einer Datei verwenden, `du` und `stat`. Im Folgenden finden Sie ein Beispiel für die Verwendung des `du` Dienstprogramms für eine leere Datei, das die `-k` Option enthält, die Ausgabe in Kilobyte zurückzugeben.

```
$ du -k file
4      file
```

Das folgende Beispiel zeigt, wie das `stat` Hilfsprogramm für eine leere Datei verwendet wird, um den Speicherverbrauch der Datei zurückzugeben.

```
$ /usr/bin/stat --format="%b*%B" file | bc  
4096
```

Wenn Sie die Größe eines Verzeichnisses messen möchten, verwenden Sie das Dienstprogramm `stat`. Suchen Sie den Wert `Blocks`, und multiplizieren Sie diesen Wert mit der Blockgröße. Es folgt ein Beispiel für die Verwendung des Dienstprogramms `stat` mit einem leeren Verzeichnis:

```
$ /usr/bin/stat --format="%b*%B" . | bc  
4096
```

Gemessene Größe eines EFS-Dateisystems

Die gemessene Größe eines EFS-Dateisystems umfasst die Summe der Größen aller aktuellen Objekte in allen EFS-Speicherklassen. Die Größe jedes Objekts wird anhand einer repräsentativen Stichprobe berechnet, die die Größe des Objekts während der gemessenen Stunde, also beispielsweise der Stunde zwischen 08:00 Uhr und 09:00 Uhr, darstellt.

Eine leere Datei trägt beispielsweise 6 KiB (2 KiB Metadaten + 4 KiB Daten) zur gemessenen Größe des Dateisystems bei. Bei der Erstellung verfügt ein Dateisystem über ein einzelnes leeres Stammverzeichnis, daher beträgt die gemessene Größe 6 KiB.

Die gemessenen Größen eines bestimmten Dateisystems legen die Nutzung fest, die dem Konto des Besitzers für das betreffende Dateisystem und die betreffende Stunde in Rechnung gestellt wird.

Note

Die berechnete gemessene Größe stellt keinen konsistenten Snapshot des Dateisystems zu einem bestimmten Zeitpunkt während dieser Stunde dar. Sie stellt vielmehr die Größe der Objekte dar, die zu unterschiedlichen Zeiten innerhalb der Stunde oder möglicherweise der Stunde davor auf dem Dateisystem vorhanden waren. Diese Größe ist eine Summe der gemessenen Größe des Dateisystems zu dieser Stunde. Die gemessene Größe eines Dateisystems ist somit letztendlich mit den gemessenen Größen der gespeicherten Objekte konsistent, wenn es keine Schreibvorgänge in das Dateisystem gibt.

Sie können die gemessene Größe für ein EFS-Dateisystem auf folgende Weise anzeigen:

- Unter Verwendung des [describe-file-systems](#) AWS CLI Befehls und der [DescribeFileSystem](#) API-Operation umfasst die Antwort Folgendes:

```
"SizeInBytes":{
    "Timestamp": 1403301078,
    "Value": 29313744866,
    "ValueInIA": 675432,
    "ValueInStandard": 29312741784
    "ValueInArchive": 327650
}
```

Wobei die gemessene Größe von auch verwendet ValueInStandard wird, um den I/O Basisdurchsatz und die Burst-Raten für Dateisysteme zu bestimmen, die den [Bursting-Durchsatz](#) verwenden.

- Sehen Sie sich die StorageBytes CloudWatch Metrik an, die die gemessene Gesamtgröße der Daten in den einzelnen Speicherklassen anzeigt. Für weitere Informationen über die StorageBytes-Metrik siehe [CloudWatch Metriken für Amazon EFS](#).
- Führen Sie in Linux bei der Terminal-Eingabeaufforderung einer EC2-Instance den Befehl df aus.

Verwenden Sie den du Befehl im Stammverzeichnis des Dateisystems nicht zur Speichermessung, da die Antwort nicht den vollständigen Datensatz wiedergibt, der für die Messung Ihres Dateisystems verwendet wurde.

Note

Mithilfe der gemessenen Größe ValueInStandard werden auch Ihre Baseline- und Burst-Raten für den E/A-Durchsatz ermittelt. Weitere Informationen finden Sie unter [Bursting-Durchsatz](#).

Messung seltener Zugriffe und Archivierung der Speicherklassen

Die EFS-Speicherklassen Infrequent Access (IA) und Archive werden in Schritten von 4 KiB berechnet und haben eine Mindestabrechnungsgebühr von 128 KiB pro Datei. IA- und Archivdatei-Metadaten (2 KiB pro Datei) werden immer in der Standard-Speicherklasse gespeichert und gemessen. Die Support für Dateien, die kleiner als 128 KiB sind, ist nur für Lebenszyklusrichtlinien verfügbar, die am oder nach 12:00 Uhr PT am 26. November 2023 aktualisiert wurden. Der Datenzugriff für IA und Archivspeicher wird in Schritten von 128 KiB gemessen.

Sie können die StorageBytes CloudWatch Metrik verwenden, um die gemessene Datengröße in jeder Speicherklasse anzuzeigen. Die Metrik zeigt auch die Gesamtzahl der Byte an, die bei der Rundung kleiner Dateien innerhalb der Speicherklassen IA und Archive verbraucht werden. Weitere Informationen zum Anzeigen von CloudWatch Metriken finden Sie unter [Zugreifen auf CloudWatch Metriken für Amazon EFS](#). Für weitere Informationen über die StorageBytes-Metrik siehe [CloudWatch Metriken für Amazon EFS](#).

Mess-Durchsatz

Amazon EFS misst den Durchsatz für Leseanfragen auf ein Drittel der Geschwindigkeit der anderen I/O Dateisystemoperationen. Wenn Sie beispielsweise einen Lese- und Schreibdurchsatz von 30 Mebibyte pro Sekunde (MiBps) erzielen, zählt der Leseteil als 10 Mebibyte MiBps des effektiven Durchsatzes, der Schreibanteil zählt als 30 MiBps und der kombinierte gemessene Durchsatz beträgt 40. MiBps Dieser kombinierte Durchsatz, bereinigt um die Verbrauchsraten, spiegelt sich in der Metrik wider. MeteredIOBytes CloudWatch

Messung des Elastischen Durchsatzes

Wenn der Elastic Throughput-Modus für ein Dateisystem aktiviert ist, zahlen Sie nur für die Menge an Metadaten und Daten, die aus dem Dateisystem gelesen oder in das Dateisystem geschrieben werden. EFS-Dateisysteme, die den Elastic Throughput Mode Meter verwenden und Metadatenlesevorgänge als Lesevorgänge und Metadaten-Schreibvorgänge als Schreibvorgänge abrechnen. Metadatenoperationen werden in Schritten von 1 KiB nach den ersten 4 KiB gemessen. Datenoperationen werden nach den ersten 32 KiB in Schritten von 1 KiB gemessen.

Note

Elastic Throughput ist zwar darauf ausgelegt, elastisch mit Ihrem Durchsatz zu skalieren, wir empfehlen jedoch, im Rahmen Ihrer betrieblichen Best Practices eine angemessene Steuerung durch die Überwachung von Metriken mit CloudWatch (MeteredIOBytes) und Nutzungswarnungen zu implementieren. Dies hilft Ihnen dabei, eine optimale Ressourcennutzung aufrechtzuerhalten und Ihre geplanten Betriebsparameter einzuhalten. Weitere Informationen finden Sie unter [Metriken mit Amazon überwachen CloudWatch](#).

Messen des bereitgestellten Durchsatzes

Bei Dateisystemen, die den Durchsatzmodus Bereitgestellt verwenden, zahlen Sie nur für die Zeit, für die der Durchsatz aktiviert ist. Amazon EFS misst Dateisysteme mit aktiviertem

Bereitstellungsdurchsatzmodus einmal pro Stunde. Für die Messung, wenn der Bereitgestellte Durchsatzmodus auf weniger als eine Stunde eingestellt ist, berechnet Amazon EFS den Zeitdurchschnitt mit einer Genauigkeit von Millisekunden.

Anzeigen der Größe der Speicherklasse

Sie können mithilfe der Amazon EFS-Konsole, der oder der EFS-API anzeigen, wie viele Daten in jeder Speicherklasse Ihres Dateisystems gespeichert sind. AWS CLI

Verwenden der Konsole

Auf der Registerkarte Gemessene Größe auf der Seite mit den Dateisystemdetails wird die aktuelle gemessene Größe des Dateisystems in binären Vielfachen von Byte (Kibibyte, Mebibyte, Gibibyte und Tebibyte) angezeigt. Die Metrik wird alle 15 Minuten ausgegeben und ermöglicht es Ihnen, die gemessene Größe Ihres Dateisystems im Zeitverlauf zu überprüfen. Gemessene Größe zeigt die folgenden Informationen zur Speichergröße des Dateisystems an:

- Die Gesamtgröße ist die Größe (in Binärbyte) der im Dateisystem gespeicherten Daten, einschließlich aller Speicherklassen.
- Größe in Standard ist die Größe (in Binärbyte) der in der EFS-Standard-Speicherklasse gespeicherten Daten.
- Größe in IA ist die Größe (in Binärbyte) der in der „EFS Infrequent Access“-Speicherklasse gespeicherten Daten. Dateien, die kleiner als 128 KB sind, werden auf 128 KB aufgerundet.
- Größe in Archive ist die Größe (in Binärbyte) der in der „EFS Archive“-Speicherklasse gespeicherten Daten. Dateien, die kleiner als 128 KB sind, werden auf 128 KiB aufgerundet.

Sie können die Metrik `Storage bytes` auch auf der Registerkarte Überwachung auf der Seite mit den Dateisystemdetails in der Amazon-EFS-Konsole anzeigen. Weitere Informationen finden Sie unter [Zugreifen auf CloudWatch Metriken für Amazon EFS](#).

Mit dem AWS CLI

Sie können mithilfe der AWS CLI oder EFS-API anzeigen, wie viele Daten in jeder Speicherklasse Ihres Dateisystems gespeichert sind. Zeigen Sie Datenspeicherdetails an, indem Sie den `describe-file-systems`-CLI-Befehl aufrufen (die entsprechende API-Operation ist [DescribeFileSystems](#)).

```
$ aws efs describe-file-systems \
--region us-west-2 \
--profile adminuser
```

ValueInIA zeigt in der Antwort die zuletzt gemessene Größe in Byte in der „Infrequent Access“-Speicherklasse des Dateisystems an. ValueInStandard zeigt die zuletzt gemessene Größe in Byte in der Standard-Speicherklasse an. ValueInArchive zeigt die zuletzt gemessene Größe in Byte in der Archive-Speicherklasse an. Die Summe der drei Werte entspricht der Größe des gesamten Dateisystems, das in Value angezeigt wird.

```
{
  "FileSystems":[
    {
      "OwnerId":"251839141158",
      "CreationToken":"MyFileSystem1",
      "FileSystemId":"fs-47a2c22e",
      "PerformanceMode" : "generalPurpose",
      "CreationTime": 1403301078,
      "LifeCycleState":"created",
      "NumberOfMountTargets":1,
      "SizeInBytes":{
        "Value": 29313746702,
        "ValueInIA": 675432,
        "ValueInStandard": 29312741784,
        "ValueInArchive":329486
      },
      "ThroughputMode": "elastic"
    }
  ]
}
```

Informationen zu weiteren Möglichkeiten zum Anzeigen und Messen der Festplattennutzung finden Sie unter [Messung von EFS-Dateisystemobjekten](#).

Metriken mit Amazon überwachen CloudWatch

Sie können Dateisysteme mithilfe von Amazon überwachen CloudWatch, das Rohdaten aus Amazon EFS sammelt und zu lesbaren Metriken nahezu in Echtzeit verarbeitet. Diese Statistiken werden für einen Zeitraum von 15 Monaten aufgezeichnet, damit Sie einen besseren Überblick darüber erhalten, welche Leistung Ihre Webanwendung oder Ihr Service liefern.

Standardmäßig werden Amazon EFS-Metriken automatisch in Abständen von 1 Minute CloudWatch an gesendet, sofern nicht für einzelne Metriken etwas anderes angegeben ist. Die Amazon EFS-Konsole zeigt eine Reihe von Diagrammen an, die auf den Rohdaten von Amazon basieren CloudWatch. Je nach Ihren Anforderungen ziehen Sie es möglicherweise vor, Daten für Ihre Dateisysteme aus CloudWatch den Diagrammen in der Konsole abzurufen.

Weitere Informationen zu Amazon CloudWatch finden Sie unter [Was ist Amazon CloudWatch?](#) im CloudWatch Amazon-Benutzerhandbuch.

Amazon CloudWatch EFS-Metriken werden als Rohbytes gemeldet. Bytes werden nicht auf eine Dezimalzahl oder ein binäres Vielfaches der Einheit gerundet.

Themen

- [CloudWatch Metriken für Amazon EFS](#)
- [Zugreifen auf CloudWatch Metriken für Amazon EFS](#)
- [Verwenden von CloudWatch Metriken für Amazon EFS](#)
- [Verwenden von metrischer Mathematik mit CloudWatch Metriken](#)
- [Überwachung erfolgreicher und fehlgeschlagener Einhängeversuche](#)
- [CloudWatch Alarmer zur Überwachung von Amazon EFS erstellen](#)

CloudWatch Metriken für Amazon EFS

Amazon EFS-Metriken verwenden den EFS Namespace. Der AWS/EFS-Namespace enthält die folgenden Metriken. Alle Metriken außer für `TimeSinceLastSync` beziehen sich auf eine einzelne Dimension, `FileSystemId`. Die Dateisystem-ID kann der Amazon-EFS-Konsole entnommen werden und hat das Format `fs-abcdef0123456789a`.

TimeSinceLastSync

Zeigt die Zeit an, die seit der letzten erfolgreichen Synchronisierung mit dem Zieldateisystem in einer Replikationskonfiguration vergangen ist. Alle Änderungen an Daten im Quelldateisystem, die vor `TimeSinceLastSync` vorgenommen wurden, wurden erfolgreich in das Zieldateisystem repliziert. Alle Änderungen an der Quelle, die nach `TimeSinceLastSync` vorgenommen wurden, werden möglicherweise nicht vollständig repliziert.

Diese Metrik verwendet zwei Dimensionen:

- **FileSystemIdDimension** — ID des Quelldateisystems in der Replikationskonfiguration.
- **DestinationFileSystemIdDimension** — ID des Zieldateisystems in der Replikationskonfiguration.

Einheiten: Sekunden

Gültige Statistiken: Minimum, Maximum, Average

PercentIOLimit

Zeigt an, wie nahe ein Dateisystem daran ist, die I/O Grenze des Allzweck-Leistungsmodus zu erreichen.

Einheiten: Prozent

Gültige Statistiken: Minimum, Maximum, Average

BurstCreditBalance

Die Anzahl von Burst-Gutschriften, über die ein Dateisystem verfügt. Burst-Gutschriften berechtigen das Dateisystem, den Durchsatz für bestimmte Zeiträume über die Grundrate eines Dateisystems hinaus zu erhöhen.

Die Minimum-Statistik ist die kleinste Burst-Gutschrift für eine beliebige Minute des entsprechenden Zeitraums. Die Maximum-Statistik ist die größte Burst-Gutschrift für eine beliebige Minute des entsprechenden Zeitraums. Die Average-Statistik ist die durchschnittliche Burst-Gutschrift während des entsprechenden Zeitraums.

Einheiten: Byte

Gültige Statistiken: Minimum, Maximum, Average

PermittedThroughput

Die maximale Durchsatzmenge, die ein Dateisystem bewältigen kann.

- Bei Dateisystemen, die Elastic Throughput verwenden, spiegelt dieser Wert den maximalen Schreibdurchsatz des Dateisystems wider.
- Wenn bei Dateisystemen, die den bereitgestellten Durchsatz verwenden, die in der EFS-Standardspeicherklasse gespeicherte Datenmenge Ihrem Dateisystem ermöglicht, einen höheren Durchsatz zu erzielen, als Sie bereitgestellt haben, spiegelt diese Metrik den höheren Durchsatz und nicht die bereitgestellte Menge wider.

- Bei Dateisystemen mit Bursting-Durchsatz ist dieser Wert eine Funktion der Dateisystemgröße und `BurstCreditBalance`

Die Minimum-Statistik ist der kleinste Durchsatz für eine beliebige Minute des entsprechenden Zeitraums. Die Maximum-Statistik ist der höchste Durchsatz für eine beliebige Minute des entsprechenden Zeitraums. Die Average-Statistik ist der durchschnittliche Durchsatz, der während des entsprechenden Zeitraums erlaubt ist.

 Note

Lesevorgänge werden mit einem Drittel der Rate anderer Vorgänge gemessen.

Einheiten: Byte pro Sekunde

Gültige Statistiken: Minimum, Maximum, Average

MeteredIOBytes

Die Anzahl der gemessenen Byte für jeden Dateisystemvorgang, einschließlich Datenlese-, Datenschreib- und Metadatenoperationen, wobei Lesevorgänge je nach Durchsatzlimit abgezinst werden.

Sie können einen [CloudWatch metrischen mathematischen Ausdruck](#) erstellen, der mit verglichen `MeteredIOBytes` werden kann. `PermittedThroughput` Wenn diese Werte gleich sind, verbrauchen Sie den gesamten Durchsatz, der Ihrem Dateisystem zugewiesen ist. In diesem Fall könnten Sie erwägen, den Durchsatzmodus des Dateisystems zu ändern, um einen höheren Durchsatz zu erzielen.

Die Sum-Statistik ist die Gesamtzahl von gemessenen Byte, die mit allen Dateisystemoperationen verknüpft sind. Die Minimum-Statistik ist die Größe der kleinsten Operation während des jeweiligen Zeitraums. Die Maximum-Statistik ist die Größe der größten Operation während des jeweiligen Zeitraums. Die Average-Statistik ist die durchschnittliche Größe einer Operation während des jeweiligen Zeitraums. Die `SampleCount`-Statistik stellt eine Zählung aller Leseoperationen zur Verfügung.

Einheiten:

- Byte für die Statistiken Minimum, Maximum, Average und Sum statistics.
- Anzahl für `SampleCount`.

Gültige Statistiken: Minimum, Maximum, Average, Sum, SampleCount

Insgesamt IOBytes

Die tatsächliche Anzahl von Byte für jeden Dateisystemvorgang, der von Amazon EFS verarbeitet wird, ohne Leserabatte. Diese Zahl kann von der tatsächlich von Ihren Anwendungen angeforderten Menge abweichen, da sie Mindestwerte enthält. Diese Zahl kann auch höher sein als die unter angegebenen Zahlen. PermittedThroughput

Datenoperationen werden mit 32 KiB und andere Operationen mit 4 KiB gemessen. Nach dem Mindestwert werden alle Operationen pro KiB gemessen.

Die Sum-Statistik ist die Gesamtzahl von Byte, die mit allen Dateisystemoperationen verknüpft sind. Die Minimum-Statistik ist die Größe der kleinsten Operation während des jeweiligen Zeitraums. Die Maximum-Statistik ist die Größe der größten Operation während des jeweiligen Zeitraums. Die Average-Statistik ist die durchschnittliche Größe einer Operation während des jeweiligen Zeitraums. Die SampleCount-Statistik stellt eine Zählung aller Leseoperationen zur Verfügung.

Note

Zum Berechnen der durchschnittlichen Operationen pro Sekunde für einen Zeitraum dividieren Sie die SampleCount-Statistik durch die Anzahl von Sekunden in dem Zeitraum. Zum Berechnen des durchschnittlichen Durchsatzes (Byte pro Sekunde) für einen Zeitraum dividieren Sie die Sum-Statistik durch die Anzahl von Sekunden in dem Zeitraum.

Einheiten:

- Byte für die Statistiken Minimum, Maximum, Average und Sum statistics.
- Anzahl für SampleCount.

Gültige Statistiken: Minimum, Maximum, Average, Sum, SampleCount

DataReadIOBytes

Die tatsächliche Anzahl von Byte für jeden Lesevorgang im Dateisystem.

Die Sum-Statistik ist die Gesamtzahl von Byte, die mit den Leseoperationen verknüpft sind. Die Minimum-Statistik ist die Größe der kleinsten Leseoperation während des jeweiligen Zeitraums.

Die Maximum-Statistik ist die Größe der größten Leseoperation während des jeweiligen Zeitraums. Die Average-Statistik ist die durchschnittliche Größe der Leseoperationen während des jeweiligen Zeitraums. Die SampleCount-Statistik stellt eine Anzahl von Leseoperationen zur Verfügung.

Einheiten:

- Byte für Minimum, Maximum, Average und Sum.
- Anzahl für SampleCount.

Gültige Statistiken: Minimum, Maximum, Average, Sum, SampleCount

DataWriteIOBytes

Die tatsächliche Anzahl von Byte für jeden Schreibvorgang im Dateisystem.

Die Sum-Statistik ist die Gesamtzahl von Byte, die mit den Schreiboperationen verknüpft sind. Die Minimum-Statistik ist die Größe der kleinsten Schreiboperation während des jeweiligen Zeitraums. Die Maximum-Statistik ist die Größe der größten Schreiboperation während des jeweiligen Zeitraums. Die Average-Statistik ist die durchschnittliche Größe der Schreiboperationen während des jeweiligen Zeitraums. Die SampleCount-Statistik stellt eine Anzahl von Schreiboperationen zur Verfügung.

Einheiten:

- Byte ist die Einheit für die Statistiken Minimum, Maximum, Average und Sum.
- Anzahl für SampleCount.

Gültige Statistiken: Minimum, Maximum, Average, Sum, SampleCount

MetadataIOBytes

Die tatsächliche Anzahl von Byte für jeden Metadatenvorgang.

Die Sum-Statistik ist die Gesamtzahl von Byte, die mit den Metadatenoperationen verknüpft sind. Die Minimum-Statistik ist die Größe der kleinsten Metadatenoperation während des jeweiligen Zeitraums. Die Maximum-Statistik ist die Größe der größten Metadatenoperation während des jeweiligen Zeitraums. Die Average-Statistik ist die Größe der durchschnittlichen Metadatenoperation während des jeweiligen Zeitraums. Die SampleCount-Statistik stellt eine Anzahl von Metadatenoperationen zur Verfügung.

Einheiten:

- Byte ist die Einheit für die Statistiken Minimum, Maximum, Average und Sum.

- Anzahl für SampleCount.

Gültige Statistiken: Minimum, Maximum, Average, Sum, SampleCount

MetadataReadIOBytes

Die tatsächliche Anzahl von Byte für jeden Metadaten-Lesevorgang.

Die Sum Statistik gibt die Gesamtzahl der Byte an, die mit Leseoperationen für Metadaten verknüpft sind. Die Minimum Statistik gibt die Größe des kleinsten Metadatenlesevorgangs während des Zeitraums an. Die Maximum Statistik gibt die Größe des größten Metadatenlesevorgangs in dem Zeitraum an. Die Average Statistik gibt die durchschnittliche Größe der Leseoperationen für Metadaten in diesem Zeitraum an. Die SampleCount Statistik gibt die Anzahl der Leseoperationen für Metadaten an.

Einheiten:

- Byte ist die Einheit für die Statistiken Minimum, Maximum, Average und Sum.
- Anzahl für SampleCount.

Gültige Statistiken: Minimum, Maximum, Average, Sum, SampleCount

MetadataWriteIOBytes

Die tatsächliche Anzahl von Byte für jeden Metadaten-Schreibvorgang.

Die Sum Statistik gibt die Gesamtzahl der Byte an, die mit Schreibvorgängen für Metadaten verknüpft sind. Die Minimum Statistik gibt die Größe des kleinsten Metadaten-Schreibvorgangs während des Zeitraums an. Die Maximum Statistik gibt die Größe des größten Schreibvorgangs für Metadaten in dem Zeitraum an. Die Average Statistik gibt die durchschnittliche Größe der Schreibvorgänge für Metadaten in diesem Zeitraum an. Die SampleCount Statistik gibt die Anzahl der Schreibvorgänge für Metadaten an.

Einheiten:

- Byte ist die Einheit für die Statistiken Minimum, Maximum, Average und Sum.
- Anzahl für SampleCount.

Gültige Statistiken: Minimum, Maximum, Average, Sum, SampleCount

ClientConnections

Die Anzahl von Client-Verbindungen mit einem Dateisystem. Bei Verwendung eines Standard-Clients ist eine Verbindung je aufgespielter Amazon-EC2-Instance vorhanden.

 Note

Zum Berechnen der durchschnittlichen `ClientConnections` für Zeiträume über eine Minute dividieren Sie die Sum-Statistik durch die Anzahl von Minuten in dem Zeitraum.

Einheiten: Anzahl von Client-Verbindungen

Gültige Statistiken: Sum

StorageBytes

Die Größe des Dateisystems in Byte, einschließlich der Datenmenge, die in den EFS-Speicherklassen gespeichert ist. Diese Metrik wird CloudWatch alle 15 Minuten ausgegeben.

Die StorageBytes Metrik hat die folgenden Dimensionen:

- **Total** ist die gemessene Größe (in Byte) der im Dateisystem gespeicherten Daten in allen Speicherklassen. Für die Speicherklassen EFS Infrequent Access (IA) und EFS Archive werden Dateien, die kleiner als 128 KB sind, auf 128 KB gerundet.
- **Standard** ist die gemessene Größe (in Byte) der in der EFS-Standardspeicherklasse gespeicherten Daten.
- **IA** ist die tatsächliche Größe (in Byte) der in der EFS-Speicherklasse für seltenen Zugriff gespeicherten Daten.
- **IASizeOverhead** ist die Differenz (in Byte) zwischen der tatsächlichen Größe der Daten in der EFS-Speicherklasse für seltenen Zugriff (in der IA Dimension angegeben) und der gemessenen Größe der Speicherklasse, nachdem kleine Dateien auf 128 KB gerundet wurden.
- **Archive** ist die tatsächliche Größe (in Byte) der in der EFS-Archivspeicherklasse gespeicherten Daten.
- **ArchiveSizeOverhead** ist die Differenz (in Byte) zwischen der tatsächlichen Größe der Daten in der EFS-Archivspeicherklasse (in der Archive Dimension angegeben) und der gemessenen Größe der Speicherklasse, nachdem kleine Dateien auf 128 KB gerundet wurden.

Einheiten: Byte

Gültige Statistiken: Minimum, Maximum, Average

 Note

StorageBytes wird auf der Seite mit den Dateisystemmetriken der Amazon-EFS-Konsole angezeigt und verwendet 1024 Basiseinheiten (Kibibyte, Mebibyte, Gibibyte und Tebibyte).

Zugreifen auf CloudWatch Metriken für Amazon EFS

Sie können Amazon EFS-Metriken für CloudWatch auf verschiedene Arten anzeigen:

- In der Amazon-EFS-Konsole
- In der CloudWatch Konsole
- Verwenden der CloudWatch CLI
- Verwenden der CloudWatch API

So zeigen Sie CloudWatch Metriken und Alarme an (Amazon EFS-Konsole)

1. Melden Sie sich bei der an AWS-Managementkonsole und öffnen Sie die Amazon EFS-Konsole unter <https://console.aws.amazon.com/efs/>.
2. Wählen Sie Dateisysteme aus.
3. Wählen Sie das Dateisystem aus, für das Sie CloudWatch Metriken anzeigen möchten.
4. Wählen Sie Überwachung aus, um die Seite mit den Dateisystem-Metriken anzuzeigen.

Auf der Seite Dateisystem-Metriken wird ein Standardsatz von CloudWatch Metriken für das Dateisystem angezeigt. Alle CloudWatch Alarme, die Sie konfiguriert haben, werden ebenfalls mit diesen Metriken angezeigt. Bei Dateisystemen, die den Modus Max I/O Performance verwenden, beinhaltet der Standardsatz von Metriken den Burst-Guthaben-Saldo anstelle des I/O-Limits in Prozent. Sie können die Standardeinstellungen im Dialogfeld „Metrik-Einstellungen“ überschreiben, auf das Sie durch Öffnen der Einstellungen zugreifen können.

 Note

Die Metrik Durchsatzauslastung (%) ist keine CloudWatch Metrik; sie wird mithilfe CloudWatch metrischer Mathematik abgeleitet.

5. Sie können die Art und Weise, wie Metriken und Alarme angezeigt werden, mithilfe der Steuerelemente auf der Seite Dateisystem-Metriken wie folgt anpassen.
 - Schalten Sie im Anzeigemodus zwischen Zeitreihen und Einzelwert um.
 - Zeigt alle für das Dateisystem konfigurierten CloudWatch Alarme an oder blendet sie aus.
 - Wählen Sie Mehr anzeigen in CloudWatch, um die Metriken unter anzuzeigen CloudWatch.
 - Wählen Sie Zum Dashboard hinzufügen, um Ihr CloudWatch Dashboard zu öffnen und die angezeigten Metriken hinzuzufügen.
 - Passen Sie das angezeigte Zeitfenster für die Metrik von 1 Stunde bis 1 Woche an.

Um CloudWatch Metriken und Alarme anzuzeigen (CloudWatch Konsole)

1. Öffnen Sie die CloudWatch Konsole unter <https://console.aws.amazon.com/cloudwatch/>.
2. Wählen Sie im Navigationsbereich Metriken aus.
3. Wählen Sie den EFS-Namespace aus.
4. (Optional) Geben Sie den Namen einer Metrik in das Suchfeld ein, um sie anzuzeigen.
5. (Optional) Um nach Dimensionen zu filtern, wählen Sie FileSystemId.

Um auf Metriken von der zuzugreifen AWS CLI

- Verwenden Sie den Befehl [list-metrics](#) mit dem --namespace "AWS/EFS"-Namespace. Weitere Informationen finden Sie in der [AWS CLI -Befehlsreferenz](#).

Um über die CloudWatch API auf Metriken zuzugreifen

- Rufen Sie die Seite [GetMetricStatistics](#) auf. Weitere Informationen finden Sie in der [Amazon CloudWatch API-Referenz](#).

Verwenden von CloudWatch Metriken für Amazon EFS

Die von Amazon EFS gemeldeten Metriken bieten Informationen, die Sie auf unterschiedliche Weise analysieren können. In der folgenden Liste finden Sie einige häufige Verwendungszwecke für die Metriken. Es handelt sich dabei um Vorschläge für den Einstieg und nicht um eine umfassende Liste.

Wie gehe ich vor?	Relevante Metriken
Wie kann ich meinen Durchsatz bestimmen?	Sie können die tägliche Sum Statistik der <code>TotalIOBytes</code> -Metrik überwachen, um Ihren Durchsatz zu sehen.
Wie kann ich die Gesamtmenge der übertragenen Daten ermitteln?	Sie können die tägliche Sum Statistik von <code>überwachenMeteredIOBytes</code> , um Ihre gesamten übertragenen Daten zu sehen.
Wie kann ich die Anzahl der Amazon-EC2-Instances, die mit einem Dateisystem verbunden sind, nachverfolgen?	Sie können die Sum-Statistik der <code>ClientConnections</code> -Metrik überwachen. Zum Berechnen der durchschnittlichen <code>ClientConnections</code> für Perioden über eine Minute dividieren Sie die Summe durch die Anzahl der Minuten in der Periode.
Wie kann ich mein Spitzenkreditsaldo sehen?	Sie können Ihr Saldo sehen, indem Sie die <code>BurstCreditBalance</code> -Metrik für Ihr Dateisystem überwachen. Weitere Informationen zu Spitzenwerten und Spitzenguthaben finden Sie unter Bursting-Durchsatz .

Überwachung der Durchsatzleistung

Die CloudWatch Metriken für die Durchsatzüberwachung — `TotalIOBytes`, `ReadIOBytes`, `WriteIOBytes`, und `MetadataIOBytes` — stellen den tatsächlichen Durchsatz dar, den Sie in Ihrem Dateisystem erzielen. Die Metrik `MeteredIOBytes` stellt die Berechnung des gemessenen Gesamtdurchsatzes dar, den Sie erzielen. Sie können das Diagramm zur Durchsatzauslastung (%) im Bereich Überwachung der Amazon-EFS-Konsole verwenden, um Ihre Durchsatzauslastung zu überwachen. Wenn Sie benutzerdefinierte CloudWatch Dashboards oder ein anderes Überwachungstool verwenden, können Sie einen [CloudWatch metrischen mathematischen Ausdruck](#) erstellen, der `MeteredIOBytes` sich `PermittedThroughput` mit vergleichen lässt.

`PermittedThroughput` misst die Menge des zulässigen Durchsatzes für das Dateisystem. Dieser Wert basiert auf einer der folgenden Methoden:

- Bei Dateisystemen mit Elastic Throughput spiegelt dieser Wert den maximalen Schreibdurchsatz des Dateisystems wider.
- Wenn bei Dateisystemen, die den bereitgestellten Durchsatz verwenden, die in der EFS-Standard-speicherklasse gespeicherte Datenmenge Ihrem Dateisystem ermöglicht, einen höheren Durchsatz zu erzielen, als Sie bereitgestellt haben, spiegelt diese Metrik den höheren Durchsatz und nicht die bereitgestellte Menge wider.
- Bei Dateisystemen, die den Bursting-Durchsatz verwenden, ist dieser Wert eine Funktion der Dateisystemgröße und. `BurstCreditBalance` Überwachen Sie `BurstCreditBalance`, um sicherzustellen, dass Ihr Dateisystem mit seiner Burst-Rate und nicht mit seiner Basisrate arbeitet. Wenn der Saldo konstant bei oder nahe Null liegt, sollten Sie erwägen, zu Elastic Throughput oder Provisioned Throughput zu wechseln, um zusätzlichen Durchsatz zu erzielen.

Wenn die Werte für `MeteredIOBytes` und `PermittedThroughput` gleich sind, verbraucht Ihr Dateisystem den gesamten verfügbaren Durchsatz. Für Dateisysteme, die den bereitgestellten Durchsatz verwenden, können Sie zusätzlichen Durchsatz bereitstellen.

Verwenden von metrischer Mathematik mit CloudWatch Metriken

Mit metrischer Mathematik können Sie mehrere CloudWatch Amazon-Metriken abfragen und mathematische Ausdrücke verwenden, um neue Zeitreihen auf der Grundlage dieser Metriken zu erstellen. Sie können die resultierenden Zeitreihen in der CloudWatch Konsole visualisieren und sie zu Dashboards hinzufügen. Beispielsweise können Sie Amazon-EFS-Metriken verwenden, um die Beispielanzahl von `DataRead-Operationen` geteilt durch 60 zu berechnen. Das Ergebnis ist die durchschnittliche Anzahl der Lesevorgänge pro Sekunde in Ihrem Dateisystem für ein bestimmtes 1-Minuten-Intervall. Weitere Informationen zur metrischen Mathematik finden Sie unter [Verwenden von mathematischen Ausdrücken mit CloudWatch Metriken](#) im CloudWatch Amazon-Benutzerhandbuch.

Im Folgenden finden Sie einige nützliche Ausdrücke für Metrikberechnungen mit Amazon EFS.

Themen

- [Metrische Mathematik: Durchsatz in MiBps](#)
- [Metrikberechnung: Durchsatz in Prozent](#)
- [Metrikberechnung: Prozentsatz der zulässigen Auslastung des Durchsatzes](#)
- [Metrikberechnung: Durchsatz in IOPS](#)
- [Metrikberechnung: Prozentsatz der IOPS](#)
- [Metrische Mathematik: I/O Durchschnittsgröße in KiB](#)

- [Verwenden von metrischer Mathematik mithilfe einer CloudFormation Vorlage für Amazon EFS](#)

Metrische Mathematik: Durchsatz in MiBps

Um den durchschnittlichen Durchsatz (in MiBps) für einen Zeitraum zu berechnen, wählen Sie zunächst eine Summenstatistik (`DataReadIOBytes`, `DataWriteIOBytes`, `MetadataIOBytes`, oder `TotalIOBytes`). Konvertieren Sie den Wert anschließend in MiB und teilen Sie diese Zahl durch die Anzahl der Sekunden in dem Intervall.

Angenommen, Ihre Beispiellogik sieht wie folgt aus: (Summe von `TotalIOBytes` ÷ 1.048.576 (zu konvertieren in MiB)) ÷ Sekunden im Intervall

Dann lauten Ihre CloudWatch Metrikinformationen wie folgt.

ID (ID)	Verwendbare Metriken	Statistik	Zeitraum
m1	• <code>DataReadIOBytes</code> • <code>DataWriteIOBytes</code> • <code>MetadataIOBytes</code> • <code>TotalIOBytes</code>	sum	1 Minute

ID und Ausdruck Ihrer Metrikberechnung lauten wie folgt.

ID (ID)	Expression
e1	<code>(m1/1048576)/PERIOD(m1)</code>

Metrikberechnung: Durchsatz in Prozent

Dieser metrische mathematische Ausdruck berechnet den Prozentsatz des Gesamtdurchsatzes, der für die verschiedenen I/O Typen verwendet wird, z. B. den Prozentsatz des Gesamtdurchsatzes,

der durch Leseanforderungen gesteuert wird. Zum Berechnen des Gesamtdurchsatzes der verschiedenen E/A-Typen (`DataReadIOBytes`, `DataWriteIOBytes` oder `MetadataIOBytes`) für einen Zeitbereich multiplizieren Sie zunächst die entsprechende Summenstatistik mit 100. Teilen Sie dann das Ergebnis durch die Summenstatistik von `TotalIOBytes` für das gleiche Intervall.

Angenommen, Ihre Beispiellogik sieht wie folgt aus: (Summe von `DataReadIOBytes` x 100 (zu konvertieren in Prozent)) ÷ Summe von `TotalIOBytes`

Dann lauten Ihre CloudWatch Metrikinformationen wie folgt.

ID (ID)	Verwendbare Metrik oder Metriken	Statistik	Zeitraum
m1	• <code>TotalIOBytes</code>	sum	1 Minute
m2	• <code>DataReadIOBytes</code>	sum	1 Minute

ID und Ausdruck Ihrer Metrikberechnung lauten wie folgt.

ID (ID)	Expression
e1	$(m2 * 100) / m1$

Metrikberechnung: Prozentsatz der zulässigen Auslastung des Durchsatzes

Um den Prozentsatz der zulässigen Durchsatzauslastung (`MeteredIOBytes`) für einen bestimmten Zeitraum zu berechnen, multiplizieren Sie den Durchsatz zunächst MiBps mit 100. Teilen Sie dann das Ergebnis durch die Durchschnittsstatistik von `PermittedThroughput` umgerechnet in MiB für denselben Zeitraum.

Angenommen, Ihre Beispiellogik lautet wie folgt: (metrischer mathematischer Ausdruck für den Durchsatz in MiBps x 100 (zur Umrechnung in Prozent)) ÷ (Summe von `PermittedThroughput` ÷ 1.048.576 (zur Umrechnung von Byte in MiB))

Dann lauten Ihre CloudWatch metrischen Informationen wie folgt.

ID (ID)	Verwendbare Metrik oder Metriken	Statistik	Zeitraum
m1	MeteredIOBytes	sum	1 Minute
m2	Permitted Throughput	Durchschnitt	1 Minute

ID und Ausdruck Ihrer Metrikberechnung lauten wie folgt.

ID (ID)	Expression
e1	$(m1/1048576)/PERIOD(m1)$
e2	$m2/1048576$
e3	$((e1)*100)/(e2)$

Metrikberechnung: Durchsatz in IOPS

Zum Berechnen der durchschnittlichen Operationen pro Sekunde (IOPS) für ein Intervall dividieren Sie die Beispiellanzahlstatistik (DataReadIOBytes, DataWriteIOBytes, MetadataIOBytes oder TotalIOBytes) durch die Anzahl von Sekunden in dem Intervall.

Angenommen, Ihre Beispiellogik sieht wie folgt aus: Beispiellanzahl von DataWriteIOBytes ÷ Sekunden im Intervall

Dann lauten Ihre CloudWatch metrischen Informationen wie folgt.

ID (ID)	Verwendbare Metriken	Statistik	Zeitraum
m1	- DataReadIOBytes - DataWriteIOBytes	Beispiellanzahl	1 Minute

ID (ID)	Verwendbare Metriken	Statistik	Zeitraum
	• <code>MetadataIOBytes</code> • <code>TotalIOBytes</code>		

ID und Ausdruck Ihrer Metrikberechnung lauten wie folgt.

ID (ID)	Expression
e1	<code>m1/PERIOD(m1)</code>

Metrikberechnung: Prozentsatz der IOPS

Um den Prozentsatz der IOPS pro Sekunde der verschiedenen I/O Typen (`DataReadIOBytes`, `MetadataIOBytes`, `DataWriteIOBytes`) für einen bestimmten Zeitraum zu berechnen, multiplizieren Sie zunächst die entsprechende Statistik zur Stichprobenanzahl mit 100. Teilen Sie dann diesen Wert durch die Beispielanzahlstatistik von `TotalIOBytes` für das gleiche Intervall.

Angenommen, Ihre Beispiellogik sieht wie folgt aus: $(\text{Beispielanzahl von MetadataIOBytes} \times 100 \text{ (zu konvertieren in Prozent)}) \div \text{Beispielanzahl von TotalIOBytes}$

Dann lauten Ihre CloudWatch Metrikinformationen wie folgt.

ID (ID)	Verwendbare Metriken	Statistik	Zeitraum
m1	• <code>TotalIOBytes</code>	Beispielanzahl	1 Minute
m2	• <code>DataReadIOBytes</code> • <code>DataWriteIOBytes</code> • <code>MetadataIOBytes</code>	Beispielanzahl	1 Minute

ID und Ausdruck Ihrer Metrikberechnung lauten wie folgt.

ID (ID)	Expression
e1	$(m2 \times 100) / m1$

Metrische Mathematik: I/O Durchschnittsgröße in KiB

Um die durchschnittliche I/O Größe (in KiB) für einen Zeitraum zu berechnen, dividieren Sie die entsprechende Summenstatistik für die `MetadataIOBytes` Metrik `DataReadIOBytes``DataWriteIOBytes`, oder durch dieselbe Stichprobenzahlstatistik dieser Metrik.

Angenommen, Ihre Beispiellogik sieht wie folgt aus: $(\text{Summe von DataReadIOBytes} \div 1.024 \text{ (zu konvertieren in KiB)}) \div \text{Beispielanzahl von DataReadIOBytes}$

Dann lauten Ihre CloudWatch metrischen Informationen wie folgt.

ID (ID)	Verwendbare Metriken	Statistik	Zeitraum
m1	<code>DataReadIOBytes</code> <code>DataWriteIOBytes</code> <code>MetadataIOBytes</code>	sum	1 Minute
m2	<code>DataReadIOBytes</code> <code>DataWriteIOBytes</code> <code>MetadataIOBytes</code>	Beispielanzahl	1 Minute

ID und Ausdruck Ihrer Metrikberechnung lauten wie folgt.

ID (ID)	Expression
e1	(m1/1024)/m2

Verwenden von metrischer Mathematik mithilfe einer CloudFormation Vorlage für Amazon EFS

Sie können metrische mathematische Ausdrücke auch mithilfe von CloudFormation Vorlagen erstellen. Eine solche Vorlage steht Ihnen in den [Amazon EFS-Tutorials](#) zum Herunterladen und Anpassen zur Verfügung GitHub. Weitere Informationen zur Verwendung von CloudFormation Vorlagen finden Sie unter [Arbeiten mit CloudFormation Vorlagen](#) im AWS CloudFormation Benutzerhandbuch.

Überwachung erfolgreicher und fehlgeschlagener Einhängeversuche

Sie können Amazon CloudWatch Logs verwenden, um den Erfolg oder Misserfolg von Bereitstellungsversuchen für Ihre EFS-Dateisysteme remote zu überwachen und zu melden, ohne sich bei den Clients anmelden zu müssen. Gehen Sie wie folgt vor, um Ihre EC2-Instance so zu konfigurieren, dass sie CloudWatch Logs verwendet, um den Erfolg oder Misserfolg ihrer Dateisystem-Mount-Versuche zu überwachen.

So aktivieren Sie die Benachrichtigung über erfolgreiche oder fehlgeschlagene Bereitstellungsversuche in den Protokollen CloudWatch

1. Installieren Sie `amazon-efs-utils` auf der EC2-Instance, die das Dateisystem mountet. Für weitere Informationen siehe [Automatisches Installieren oder Aktualisieren des Amazon EFS-Clients mit AWS Systems Manager](#) oder [Manuelles Installieren des Amazon-EFS-Clients](#).
2. Installieren Sie `botocore` auf der EC2-Instance, auf der das Dateisystem gemountet werden soll. Weitere Informationen finden Sie unter [Installation und Aktualisierung botocore](#).
3. Aktivieren Sie die CloudWatch Protokollfunktion in `amazon-efs-utils`. Bei der Installation und Konfiguration erfolgt `amazon-efs-utils` die CloudWatch Protokollierung automatisch für Sie. AWS Systems Manager Wenn Sie das `amazon-efs-utils`-Paket manuell installieren, müssen Sie die Konfigurationsdatei `/etc/amazon/efs/efs-utils.conf` manuell aktualisieren, indem Sie die Kommentierung der Zeile `# enabled = true` im Abschnitt `cloudwatch-log` aufheben. Verwenden Sie einen der folgenden Befehle, um CloudWatch Logs manuell zu aktivieren.

Für Linux-Instances:

```
sudo sed -i -e '/\[cloudwatch-log\]/{N;s/# enabled = true/enabled = true/}' /etc/amazon/efs/efs-utils.conf
```

Für MacOS-Instances:

```
EFS_UTILS_VERSION= efs-utils-version  
sudo sed -i -e '/\[cloudwatch-log\]/{N;s/# enabled = true/enabled = true/;}' /usr/local/Cellar/amazon-efs-utils/${EFS_UTILS_VERSION}/libexec/etc/amazon/efs/efs-utils.conf
```

Für Mac2-Instances:

```
EFS_UTILS_VERSION= efs-utils-version  
sudo sed -i -e '/\[cloudwatch-log\]/{N;s/# enabled = true/enabled = true/;}' /opt/homebrew/Cellar/amazon-efs-utils/${EFS_UTILS_VERSION}/libexec/etc/amazon/efs/efs-utils.conf
```

- Optional können Sie die Namen der CloudWatch Protokollgruppen konfigurieren und die Aufbewahrungsdauer der Protokolle in der `efs-utils.conf` Datei festlegen. Wenn Sie CloudWatch für jedes eingehängte Dateisystem separate Protokollgruppen einrichten möchten, fügen Sie `{fs_id}` am Ende des `log_group_name` Felds in der `efs-utils.conf` Datei Folgendes hinzu:

```
[cloudwatch-log]  
log_group_name = /aws/efs/utils/{fs_id}
```

- Hängen Sie die `AmazonElasticFileSystemsUtils` AWS verwaltete Richtlinie an die IAM-Rolle an, die Sie der EC2-Instance zugewiesen haben, oder an die auf Ihrer Instance konfigurierten AWS Anmeldeinformationen. Sie können dazu Systems Manager verwenden. Weitere Informationen finden Sie unter [Schritt 1: Konfigurieren Sie ein \(IAM\)-Instance-Profil mit den erforderlichen Berechtigungen](#).

Im Folgenden finden Sie Beispiele für Protokolleinträge zum Status eines Mount-Versuchs:

```
Successfully mounted fs-12345678.efs.us-east-1.amazonaws.com at /home/ec2-user/efs  
Mount failed, Failed to resolve "fs-01234567.efs.us-east-1.amazonaws.com"
```

Um den Mount-Status in Logs anzuzeigen CloudWatch

1. Öffnen Sie die CloudWatch Konsole unter <https://console.aws.amazon.com/cloudwatch/>.
2. Wählen Sie im linken Navigationsbereich Protokollgruppen aus.
3. Wählen Sie die Gruppe/aws/efs/utlslog aus. Sie sehen einen Protokollstream für jede Kombination aus Amazon-EC2-Instance und EFS-Dateisystem.
4. Wählen Sie einen Protokollstream aus, um bestimmte Protokollereignisse wie den Status eines erfolgreichen Mount-Versuchs oder den Status eines Fehlers anzuzeigen.

CloudWatch Alarme zur Überwachung von Amazon EFS erstellen

Sie können einen CloudWatch Alarm erstellen, der eine Amazon SNS SNS-Nachricht sendet, wenn sich der Status des Alarms ändert. Ein Alarm überwacht eine Metrik über einen bestimmten, von Ihnen definierten Zeitraum. Der Alarm führt eine oder mehrere Aktionen durch, die vom Wert der Metrik im Vergleich zu einem gegebenen Schwellenwert in einer Reihe von Zeiträumen abhängt. Die Aktion ist eine Benachrichtigung, die an ein Amazon-SNS-Thema oder eine Auto Scaling-Richtlinie gesendet wird.

Bei Alarmen werden nur Aktionen für anhaltende Statusänderungen ausgelöst. CloudWatch Alarme rufen nicht nur Aktionen auf, weil sie sich in einem bestimmten Zustand befinden. Der Status muss sich geändert haben und für eine bestimmte Anzahl von Zeiträumen beibehalten worden sein.

Eine wichtige Verwendung von CloudWatch Alarmen für Amazon EFS besteht darin, die Verschlüsselung im Ruhezustand für Ihr Dateisystem zu erzwingen. Sie können bei dessen Erstellung die Verschlüsselung im Ruhezustand für ein Amazon-EFS-Dateisystem aktivieren. Um encryption-at-rest Datenrichtlinien für Amazon EFS-Dateisysteme durchzusetzen, können Sie Amazon CloudWatch verwenden, AWS CloudTrail um die Erstellung eines Dateisystems zu erkennen und zu überprüfen, ob die Verschlüsselung im Ruhezustand aktiviert ist.

Note

Derzeit können Sie keine Verschlüsselung während der Übertragung erzwingen.

Im folgenden Verfahren wird beschrieben, wie Sie Alarme für Amazon EFS erstellen.

Verwenden der Konsole

Um Alarme über die CloudWatch Konsole einzustellen

1. Melden Sie sich bei der an AWS-Managementkonsole und öffnen Sie die CloudWatch Konsole unter <https://console.aws.amazon.com/cloudwatch/>.
2. Wählen Sie Alarm erstellen aus. Dadurch wird der Assistent zum Erstellen von Alarmen gestartet.
3. Wählen Sie EFS-Metriken und durchblättern Sie die Amazon-EFS-Metriken, bis Sie die Metrik finden, auf die Sie einen Alarm setzen möchten. Um in diesem Dialogfeld nur die Amazon-EFS-Metriken anzuzeigen, suchen Sie nach der Dateisystem-ID Ihres Dateisystems. Wählen Sie die Metrik aus, um einen Alarm zu erstellen, und klicken Sie dann auf Weiter.
4. Geben Sie unter Name, Beschreibung und Whenever (Wenn) die Werte für die Metrik ein.
5. Wenn Sie Ihnen eine E-Mail senden CloudWatch möchten, wenn der Alarmstatus erreicht ist, wählen Sie im Feld Wann immer dieser Alarm: die Option Status ist ALARM. Wählen Sie im Feld Send notification to: (Benachrichtigung senden an:) ein SNS-Thema aus. Wenn Sie Create topic auswählen, können Sie den Namen und die E-Mail Adressen für eine neue E-Mail-Abonnementliste einrichten. Diese Liste wird gespeichert und erscheint für künftige Alarme in der Liste.

Note

Wenn Sie Create topic (Thema erstellen) verwenden, um ein neues Amazon-SNS-Thema einzurichten, müssen die E-Mail Adressen überprüft werden, bevor die Empfänger Benachrichtigungen erhalten. E-Mail Nachrichten werden nur gesendet, wenn der Alarm in einen Alarmzustand wechselt. Wenn es zu dieser Änderung des Alarmzustands kommt, bevor die E-Mail Adressen überprüft wurden, erhalten die Empfänger keine Benachrichtigung.

6. An diesem Punkt finden Sie im Bereich Alarm-Vorschau eine Vorschau des Alarms, den Sie gerade erstellen. Wählen Sie Alarm erstellen aus.

Verwenden Sie den AWS CLI

Um einen Alarm einzustellen mit dem AWS CLI

- Rufen Sie die Seite [put-metric-alarm](#) auf. Weitere Informationen finden Sie in der [AWS CLI - Befehlsreferenz](#).

Mithilfe der CloudWatch API

Um mithilfe der CloudWatch API einen Alarm einzustellen

- Rufen Sie die Seite [PutMetricAlarm](#) auf. Weitere Informationen finden Sie in der [Amazon CloudWatch API-Referenz](#).

Protokollieren von Amazon EFS-API-Aufrufen mit AWS CloudTrail

Amazon EFS ist in einen Service integriert AWS CloudTrail, der eine Aufzeichnung der Aktionen bereitstellt, die von einem Benutzer, einer Rolle oder einem AWS Service in Amazon EFS ausgeführt wurden. CloudTrail erfasst alle API-Aufrufe für Amazon EFS als Ereignisse, einschließlich Aufrufe von der Amazon EFS-Konsole und von Codeaufrufen an Amazon EFS-API-Operationen.

Wenn Sie einen Trail erstellen, können Sie die kontinuierliche Übermittlung von CloudTrail Ereignissen an einen Amazon S3 S3-Bucket aktivieren, einschließlich Ereignissen für Amazon EFS. Wenn Sie keinen Trail konfigurieren, können Sie die neuesten Ereignisse trotzdem in der CloudTrail Konsole im Ereignisverlauf anzeigen. Anhand der von gesammelten Informationen können Sie die Anfrage CloudTrail, die an Amazon EFS gestellt wurde, die IP-Adresse, von der aus die Anfrage gestellt wurde, wer die Anfrage gestellt hat, wann sie gestellt wurde, und weitere Details ermitteln.

Weitere Informationen finden Sie unter [Was ist AWS CloudTrail?](#) im AWS CloudTrail Benutzerhandbuch.

Amazon EFS-Informationen in CloudTrail

CloudTrail ist auf Ihrem aktiviert AWS-Konto , wenn Sie das Konto erstellen. Wenn eine Aktivität in Amazon EFS auftritt, wird diese Aktivität zusammen mit anderen CloudTrail AWS Serviceereignissen im Ereignisverlauf in einem Ereignis aufgezeichnet. Sie können aktuelle Ereignisse in Ihrem anzeigen, suchen und herunterladen AWS-Konto. Weitere Informationen finden Sie unter [Arbeiten mit dem CloudTrail Ereignisverlauf](#).

Für eine fortlaufende Aufzeichnung der Ereignisse in Ihrem AWS-Konto, einschließlich Ereignissen für Amazon EFS, erstellen Sie einen Trail. Ein Trail ermöglicht CloudTrail die Übermittlung von Protokolldateien an einen Amazon S3 S3-Bucket. Wenn Sie einen Trail in der Konsole erstellen, gilt der Trail standardmäßig für alle AWS-Regionen. Der Trail protokolliert alle Ereignisse AWS-Regionen in der AWS Partition und übermittelt die Protokolldateien an den von Ihnen angegebenen Amazon S3 S3-Bucket. Darüber hinaus können Sie andere AWS Dienste konfigurieren, um die in den CloudTrail Protokollen gesammelten Ereignisdaten weiter zu analysieren und darauf zu reagieren. Weitere Informationen finden Sie in folgenden Themen im AWS CloudTrail -Benutzerhandbuch:

- [Einen Trail für dein AWS Konto erstellen](#)
- [AWS Serviceintegrationen mit Protokollen CloudTrail](#)
- [Konfiguration von Amazon SNS SNS-Benachrichtigungen für CloudTrail](#)
- [Empfangen von CloudTrail Protokolldateien aus mehreren Regionen](#) und [Empfangen von CloudTrail Protokolldateien von mehreren Konten](#)

Alle Amazon EFS [Amazon-EFS-API](#) werden von protokolliert CloudTrail. Beispielsweise generieren Aufrufe von `CreateMountTarget` und `CreateTags` Operationen Einträge in den CloudTrail Protokolldateien. `CreateFileSystem`

Jeder Ereignis- oder Protokolleintrag enthält Informationen zu dem Benutzer, der die Anforderung generiert hat. Die Identitätsinformationen unterstützen Sie bei der Ermittlung der folgenden Punkte:

- Ob die Anfrage mit Root-Benutzer- oder AWS Identity and Access Management (IAM-) Benutzeranmeldedaten gestellt wurde.
- Gibt an, ob die Anforderung mit temporären Sicherheitsanmeldeinformationen für eine Rolle oder einen Verbundbenutzer gesendet wurde.
- Ob die Anfrage von einem anderen AWS Dienst gestellt wurde.

Weitere Informationen finden Sie unter [CloudTrail UserIdentity-Element](#) im AWS CloudTrail Benutzerhandbuch.

Erläuterungen der Amazon-EFS-Protokolldateieinträge

Ein Trail ist eine Konfiguration, die die Übertragung von Ereignissen als Protokolldateien an einen von Ihnen angegebenen Amazon S3 S3-Bucket ermöglicht. CloudTrail Protokolldateien enthalten einen oder mehrere Protokolleinträge. Ein Ereignis stellt eine einzelne Anforderung aus einer beliebigen Quelle dar und enthält Informationen über die angeforderte Aktion, Datum und Uhrzeit der Aktion,

Anforderungsparameter usw. CloudTrail Protokolldateien sind kein geordneter Stack-Trace der öffentlichen API-Aufrufe, sodass sie nicht in einer bestimmten Reihenfolge angezeigt werden.

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der den CreateTags Vorgang demonstriert, wenn ein Tag für ein Dateisystem von der Konsole aus erstellt wird.

```
{
  "eventVersion": "1.06",
  "userIdentity": {
    "type": "Root",
    "principalId": "111122223333",
    "arn": "arn:aws:iam::111122223333:root",
    "accountId": "111122223333",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2017-03-01T18:02:37Z"
      }
    }
  },
  "eventTime": "2017-03-01T19:25:47Z",
  "eventSource": "elasticfilesystem.amazonaws.com",
  "eventName": "CreateTags",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "192.0.2.0",
  "userAgent": "console.amazonaws.com",
  "requestParameters": {
    "fileSystemId": "fs-00112233",
    "tags": [{
      "key": "TagName",
      "value": "AnotherNewTag"
    }]
  },
  "responseElements": null,
  "requestID": "dEXAMPLE-feb4-11e6-85f0-736EXAMPLE75",
  "eventID": "eEXAMPLE-2d32-4619-bd00-657EXAMPLEe4",
  "eventType": "AwsApiCall",
  "apiVersion": "2015-02-01",
  "recipientAccountId": "111122223333"
}
```

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der die DeleteTags Aktion demonstriert, wenn ein Tag für ein Dateisystem von der Konsole gelöscht wird.

```
{
  "eventVersion": "1.06",
  "userIdentity": {
    "type": "Root",
    "principalId": "111122223333",
    "arn": "arn:aws:iam::111122223333:root",
    "accountId": "111122223333",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2017-03-01T18:02:37Z"
      }
    }
  },
  "eventTime": "2017-03-01T19:25:47Z",
  "eventSource": "elasticfilesystem.amazonaws.com",
  "eventName": "DeleteTags",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "192.0.2.0",
  "userAgent": "console.amazonaws.com",
  "requestParameters": {
    "fileSystemId": "fs-00112233",
    "tagKeys": []
  },
  "responseElements": null,
  "requestID": "dEXAMPLE-feb4-11e6-85f0-736EXAMPLE75",
  "eventID": "eEXAMPLE-2d32-4619-bd00-657EXAMPLEe4",
  "eventType": "AwsApiCall",
  "apiVersion": "2015-02-01",
  "recipientAccountId": "111122223333"
}
```

Protokolleinträge für serviceverknüpfte EFS-Rollen

Die mit dem Service verknüpfte Amazon EFS-Rolle führt API-Aufrufe an AWS Ressourcen durch. Es werden CloudTrail Protokolleinträge `username: AWSServiceRoleForAmazonElasticFileSystem` für Aufrufe angezeigt, die von der

dienstverknüpften EFS-Rolle getätigt wurden. Weitere Informationen zu EFS und serviceverknüpften Rollen finden Sie unter [Verwendung von Service-gebundenen Rollen für Amazon EFS](#).

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der eine CreateServiceLinkedRole Aktion demonstriert, wenn Amazon EFS die AWSService RoleForAmazonElasticFileSystem serviceverknüpfte Rolle erstellt.

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "111122223333",
    "arn": "arn:aws:iam::111122223333:user/user1",
    "accountId": "111122223333",
    "accessKeyId": "A111122223333",
    "userName": "user1",
    "sessionContext": {
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2019-10-23T22:45:41Z"
      }
    },
    "invokedBy": "elasticfilesystem.amazonaws.com"
  },
  "eventTime": "2019-10-23T22:45:41Z",
  "eventSource": "iam.amazonaws.com",
  "eventName": "CreateServiceLinkedRole",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "192.0.2.0",
  "userAgent": "user_agent",
  "requestParameters": {
    "aWSServiceName": "elasticfilesystem.amazonaws.com"
  },
  "responseElements": {
    "role": {
      "assumeRolePolicyDocument":
"111122223333-10-111122223333Statement111122223333Action111122223333AssumeRole111122223333Effe
%22%3A%20%22Allow%22%2C%20%22Principal%22%3A%20%7B%22Service%22%3A%20%5B%22
elasticfilesystem.amazonaws.com%22%5D%7D%7D%5D%7D",
      "arn": "arn:aws:iam::111122223333:role/aws-service-role/
elasticfilesystem.amazonaws.com/AWSServiceRoleForAmazonElasticFileSystem",
      "roleId": "111122223333",
      "createDate": "Oct 23, 2019 10:45:41 PM",
    }
  }
}
```

```

        "roleName": "AWSServiceRoleForAmazonElasticFileSystem",
        "path": "/aws-service-role/elasticfilesystem.amazonaws.com/"
    }
},
"requestID": "11111111-2222-3333-4444-abcdef123456",
"eventID": "11111111-2222-3333-4444-abcdef123456",
"eventType": "AwsApiCall",
"recipientAccountId": "111122223333"
}

```

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der eine `CreateNetworkInterface` Aktion demonstriert, die von der `AWSServiceRoleForAmazonElasticFileSystem` serviceverknüpften Rolle ausgeführt wurde, wie in der beschrieben. `sessionContext`

```

{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
    "arn": "arn:aws:sts::0123456789ab:assumed-role/AWSServiceRoleForAmazonElasticFileSystem/0123456789ab",
    "accountId": "0123456789ab",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:iam::0123456789ab:role/aws-service-role/elasticfilesystem.amazonaws.com/AWSServiceRoleForAmazonElasticFileSystem",
        "accountId": "0123456789ab",
        "userName": "AWSServiceRoleForAmazonElasticFileSystem"
      },
      "webIdFederationData": {},
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2019-10-23T22:50:05Z"
      }
    },
    "invokedBy": "AWS Internal"
  },
  "eventTime": "2019-10-23T22:50:05Z",
  "eventSource": "ec2.amazonaws.com",
  "eventName": "CreateNetworkInterface",
  "awsRegion": "us-east-1",

```

```

"sourceIpAddress": "elasticfilesystem.amazonaws.com",
"userAgent": "elasticfilesystem.amazonaws.com",
"requestParameters": {
    "subnetId": "subnet-71e2f83a",
    "description": "EFS mount target for fs-1234567 (fsmt-1234567)",
    "groupSet": {},
    "privateIpAddressesSet": {}
},
"responseElements": {
    "requestId": "0708e4ad-03f6-4802-b4ce-4ba987d94b8d",
    "networkInterface": {
        "networkInterfaceId": "eni-0123456789abcdef0",
        "subnetId": "subnet-12345678",
        "vpcId": "vpc-01234567",
        "availabilityZone": "us-east-1b",
        "description": "EFS mount target for fs-1234567 (fsmt-1234567)",
        "ownerId": "666051418590",
        "requesterId": "0123456789ab",
        "requesterManaged": true,
        "status": "pending",
        "macAddress": "00:bb:ee:ff:aa:cc",
        "privateIpAddress": "192.0.2.0",
        "privateDnsName": "ip-192-0-2-0.ec2.internal",
        "sourceDestCheck": true,
        "groupSet": {
            "items": [
                {
                    "groupId": "sg-c16d65b6",
                    "groupName": "default"
                }
            ]
        },
        "privateIpAddressesSet": {
            "item": [
                {
                    "privateIpAddress": "192.0.2.0",
                    "primary": true
                }
            ]
        },
        "tagSet": {}
    }
},
"requestID": "11112222-3333-4444-5555-666666777777",

```

```

    "eventID": "aaaabbbb-1111-2222-3333-444444555555",
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
  }

```

Protokolleinträge für EFS-Authentifizierung

Amazon EFS-Autorisierung für NFS-Clients, Emitts `NewClientConnection` und `UpdateClientConnection` CloudTrail Ereignisse. Ein `NewClientConnection`-Ereignis wird ausgelöst, wenn eine Verbindung unmittelbar nach einer ersten Verbindung und unmittelbar nach einer erneuten Verbindung autorisiert wird. `UpdateClientConnection` wird ausgegeben, wenn eine Verbindung erneut autorisiert wird und sich die Liste der zulässigen Aktionen geändert hat. Das Ereignis wird auch ausgelöst, wenn die neue Liste der zulässigen Aktionen nicht `ClientMount` enthält. Weitere Informationen zur EFS-Autorisierung finden Sie unter [Verwendung von IAM zur Steuerung des Zugriffs auf Dateisysteme](#).

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der ein `NewClientConnection` Ereignis demonstriert.

```

{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
    "arn": "arn:aws:sts::0123456789ab:assumed-role/abcdef0123456789",
    "accountId": "0123456789ab",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE ",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "AIDACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:iam::0123456789ab:role/us-east-2",
        "accountId": "0123456789ab",
        "userName": "username"
      },
      "webIdFederationData": {},
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2019-12-23T17:50:16Z"
      },
      "ec2RoleDelivery": "1.0"
    }
  }
}

```

```

    },
    "eventTime": "2019-12-23T18:02:12Z",
    "eventSource": "elasticfilesystem.amazonaws.com",
    "eventName": "NewClientConnection",
    "awsRegion": "us-east-2",
    "sourceIPAddress": "AWS Internal",
    "userAgent": "elasticfilesystem",
    "requestParameters": null,
    "responseElements": null,
    "eventID": "27859ac9-053c-4112-ae3-f3429719d460",
    "readOnly": true,
    "resources": [
      {
        "accountId": "0123456789ab",
        "type": "AWS::EFS::FileSystem",
        "ARN": "arn:aws:elasticfilesystem:us-east-2:0123456789ab:file-system/
fs-01234567"
      },
      {
        "accountId": "0123456789ab",
        "type": "AWS::EFS::AccessPoint",
        "ARN": "arn:aws:elasticfilesystem:us-east-2:0123456789ab:access-point/
fsap-0123456789abcdef0"
      }
    ],
    "eventType": "AwsServiceEvent",
    "recipientAccountId": "0123456789ab",
    "serviceEventDetails": {
      "permissions": {
        "ClientRootAccess": true,
        "ClientMount": true,
        "ClientWrite": true
      }
    },
    "sourceIpAddress": "10.7.3.72"
  }
}

```

Amazon EFS-Protokolldateieinträge für encrypted-at-rest Dateisysteme

In Amazon EFS können Sie Verschlüsselung im Ruhezustand, während der Übertragung oder beides verwenden. Weitere Informationen finden Sie unter [Datenverschlüsselung in Amazon EFS](#).

Amazon EFS sendet [Verschlüsselungskontext](#), wenn AWS KMS API-Anfragen zur Generierung von Datenschlüsseln und Entschlüsselung von Amazon EFS-Daten gestellt werden. Die Dateisystem-ID ist der Verschlüsselungskontext für alle Dateisysteme, die im Ruhezustand verschlüsselt sind. Im `requestParameters` Feld eines CloudTrail Protokolleintrags sieht der Verschlüsselungskontext wie folgt aus.

```
"EncryptionContextEquals": {}  
"aws:elasticfilesystem:filesystem:id" : "fs-4EXAMPLE"
```


Verwenden von AWS Abrechnungs- und Nutzungsberichten

Wenn Sie Amazon EFS verwenden, zahlen Sie nur für die Ressourcen, die Sie nutzen, und es fallen keine Mindest- oder Einrichtungsgebühren an. Wie bei anderen AWS Diensten werden die Preise zwar auf monatlicher Basis angegeben, Ihre Nutzung wird jedoch pro Stunde anteilig berechnet, und Ihnen wird Ihre durchschnittliche Nutzung über einen Monat in Rechnung gestellt.

AWS bietet die folgenden Berichte für Amazon EFS:

- **Abrechnungsberichte** — Mehrere Berichte, die einen allgemeinen Überblick über alle Aktivitäten der von Ihnen verwendeten AWS Services bieten, einschließlich Amazon EFS. AWS stellt dem Besitzer des EFS-Dateisystems immer die Amazon EFS-Gebühren in Rechnung. Weitere Informationen über Abrechnungsberichte finden Sie unter [AWS Billing Berichte für Amazon EFS](#).
- **Nutzungsberichte** — Eine Zusammenfassung der Aktivitäten für einen bestimmten Service, aggregiert nach Stunde, Tag oder Monat. Sie können wählen, welche Nutzungsart und welche Operation aufgenommen werden soll. Sie können auch festlegen, wie die Daten aggregiert werden. Weitere Informationen finden Sie unter [AWS Nutzungsberichte für Amazon EFS](#).

Die folgenden Themen enthalten Informationen zur Abrechnung und Nutzungsberichterstattung für Amazon EFS.

Themen

- [EFS-Dateisystem-Tags für die Kostenzuweisung verwenden](#)
- [AWS Billing Berichte für Amazon EFS](#)
- [AWS Nutzungsberichte für Amazon EFS](#)
- [Grundlegendes zu Abrechnungs- und Nutzungsberichten für Amazon EFS](#)

EFS-Dateisystem-Tags für die Kostenzuweisung verwenden

Um die Speicherkosten oder andere Kriterien für einzelne Projekte oder Projektgruppen nachzuverfolgen, kennzeichnen Sie Ihre EFS-Dateisysteme mit Kostenzuweisungs-Tags. Ein Kostenzuweisungs-Tag ist ein Schlüssel-Wert-Paar, das Sie einem Dateisystem zuordnen. Nachdem Sie die Kostenzuordnungs-Tags aktiviert haben, AWS verwendet diese Tags, um Ihre Ressourcenkosten in Ihrem Kostenzuordnungsbericht zu organisieren.

Der Kostenzuordnungsbericht listet die AWS Nutzung für Ihr Konto nach Produktkategorie und verknüpftem Kontonutzer auf. Der Bericht enthält die gleichen Einzelposten wie der detaillierte Fakturierungsbericht (siehe [Grundlegendes zu Abrechnungs- und Nutzungsberichten für Amazon EFS](#)), und zusätzliche Spalten für Ihre Tag-Schlüssel.

AWS bietet zwei Arten von Kostenzuordnungs-Tags: AWS generierte Tags und benutzerdefinierte Tags. AWS definiert, erstellt und wendet den AWS-generierten `aws:createdBy` Tag-Schlüssel für Sie nach einem Amazon CreateFileSystem EFS-Ereignis an. Sie definieren, erstellen und wenden benutzerdefinierte Tags auf Ihre Dateisysteme an.

Sie müssen beide Tag-Typen in der Fakturierungs- und Kostenverwaltungskonsole separat aktivieren, damit sie in Ihren monatlichen Berichten angezeigt werden können. Weitere Informationen zu AWS-generierten Tags finden Sie im AWS Billing Benutzerhandbuch unter [Organisieren und Nachverfolgen von Kosten mithilfe von AWS Kostenverteilungs-Tags](#).

- Weitere Informationen zum Erstellen benutzerdefinierter Tags für EFS-Ressourcen finden Sie unter [Taggen von EFS-Ressourcen](#).
- Weitere Informationen zur Aktivierung von benutzerdefinierten Stichwörtern in der Billing and Cost Management-Konsole finden Sie im AWS Billing Benutzerhandbuch unter [Aktivieren von benutzerdefinierten Kostenzuordnungs-Tags](#).
- Weitere Informationen zur Aktivierung von AWS-generierten Tags in der Billing and Cost Management-Konsole finden Sie im AWS Billing Benutzerhandbuch unter [Aktivieren von AWS-generierten Kostenzuordnungs-Tags](#).

Benutzerdefinierte Kostenzuordnungs-Markierungen

Ein benutzerdefiniertes Kostenzuordnungs-Tag umfasst folgende Komponenten:

- Der Tag-Schlüssel. Der Tag-Schlüssel ist der Name des Markierungen. Im Tag `project/Trinity` beispielsweise ist `project` der Schlüssel. Die Tag-Schlüssel ist eine Zeichenfolge, in der zwischen Groß- und Kleinschreibung unterschieden wird und die 1 bis 128 Unicode-Zeichen enthalten kann.
- Der Tag-Wert. Der Tag-Wert ist eine erforderliche Zeichenfolge. Im Tag `project/Trinity` beispielsweise ist `Trinity` der Wert. Die Tag-Wert ist eine Zeichenfolge, in der zwischen Groß- und Kleinschreibung unterschieden wird und die 0 bis 256 Unicode-Zeichen enthalten kann.

Weitere Informationen zu benutzerdefinierten Stichwörtern finden Sie im Benutzerhandbuch unter [Verwenden von benutzerdefinierten Kostenverrechnungstags](#).AWS Billing

AWS Billing Berichte für Amazon EFS

Ihre monatliche Rechnung von AWS unterteilt Ihre Nutzungsinformationen und Kosten nach AWS Service und Funktion. Es sind mehrere AWS Abrechnungsberichte verfügbar: der Monatsbericht, der Kostenzuweisungsbericht und detaillierte Abrechnungsberichte. Weitere Informationen finden Sie im AWS BillingBenutzerhandbuch unter [Grundlegendes zu Ihrer Rechnung](#).

Für detailliertere Berichte über Ihre Amazon EFS-Speichernutzung verwenden Sie, AWS Data Exports um Exporte von AWS Cost and Usage Report (AWS CUR) zu erstellen. Mit AWS CUR 2.0 veröffentlichen Sie Ihre AWS Abrechnungsberichte in einem Amazon Simple Storage Service (Amazon S3) -Bucket, den Sie besitzen. Der Amazon EFS-Nutzungsbericht listet die Vorgänge nach Nutzungstyp und auf AWS-Region. Weitere Informationen finden Sie im [Cost and Usage Report \(CUR\) 2.0](#) im AWS Data Exports Benutzerhandbuch.

EFS-Dateisysteme werden auf der Grundlage der folgenden Nutzungskategorien abgerechnet.

- Speicher (pro Gigabyte [GB]) — Die Menge der in Ihren EFS-Dateisystemen pro Monat gespeicherten Daten, wobei die Preise je nach Speicherklasse (EFS Standard, EFS Infrequent Access (IA) und EFS Archive) variieren.
- Durchsatz (pro GB oder Mebibyte pro Sekunde [MiBps]) — Die Menge der übertragenen Daten (Lese-/Schreibvorgänge) oder die Menge, die Ihre bereitgestellte Menge übersteigt, hängt vom Durchsatz des Dateisystems pro Monat ab.
- Datenzugriff und -übertragung (pro MiBps) — Die Menge der pro Monat gelesenen und in die Speicherklassen EFS Infrequent Access (IA) und EFS Archive übertragenen Daten.
- Backup-Speicher (pro GB) — Die Menge an Speicherplatz, die von Backups pro Monat verbraucht wird. Wenn Sie die Aufbewahrungsdauer Ihrer Backups verlängern oder zusätzliche, vom Benutzer initiierte Volumen-Backups durchführen, erhöht sich die Menge an Backup-Speicherplatz, den Ihr Dateisystem verbraucht. Weitere Informationen finden Sie unter [Sicherung von EFS-Dateisystemen](#).
- Replikation (pro GB) — Die aktuellen Speicher- und Lese-, Schreib- und Tiering-Aktivitätsraten für Ihre EFS-Quell- und Zieldateisysteme sowie alle anfallenden Gebühren für Datenübertragungsaktivitäten zwischen diesen Systemen pro Monat.

Ausführliche Informationen zu den Amazon EFS-Gebühren finden Sie unter [Amazon EFS-Preise](#).

Informationen zum Verständnis der in den Abrechnungs- und Nutzungsberichten für Amazon EFS verwendeten Codes und Abkürzungen finden Sie unter [Grundlegendes zu Abrechnungs- und Nutzungsberichten für Amazon EFS](#).

AWS Nutzungsberichte für Amazon EFS

Wenn Sie einen Nutzungsbericht herunterladen, können Sie die Nutzungsdaten nach Stunde, Tag und Monat zusammenfassen. Der Amazon EFS-Nutzungsbericht listet die Vorgänge nach Nutzungstyp und auf AWS-Region. Für detailliertere Berichte über Ihre Amazon EFS-Speichernutzung laden Sie dynamisch generierte AWS Nutzungsberichte herunter. Sie können wählen, welche Nutzungsart, welche Operation und welcher Zeitraum aufgenommen werden soll. Sie können auch festlegen, wie die Daten aggregiert werden. Weitere Informationen zum Herunterladen von Nutzungsberichten finden Sie unter [Herunterladen eines AWS Nutzungsberichts](#) im AWS Data Exports User Guide.

Der Amazon EFS-Nutzungsbericht enthält die folgenden Informationen:

- Service — Amazon EFS
- UsageType— Einer der folgenden Werte:
 - Ein Code zur Identifizierung des Speichertyps
 - Ein Code zur Identifizierung der Anfrage
 - Ein Code zur Identifizierung des Typs der Datenübertragung
 - Ein Code, der den Durchsatzmodus identifiziert
 - Ein Code, der die Backup-Nutzung identifiziert

Eine ausführliche Erläuterung der Amazon EFS-Nutzungstypen finden Sie unter [Grundlegendes zu Abrechnungs- und Nutzungsberichten für Amazon EFS](#).

- Ressource — Der Name der Ressource, die der aufgeführten Nutzung zugeordnet ist.
- StartTime— Startzeit des Tages, für den die Nutzung gilt, in koordinierter Weltzeit (UTC).
- EndTime— Endzeit des Tages, für den die Verwendung gilt, in koordinierter Weltzeit (UTC).
- UsageValue— Einer der folgenden Volumenwerte. Die typische Maßeinheit für Daten ist Gigabyte (GB). Abhängig vom Dienst und dem Bericht können stattdessen Terabytes (TB) angezeigt werden.
 - Die Anzahl der Anforderungen in dem angegebenen Zeitraum
 - Die übertragene Datenmenge

- Die Menge der in einer bestimmten Stunde gespeicherten Daten

Informationen zum Verständnis der in den Abrechnungs- und Nutzungsberichten für Amazon EFS verwendeten Codes und Abkürzungen finden Sie unter [Grundlegendes zu Abrechnungs- und Nutzungsberichten für Amazon EFS](#).

Grundlegendes zu Abrechnungs- und Nutzungsberichten für Amazon EFS

In den Abrechnungs- und Nutzungsberichten von Amazon EFS werden Codes und Abkürzungen verwendet. Ersetzen Sie die Verwendungsarten in der folgenden Tabelle *region* durch Abkürzungen aus dieser Liste:

- APE1: Asien-Pazifik (Hongkong)
- APN1: Asien-Pazifik (Tokio)
- APN2: Asien-Pazifik (Seoul)
- APN3: Asien-Pazifik (Osaka)
- APS1: Asien-Pazifik (Singapur)
- APS2: Asien-Pazifik (Sydney)
- APS3: Asien-Pazifik (Mumbai)
- APS4: Asien-Pazifik (Jakarta)
- APS5: Asien-Pazifik (Hyderabad)
- APS6: Asien-Pazifik (Melbourne)
- CAN1: Kanada (Zentral)
- CAN2: Kanada West (Calgary)
- CNN1: China (Peking)
- CNW1: China (Ningxia)
- AFS1: Afrika (Kapstadt)
- EUC2: Europa (Zürich)
- EUN1: Europa (Stockholm)
- EUS2: Europa (Spanien)
- EUC1: Europa (Frankfurt)

- EU: Europa (Irland)
- EUS1: Europa (Mailand)
- EUW2: Europa (London)
- EUW3: Europa (Paris)
- ILC1: Israel (Tel Aviv)
- MEC1: Naher Osten (VAE)
- MES1: Naher Osten (Bahrain)
- SAE1: Südamerika (São Paulo)
- UGW1: AWS GovCloud (US-West)
- UGE1: AWS GovCloud (US-Ost)
- USE1 (oder kein Präfix): US East (Nord-Virginia)
- USE2: USA Ost (Ohio)
- USW1: USA West (Nordkalifornien)
- USW2: USA West (Oregon)

Informationen zur Preisgestaltung von AWS-Region finden Sie unter [Amazon EFS-Preise](#).

Die erste Spalte in der folgenden Tabelle listet die Nutzungstypen auf, die in Ihren Abrechnungs- und Nutzungsberichten vorkommen. Die typische Maßeinheit für Daten ist Gigabyte (GB). Für den bereitgestellten Durchsatz wird jedoch stattdessen Megabyte pro Sekunde (MiBps) verwendet.

Verwendungsarten für Amazon EFS

Verwendungstyp	CloudWatch Metrik	Einheit	Granularität	Description
<i>region-</i> Archived DataAccess- Bytes	–	GB	–	Die Datenmenge, die in den EFS-Archivspeicher übertragen wurde oder auf die im EFS-Archivspeicher zugegriffen wurde.
<i>region-</i> ArchiveE	–	GB-Mo	–	Anteilige Speichernutzung für Dateien, die vor Ablauf der 90-Tage-Mindestverpflichtung aus dem EFS-Archivspeicher gelöscht wurden.

Verwendungstyp	CloudWatch Metrik	Einheit	Granularität	Description
earlyDelete-ByteHrs				
<i>region</i> -ArchiveEarlyDelete-SmallFiles	–	GB pro Monat	–	Anteilige Speichernutzung für kleine Dateien (kleiner als 128 KB), die vor Ablauf der Mindestlaufzeit von 90 Tagen aus dem EFS-Archivspeicher gelöscht wurden.
<i>region</i> -ArchiveTimedStorage-ByteHrs	StorageElasticArchives	GB pro Monat	Stündlich	Die Anzahl der GB-Monate, in denen Daten im EFS-Archivspeicher gespeichert wurden.
<i>region</i> -ArchiveTimedStorage-SmallFiles	StorageElasticArchivesSizeOnRead	GB pro Monat	Stündlich	Die Anzahl der GB-Monate, in denen kleine Objekte (kleiner als 128 KB) im EFS-Archivspeicher gespeichert wurden.
<i>region</i> -ETDataAccess-Bytes	TotalIOBytes	GB	Stundensatz	Die Datenmenge, die mit Elastic Throughput für Dateien geschrieben oder gelesen wurde.
<i>region</i> -IADataAccess-Bytes	–	GB	Stundensatz	Die Datenmenge, die in den EFS Infrequent Access (IA) -Speicher übertragen wurde oder auf die dort zugegriffen wurde.
<i>region</i> -IATimedStorage-ByteHrs	StorageElasticIA	GB-Mo	Stündlich	Die Anzahl der GB-Monate, in denen Daten im EFS IA-Speicher gespeichert wurden.

Verwendungstyp	CloudWatch Metrik	Einheiten	Granularität	Description
<i>region-</i> IATimedStorage-ET-ByteHrs	StorageE tes.IA	GB pro Monat	Stündlich	Die Anzahl der GB-Monate, in denen Daten für Dateien, die Elastic Throughput verwenden, im EFS IA-Speicher gespeichert wurden.
<i>region-</i> IATimedStorage-ET-SmallFiles	StorageE tes.IASiz eOverhe	GB pro Monat	Stündlich	Die Anzahl der GB-Monate, in denen kleine Dateien (kleiner als 128 KB) im EFS IA-Speicher gespeichert wurden, für Dateien, die Elastic Throughput verwenden.
<i>region-</i> IATimedStorage-SmallFiles	StorageE tes.IASiz eOverhe	GB-Mo	Stündlich	Die Anzahl der GB-Monate, in denen kleine Dateien (kleiner als 128 KB) im EFS IA-Speicher gespeichert wurden.
<i>region-</i> IATimedStorage-Z-ByteHrs	StorageE tes.IA	GB pro Monat	Stündlich	Die tatsächliche Größe der im EFS IA-Speicher für One Zone-Dateisysteme gespeicherten Daten.
<i>region-</i> IATimedStorage-Z-SmallFiles	StorageE tes.IASiz eOverhe	GB-Mo	Stündlich	Die Anzahl der GB-Monate, in denen kleine Dateien (kleiner als 128 KB) im EFS IA-Speicher für One Zone-Dateisysteme gespeichert wurden.
<i>region-</i> IncludedTP-MiBpsHrs	Permitter Through t	MiBps- Mein	Stündlich	Die maximale Durchsatzmenge, die ein Dateisystem bewältigen kann.
<i>region-</i> ProvisionedTP-MiBpsHrs	Permitter Through t	MiBps- Mein	Stündlich	Bei Dateisystemen, die den bereitgestellten Durchsatz verwenden, der bereitgestellte Durchsatz oder der maximal zulässige Durchsatz.

Verwendungstyp	CloudWatch Metrik	Einheit	Granularität	Description
<i>region-</i> TimedStorage-ByteHrs	StorageElasticFileSystemStandard	GB pro Monat	Stündlich	Die tatsächliche Gesamtmenge der im EFS-Standardspeicher für regionale Dateisysteme gespeicherten Daten.
<i>region-</i> TimedStorage-Z-ByteHrs	StorageElasticFileSystemOneZoneStandard	GB-Mo	Stündlich	Die tatsächliche Gesamtmenge der im EFS-Standardspeicher für One Zone-Dateisysteme gespeicherten Daten.

Hinweise

- EFS-Archivspeicher wird für eine Mindestspeicherdauer von 90 Tagen berechnet, auch wenn die Daten vor 90 Tagen gelöscht werden.
- Eine Tiering-Gebühr fällt für Daten an, die vom EFS-Standardspeicher zum EFS-Speicher mit infrequenter Zugriff (IA) und vom EFS IA-Speicher zum EFS-Standardspeicher aufgeteilt werden.
- EFS IA und EFS Archive Storage haben eine gebührenpflichtige Mindestdateigröße von 128 KiB. Dateien, die kleiner als 128 KiB sind, können diesen Speicherklassen zugeordnet werden, es fallen jedoch 128 KiB Speicherplatz zum entsprechenden Speicherklassentarif an.
- Bei Dateisystemen, die Elastic Throughput verwenden, werden Metadatenoperationen nach den ersten 4 KiB in Schritten von 1 KiB und Datenoperationen nach den ersten 32 KiB in Schritten von 1 KiB gemessen.
- Die Speichernutzung wird in binären Gigabyte (GB) berechnet, wobei 1 GB 2³⁰ Byte entspricht. Weitere Informationen [Wie Amazon EFS Dateisystem- und Objektgrößen meldet](#) zur Messung der Speicher- und Durchsatznutzung zur Berechnung Ihrer Amazon EFS-Rechnung finden Sie unter.
- Der GB-Monat wird abgeleitet, indem die Gesamtzahl der GB-Stunden über einen Monat aggregiert und dann durch die Anzahl der Stunden in diesem Monat dividiert wird.

Nachverfolgung von Vorgängen in Ihren Nutzungsberichten

Operationen beschreiben die Aktion, die auf Ihrem EFS-Dateisystem durch den angegebenen Verwendungstyp ausgeführt wird. Vorgänge werden durch selbsterklärende Codes beschrieben, z. B. `Read` oder `Delete`. Verwenden Sie diese Codes, um zu sehen, welche Aktionen in Ihrem Dateisystem zu einer bestimmten Verwendungsart geführt haben. Wenn Sie einen Nutzungsbericht erstellen, können Sie wählen, All Operations aufzunehmen oder eine bestimmte Operation, z. B. `Write`, für die ein Bericht erstellt werden soll.

Leistungsspezifikationen von Amazon EFS

Die folgenden Abschnitte geben einen Überblick über Amazon-EFS-Leistung und wie sich Ihre Dateisystemkonfiguration auf wichtige Leistungsdimensionen auswirkt. Wir bieten auch einige wichtige Tipps und Empfehlungen zur Optimierung der Leistung Ihres Dateisystems.

Themen

- [Zusammenfassung der Leistung](#)
- [Speicherklassen](#)
- [Leistungsmodi](#)
- [Durchsatzmodi](#)
- [Tipps zur Amazon-EFS-Leistung](#)
- [Behebung von Amazon EFS-Leistungsproblemen](#)
- [Beheben von AML- und Kernel-Problemen](#)

Zusammenfassung der Leistung

Die Dateisystemleistung wird in der Regel anhand der Dimensionen Latenz, Durchsatz und Input/Output Operationen pro Sekunde (IOPS) gemessen. Die Leistung von Amazon EFS in diesen Dimensionen hängt von der Konfiguration Ihres Dateisystems ab. Die folgenden Konfigurationen wirken sich auf die Leistung eines Amazon-EFS-Dateisystems aus:

- Dateisystemtyp – Regional oder One Zone
- Leistungsmodus – Allzweck oder Max. E/A

Important

Der Modus Max I/O Performance hat höhere Latenzen pro Vorgang als der Allzweck-Performance-Modus. Für eine schnellere Leistung empfehlen wir, immer den Allzweck-Leistungsmodus zu verwenden. Weitere Informationen finden Sie unter [Leistungsmodi](#).

- Durchsatzmodus – Elastic, Bereitgestellt oder Bursting

In der folgenden Tabelle sind die Leistungsspezifikationen für Dateisysteme, die den Allzweck-Leistungsmodus verwenden, sowie die möglichen unterschiedlichen Kombinationen von Dateisystemtyp und Durchsatzmodus aufgeführt.

Leistungsspezifikationen für Dateisysteme, die den Allzweck-Leistungsmodus verwenden

Konfiguration von Speicher und Durchsatz		Latenz ¹		Maximale IOPS		Maximaler Durchsatz		
Dateisystemtyp	Durchsatzmodus	Lesevorgänge	Schreibvorgänge	Lesevorgänge	Schreibvorgänge	Per-file-system liest ²	Per-file-system schreibt ²	Lesen/Schreiben pro Client
Regional	Elastic	~1 Millisekunden (ms)	~2,7 Millisekunden (ms)	900.000—2.500.000 ³	500.000 ³	20—60 Gibibyte pro Sekunde () GiBps	1—5 GiBps	1.500 Mebibyte pro Sekunde (4) MiBps
Regional	Bereitgestellt	~1 ms	~2,7 ms	55.000	25.000	3—10 GiBps	1—3,33 GiBps	500 MiBps
Regional	Bursting	~1 ms	~2,7 ms	35 000	7.000	3—5 GiBps	1—3 GiBps	500 MiBps
One Zone	Elastisch, bereitgestellt, explosionssartig	~1 ms	~1,6 ms	35 000	7.000	3,5 GiBps	1 ⁵ GiBps	500 MiBps

- Die angegebenen Latenzwerte stellen eine optimale Leistung unter optimalen Bedingungen dar. Die tatsächlichen Ergebnisse können je nach Netzwerk, Arbeitslast und Systemfaktoren variieren.

2. Der maximale Lese- und Schreibdurchsatz hängt von der AWS-Region ab. Ein Durchsatz, der den maximalen Durchsatz AWS-Region von an überschreitet, erfordert eine Erhöhung der Durchsatzquote. Jede Anfrage nach zusätzlichem Durchsatz wird vom Amazon EFS-Serviceteam auf case-by-case Basis geprüft. Die Genehmigung kann von Ihrer Art des Workloads abhängen. Weitere Informationen zum Anfordern einer Kontingenterhöhung finden Sie unter [Amazon EFS-Kontingente](#).
3. Standardmäßig bieten Dateisysteme, die Elastic Throughput verwenden, maximal 90.000 Lese-IOPS für selten aufgerufene Daten, 250.000 Lese-IOPS für häufig abgerufene Daten und 50.000 Schreib-IOPS. Wenn Ihr Workload mehr IOPS benötigt, können Sie eine Erhöhung um das bis zu 10-fache dieser Zahlen beantragen. Weitere Informationen finden Sie unter [Amazon-EFS-Kontingente, die Sie erhöhen können](#). Es gelten zusätzliche Empfehlungen, um maximale IOPS zu erreichen. Weitere Informationen finden Sie unter [the section called “Optimierung von Workloads, die einen hohen Durchsatz und IOPS erfordern”](#).
4. Der maximale kombinierte Lese- und Schreibdurchsatz beträgt 1.500 MiBps für Dateisysteme, die Elastic Throughput verwenden und mit Version 2.0 oder höher des Amazon EFS-Clients (amazon-efs-utils Version) oder des Amazon EFS CSI-Treibers (aws-efs-csi-driver) gemountet wurden. Für alle anderen Dateisysteme liegt das Durchsatzlimit bei 500 MiBps. Weitere Informationen zum Amazon EFS-Client finden Sie unter [Den Amazon EFS-Client installieren](#).
5. One-Zone-Dateisysteme, die Bursting-Durchsatz verwenden, können den gleichen per-file-system Lese- und Schreibdurchsatz erzielen wie regionale Dateisysteme, die Bursting-Durchsatz verwenden (maximal 5 Lesevorgänge GiBps für Lese- und 3 GiBps Schreibvorgänge).

Speicherklassen

Amazon-EFS-Speicherklassen sind je nach Anwendungsfall für die effektivste Speicherung konzipiert.

- Die EFS-Standard-Speicherklasse verwendet Solid-State-Drive-Speicher (SSD), um die geringste Latenz für häufig aufgerufene Dateien zu gewährleisten. Diese Speicherklasse bietet Latenzen im ersten Byte von nur 1 Millisekunde für Lesevorgänge und 2,7 Millisekunden für Schreibvorgänge.
- Die Speicherklassen EFS Infrequent Access (IA) und EFS Archive speichern Daten, auf die seltener zugegriffen wird, für die nicht die Latenzleistung erforderlich ist, die für häufig abgerufene Daten erforderlich ist. Diese Speicherklassen bieten Latenzen im ersten Byte von mehreren zehn Millisekunden.

Weitere Informationen über EFS-Speicherklassen finden Sie unter [the section called “EFS-Speicherklassen”](#).

Leistungsmodi

Amazon EFS bietet zwei Leistungsmodi: Allzweck und Max. E/A.

- Der Allzweckmodus hat die niedrigste Latenz pro Vorgang und ist der Standardleistungsmodus für Dateisysteme. One-Zone-Dateisysteme verwenden immer den Allzweck-Leistungsmodus. Für eine schnellere Leistung empfehlen wir, immer den Allzweck-Leistungsmodus zu verwenden.
- I/O Der Max-Modus ist ein Performance-Typ der vorherigen Generation, der für stark parallelisierte Workloads konzipiert wurde, die höhere Latenzen tolerieren können als der Allzweckmodus. I/O Der Max-Modus wird für One Zone-Dateisysteme oder Dateisysteme, die Elastic Throughput verwenden, nicht unterstützt.

Important

Aufgrund der höheren Latenzen pro Vorgang beim Modus „Max. E/A“ empfehlen wir, für alle Dateisysteme den Allzweckleistungsmodus zu verwenden.

Um sicherzustellen, dass Ihr Workload innerhalb der IOPS-Grenze bleibt, die für Dateisysteme verfügbar ist, die den allgemeinen Leistungsmodus verwenden, können Sie die `PercentIOLimit` CloudWatch Metrik überwachen. Weitere Informationen finden Sie unter [CloudWatch Metriken für Amazon EFS](#).

Anwendungen können ihre IOPS elastisch bis zu dem mit dem Leistungsmodus verbundenen Grenzwert skalieren. IOPS werden Ihnen nicht separat in Rechnung gestellt; sie sind in der Durchsatzabrechnung eines Dateisystems enthalten. Jede NFS-Anfrage (Network File System) wird als Durchsatz von 4 Kilobyte (KB) oder als tatsächliche Anfrage- und Antwortgröße berechnet, je nachdem, welcher Wert größer ist.

Durchsatzmodi

Der Durchsatzmodus eines Dateisystems bestimmt den Durchsatz, der Ihrem Dateisystem zur Verfügung steht. Amazon EFS bietet drei Durchsatzmodi: Elastic, Bereitgestellt und Bursting. Der Lesedurchsatz wird reduziert, damit Sie einen höheren Lese- als Schreibdurchsatz erzielen können.

Der maximale Durchsatz, der in jedem Durchsatzmodus verfügbar ist, hängt von der AWS-Region ab. Weitere Informationen zum maximalen Dateisystemdurchsatz in den verschiedenen Regionen finden Sie unter [Amazon EFS-Kontingente](#).

Ihr Dateisystem kann zusammen einen Lese- und Schreibdurchsatz von 100 % erreichen. Wenn Ihr Dateisystem beispielsweise 33 % seines Limits für den Lesedurchsatz ausnutzt, kann das Dateisystem gleichzeitig bis zu 67 % seines Limits für den Schreibdurchsatz erreichen. Sie können die Durchsatzauslastung Ihres Dateisystems im Diagramm Durchsatzauslastung (%) auf der Seite mit den Dateisystemdetails der Konsole überwachen. Weitere Informationen finden Sie unter [Überwachung der Durchsatzleistung](#).

Auswählen des richtigen Durchsatzmodus für ein Dateisystem

Die Wahl des richtigen Durchsatzmodus für Ihr Dateisystem hängt von den Leistungsanforderungen Ihres Workloads ab.

- **Elastischer Durchsatz (empfohlen)** — Verwenden Sie den standardmäßigen Elastic Throughput, wenn Sie hohe oder unvorhersehbare Workloads haben und Leistungsanforderungen haben, die schwer vorherzusagen sind, oder wenn Ihre Anwendung den Durchsatz mit einem average-to-peak Verhältnis von 5% oder weniger steigert. Weitere Informationen finden Sie unter [Elastischer Durchsatz](#).
- **Bereitgestellter Durchsatz** — Verwenden Sie den bereitgestellten Durchsatz, wenn Sie die Leistungsanforderungen Ihres Workloads kennen oder wenn Ihre Anwendung den Durchsatz mit einem average-to-peak Verhältnis von 5% oder mehr steigert. Weitere Informationen finden Sie unter [Bereitgestellter Durchsatz](#).
- **Bursting-Durchsatz** — Verwenden Sie den Bursting-Durchsatz, wenn Sie einen Durchsatz wünschen, der mit der Speichermenge in Ihrem Dateisystem skaliert.

Wenn Sie nach der Verwendung des Bursting-Durchsatzes feststellen, dass Ihre Anwendung durch den Durchsatz eingeschränkt ist (sie verwendet beispielsweise mehr als 80% des zulässigen Durchsatzes oder Sie haben alle Ihre Burst-Credits aufgebraucht), sollten Sie entweder Elastic oder Provisioned Throughput verwenden. Weitere Informationen finden Sie unter [Bursting-Durchsatz](#).

Weitere Informationen zu Amazon-EFS-Metriken finden Sie unter [CloudWatch Metriken für Amazon EFS](#).

Elastischer Durchsatz

Für Dateisysteme, die Elastic Throughput verwenden, skaliert Amazon EFS die Durchsatzleistung automatisch nach oben oder unten, um den Anforderungen Ihrer Workload-Aktivität gerecht zu werden. Elastischer Durchsatz ist der beste Durchsatzmodus für hohe oder unvorhersehbare Workloads mit Leistungsanforderungen, die schwer vorherzusagen sind, oder für Anwendungen, bei denen der Durchsatz im Durchschnitt bei 5% oder weniger des Spitzendurchsatzes liegt (das average-to-peak Verhältnis).

Da die Durchsatzleistung für Dateisysteme mit Elastic Throughput automatisch skaliert wird, müssen Sie die Durchsatzkapazität nicht spezifizieren oder bereitstellen, um Ihre Anwendungsanforderungen zu erfüllen. Sie zahlen nur für die Menge der gelesenen oder geschriebenen Metadaten und Daten, und Sie sammeln oder verbrauchen bei der Nutzung von Elastic Throughput keine zusätzlichen Credits.

Note

Elastic Throughput ist zwar darauf ausgelegt, elastisch mit Ihrem Durchsatz zu skalieren, wir empfehlen jedoch, im Rahmen Ihrer betrieblichen Best Practices eine angemessene Steuerung durch Monitoring-Metriken mit CloudWatch (gemessenenIOBytes) und Nutzungswarnungen zu implementieren. Dies hilft Ihnen dabei, eine optimale Ressourcennutzung aufrechtzuerhalten und Ihre geplanten Betriebsparameter einzuhalten. Weitere Informationen finden Sie unter [Metriken mit Amazon überwachen CloudWatch](#).

Informationen zu den Elastic-Durchsatzgrenzwerten pro Region finden Sie unter [Amazon-EFS-Kontingente, die Sie erhöhen können](#).

Bereitgestellter Durchsatz

Mit Provisioned Throughput geben Sie einen Durchsatz an, den das Dateisystem unabhängig von der Größe oder dem hohen Guthabenstand des Dateisystems erreichen kann. Verwenden Sie den bereitgestellten Durchsatz, wenn Sie die Leistungsanforderungen Ihres Workloads kennen oder wenn Ihre Anwendung den Durchsatz um 5% oder mehr des average-to-peak Verhältnisses erhöht.

Bei Dateisystemen, die den bereitgestellten Durchsatz verwenden, wird Ihnen der für das Dateisystem aktivierte Durchsatz in Rechnung gestellt. Der in einem Monat in Rechnung gestellte Durchsatzbetrag basiert auf dem bereitgestellten Durchsatz, der den in Ihrem Dateisystem

enthaltenen Basisdurchsatz aus dem Standardspeicher übersteigt, bis zu den geltenden Limits für den Bursting-Basisdurchsatz in der AWS-Region.

Wenn der Basisdurchsatz des Dateisystems den bereitgestellten Durchsatz überschreitet, wird automatisch der für das Dateisystem zulässige Bursting-Durchsatz verwendet (bis zu den jeweils geltenden Grenzwerten für den Bursting-Baseline-Durchsatz). AWS-Region

Informationen zu den pro Region bereitgestellten Durchsatzgrenzwerten finden Sie unter. [Amazon-EFS-Kontingente, die Sie erhöhen können](#)

Bursting-Durchsatz

Ein Bursting-Durchsatz wird für Workloads empfohlen, bei denen ein Durchsatz erforderlich ist, der mit der Speichermenge in Ihrem Dateisystem skaliert. Beim Bursting-Durchsatz ist der Basisdurchsatz proportional zur Größe des Dateisystems in der Standard-Speicherkategorie, und zwar mit einer Rate von 50 KiBps pro GiB Speicher. Burst-Guthaben fallen an, wenn das Dateisystem weniger als die Basisdurchsatzrate verbraucht, und werden abgezogen, wenn der Durchsatz die Basisrate überschreitet.

Wenn Burst-Credits verfügbar sind, kann ein Dateisystem einen Durchsatz von bis zu 100 MiBps pro TiB im Standardspeicher (50 KiBps pro GiB) bis zum AWS-Region Limit von mindestens 100 MiBps erhöhen. Wenn keine Burst-Credits verfügbar sind, kann ein Dateisystem bis zu 50 MiBps pro TiB Speicherplatz speichern, mindestens jedoch 1. MiBps

Informationen zum Bursting-Durchsatz pro Region finden Sie unter. [General resource quotas that cannot be changed](#)

Wissenswertes zu Amazon-EFS-Burst-Guthaben

Beim Bursting-Durchsatz sammelt jedes Dateisystem im Laufe der Zeit Burst-Credits mit einer Basisrate, die von der Größe des Dateisystems bestimmt wird, das in der EFS-Standard-Speicherkategorie gespeichert ist. Die Basisrate beträgt 50 MiBps pro Tebibyte [TiB] Speicher (entspricht 50 KiBps pro GiB Speicher). Amazon EFS misst Lesevorgänge bis zu einem Drittel der Rate von Schreibvorgängen, sodass das Dateisystem eine Basisrate von bis zu 150 KiBps pro GiB Lesedurchsatz oder 50 KiBps pro GiB Schreibdurchsatz erreichen kann.

Ein Dateisystem kann den Durchsatz kontinuierlich mit seiner gemessenen Basisrate erhöhen. Ein Dateisystem sammelt immer dann Burst-Guthaben an, wenn es inaktiv ist oder den Durchsatz unter die gemessene Basisrate treibt. Gesammelte Burst-Gutschriften ermöglichen dem Dateisystem, den Durchsatz über die Grundrate hinaus zu erhöhen.

Beispielsweise hat ein Dateisystem mit 100 GiB an gemessenen Daten in der Standardspeicherklasse einen Basisdurchsatz von 5. MiBps. Über einen Zeitraum von 24 Stunden Inaktivität erhält das Dateisystem Guthaben im Wert von 432.000 MiB ($5 \text{ MiB} \times 86.400 \text{ Sekunden} = 432.000 \text{ MiB}$), das verwendet werden kann, um 72 Minuten MiBps lang bei 100 zu bursten ($432.000 \text{ MiB} \div 100 = 72 \text{ Minuten}$). MiBps

Dateisysteme, die größer als 1 TiB sind, können stets für bis zu 50 % der Zeit ein Bursting ausführen, wenn sie über die verbleibenden 50 % der Zeit inaktiv sind.

Die folgende Tabelle enthält Beispiele für das Bursting-Verhalten.

Größe des Dateisystems	Bursting-Durchsatz	Basisdurchsatz
100 GiB gemessene r Daten im Standardspeicher	- Bis zu 72 Minuten pro Tag auf 300 () MiBps im Nur-Lese-Modus hochfahren oder - Bis zu 72 Minuten pro MiBps Tag im Burst-Modus auf 100 mit Schreibzugriff	- Bis zu 15 MiBps Laufwerke können kontinuierlich nur gelesen werden - Bis zu 5 Laufwerke ununterbrochen mit Schreibzugriff MiBps
1 TiB gemessene r Daten im Standardspeicher	- Schnellzugriff auf 300 MiBps Schreibzugriff für 12 Stunden pro Tag, oder - Erhöhen Sie den Wert auf 100, wenn MiBps Sie nur 12 Stunden pro Tag schreiben	- Drive 150 MiBps kann kontinuierlich nur gelesen werden - Laufwerk 50 MiBps ist kontinuierlich schreibgeschützt
10 TiB gemessene r Daten im Standardspeicher	- Burst 3 mit GiBps Schreibzugriff für 12 Stunden pro Tag, oder 1 Burst-Modus mit GiBps Schreibzugriff für 12 Stunden pro Tag	- Drive 1.5 durchgehend schreibgeschützt GiBps - Drive 500 MiBps ist kontinuierlich schreibgeschützt
Im Allgemeinen größere Dateisysteme	für 12 Stunden pro Tag auf 300 MiBps Nur-Lesevorgänge pro TiB Speicher hochfahren, oder	Fahren Sie kontinuierlich 150 MiBps schreibgeschützte Laufwerke pro TiB Speicher

Größe des Dateisystems	Bursting-Durchsatz	Basisdurchsatz
	Steigen Sie auf 100 MiBps Schreibzugriff pro TiB Speicher für 12 Stunden pro Tag	Führen Sie kontinuierlich 50 MiBps Nur-Lese-Schreibvorgänge pro TiB Speicher durch

Note

Amazon EFS bietet einen gemessenen Durchsatz von 1 MiBps für alle Dateisysteme, auch wenn die Basisrate niedriger ist.

Die Größe des Dateisystems, die für die Ermittlung der Basisrate und der Burst-Rate verwendet wird, ist die gemessene ValueInStandard-Größe, die über die [DescribeFileSystems](#)-API-Operation verfügbar ist.

Dateisysteme unter 1 TiB können Guthabenschriften bis zu einer Höhe von maximal 2,1 TiB erwerben. Dateisysteme über 1 TiB können Guthabenschriften bis zu einer Höhe von 2,1 TiB pro gespeichertem TiB erwerben. Dieses Verhalten bedeutet, dass Dateisysteme genügend Guthaben ansammeln können, um ein kontinuierliches Bursting über bis zu 12 Stunden auszuführen.

Einschränkungen beim Umschalten des Durchsatzes und beim Ändern der bereitgestellten Menge

Sie können den Durchsatzmodus eines vorhandenen Dateisystems wechseln und die Durchsatzmenge ändern. Nach dem Umschalten des Durchsatzmodus auf Bereitgestellter Durchsatz oder der Änderung der Menge des bereitgestellten Durchsatzes sind die folgenden Aktionen jedoch für einen Zeitraum von 24 Stunden eingeschränkt:

- Wechsel vom Modus „Bereitgestellter Durchsatz“ in den Durchsatzmodus „Elastic“ oder „Bursting“.
- Verringerung der Menge des bereitgestellten Durchsatzes.

Tipps zur Amazon-EFS-Leistung

Berücksichtigen Sie bei der Verwendung von Amazon EFS die folgenden Tipps zur Leistung:

Durchschnittliche I/O Größe

Die verteilte Struktur von Amazon EFS unterstützt einen hohen Grad an Verfügbarkeit, Beständigkeit und Skalierbarkeit. Diese verteilte Architektur führt zu einer geringfügigen Latenz bei den einzelnen Dateivorgängen. Aufgrund dieser Latenz pro Vorgang steigt der Gesamtdurchsatz im Allgemeinen mit steigender I/O Durchschnittsgröße, da sich der Overhead bei einer größeren Datenmenge amortisiert.

Optimierung von Workloads, die einen hohen Durchsatz und IOPS erfordern

Verwenden Sie für Workloads, die einen hohen Durchsatz und hohe IOPS erfordern, regionale Dateisysteme, die mit dem allgemeinen Leistungsmodus und dem elastischen Durchsatz konfiguriert sind.

Note

Um die maximale Anzahl an Lese-IOPS für Daten zu erreichen, auf die häufig zugegriffen wird, muss das Dateisystem Elastic Throughput verwenden.

Um ein Höchstmaß an Leistung zu erzielen, müssen Sie die Parallelisierung nutzen, indem Sie Ihre Anwendung oder Ihren Workload wie folgt konfigurieren.

1. Verteilen Sie den Workload gleichmäßig auf alle Clients und Verzeichnisse, wobei die Anzahl der Verzeichnisse mindestens der Anzahl der verwendeten Clients entspricht.
2. Minimieren Sie Konflikte, indem Sie einzelne Threads unterschiedlichen Datensätzen oder Dateien zuordnen.
3. Verteilen Sie die Arbeitslast auf 10 oder mehr NFS-Clients mit mindestens 64 Threads pro Client in einem einzigen Mount-Ziel.

Gleichzeitige Verbindungen

Sie können Amazon EFS-Dateisysteme auf bis zu Tausenden von Amazon EC2- und anderen AWS Recheninstanzen gleichzeitig mounten. Sie können einen höheren Durchsatz in Ihrem Dateisystem über alle Datenverarbeitungs-Instances hinweg erzielen, wenn Sie Ihre Anwendung über mehrere Instances hinweg parallelisieren können.

Anforderungsmodell

Wenn Sie asynchrone Schreibvorgänge zu Ihrem Dateisystem aktivieren, werden ausstehende Schreibvorgänge auf der Amazon-EC2-Instance gepuffert, bevor sie asynchron zu Amazon EFS geschrieben werden. Asynchrone Schreibvorgänge besitzen in der Regel niedrigere Latenzen. Bei der Ausführung asynchroner Schreibvorgänge verwendet der Kernel zusätzlichen Speicher zum Zwischenspeichern.

Ein Dateisystem, für das synchrone Schreibvorgänge aktiviert wurden oder das Dateien mittels einer Option öffnet, die den Zwischenspeicher umgeht (z. B. `O_DIRECT`), gibt synchrone Anforderungen an Amazon EFS aus. Jede Operation durchläuft einen Umlauf zwischen dem Client und Amazon EFS.

Note

Ihr Anforderungsmodell geht hinsichtlich Konsistenz (wenn Sie mehrere Amazon-EC2-Instances verwenden) und Geschwindigkeit Kompromisse ein. Die Verwendung synchroner Schreibvorgänge sorgt für mehr Datenkonsistenz, da jede Schreib Anforderungstransaktion abgeschlossen wird, bevor die nächste Anforderung verarbeitet wird. Durch die Verwendung asynchroner Schreibvorgänge wird der Durchsatz erhöht, da ausstehende Schreibvorgänge zwischengespeichert werden.

NFS-Client-Mount-Einstellungen

Überprüfen Sie, ob Sie die empfohlenen Mount-Optionen wie in [Mounting von EFS-Dateisystemen](#) und [Überlegungen zur Installation für Linux](#) beschrieben verwenden.

Beim Mounten Ihrer Dateisysteme auf Amazon-EC2-Instances unterstützt Amazon EFS die Network File System-Version 4.0 und 4.1 (NFSv4)-Protokolle. NFSv4.1 bietet im Vergleich zu NFSv4.0 (weniger als 1 000 Dateien pro Sekunde) eine bessere Leistung für parallel Lesevorgänge kleiner Dateien (mehr als 10 000 Dateien pro Sekunde). Für Amazon EC2 macOS-Instances, auf denen macOS Big Sur ausgeführt wird, wird nur NFSv4 2.0 unterstützt.

Verwenden Sie nicht die folgenden Mount-Optionen:

- `noac`, `actimeo=0`, `acregmax=0`, `acdirmax=0` – Diese Optionen deaktivieren den Attribut-Cache, was sich sehr negativ auf die Leistung auswirkt.
- `lookupcache=pos`, `lookupcache=none` – Diese Optionen deaktivieren den Dateinamen-Nachschlage-Cache, was sich sehr negativ auf die Leistung auswirkt.

- `fsc` – Diese Option aktiviert das lokale Zwischenspeichern von Dateien, ändert jedoch nichts an der Kohärenz des NFS-Cache und verringert auch nicht die Latenzen.

Note

Sie sollten Sie die Größe der Puffer für Lese- und Schreibpuffer für Ihren NFS-Client auf 1 MB erhöhen, wenn Sie Ihr Dateisystem mounten.

Optimierung der Leistung kleiner Dateien

Sie können die Leistung kleiner Dateien verbessern, indem Sie das erneute Öffnen von Dateien minimieren, die Parallelität erhöhen und Referenzdateien nach Möglichkeit bündeln.

- Minimieren Sie die Anzahl der Roundtrips zum Server.

Schließen Sie Dateien nicht unnötig, wenn Sie sie später in einem Workflow benötigen. Wenn Sie Dateideskriptoren geöffnet lassen, können Sie direkt auf die lokale Kopie im Cache zugreifen. Operationen zum Öffnen und Schließen von Dateien und zum Schließen von Metadaten können im Allgemeinen nicht asynchron oder über eine Pipeline ausgeführt werden.

Beim Lesen oder Schreiben kleiner Dateien sind die beiden zusätzlichen Roundtrips von Bedeutung.

Jeder Roundtrip (Datei öffnen, Datei schließen) kann genauso viel Zeit in Anspruch nehmen wie das Lesen oder Schreiben von Megabyte an Massendaten. Es ist effizienter, eine Eingabe- oder Ausgabedatei zu Beginn Ihres Datenverarbeitungsauftrag einmal zu öffnen und sie für die gesamte Dauer des Auftrags geöffnet zu lassen.

- Verwenden Sie Parallelität, um die Auswirkungen von Roundtrip-Zeiten zu reduzieren.
- Bündeln Sie Referenzdateien in einer `.zip`-Datei. Einige Anwendungen verwenden eine große Menge kleiner, meist schreibgeschützter Referenzdateien. Wenn Sie diese in einer `.zip`-Datei bündeln, können Sie viele Dateien in einem Roundtrip durch Öffnen und Schließen lesen.

Das `.zip`-Format ermöglicht den wahllosen Zugriff auf einzelne Dateien.

Optimieren der Verzeichnisleistung

Wenn ein Listing (`ls`) für sehr große Verzeichnisse (über 100k Dateien) durchgeführt wird, die gleichzeitig geändert werden, können Linux-NFS-Clients hängen bleiben und keine Antwort zurückgeben. Dieses Problem wurde in Kernel 5.11 behoben, der auf die Amazon-Linux 2-Kernel 4.14, 5.4 und 5.10 portiert wurde.

Wir empfehlen, die Anzahl der Verzeichnisse in Ihrem Dateisystem möglichst auf weniger als 10 000 zu beschränken. Verwenden Sie so weit wie möglich verschachtelte Unterverzeichnisse.

Vermeiden Sie beim Auflisten eines Verzeichnisses die Angabe von Dateiattributen, wenn diese nicht erforderlich sind, da sie nicht im Verzeichnis selbst gespeichert sind.

Optimierung der NFS-Größe von `read_ahead_kb`

Das `read_ahead_kb`-NFS-Attribut definiert die Anzahl der Kilobyte, die der Linux-Kernel bei einem sequentiellen Lesevorgang vorab lesen oder vorab abrufen muss.

Bei Linux-Kernel-Versionen vor 5.4.* wird der Wert `read_ahead_kb` durch Multiplikation von `NFS_MAX_READAHEAD` mit dem Wert für `rsize` (der vom Client konfigurierten Lesepuffergröße, die in den Mount-Optionen festgelegt wurde) festgelegt. Bei Verwendung der [empfohlenen Mount-Optionen](#) setzt diese Formel `read_ahead_kb` auf 15 MB.

Note

Ab den Linux-Kernel-Versionen 5.4.* verwendet der Linux-NFS-Client einen `read_ahead_kb`-Standardwert von 128 KB. Wir empfehlen, diesen Wert auf 15 MB zu erhöhen.

Die Amazon-EFS-Mountinghilfe, die in `amazon-efs-utils`-Version 1.33.2 und höher verfügbar ist, ändert den `read_ahead_kb`-Wert nach dem Mounten des Dateisystems automatisch auf $15 * rsize$ oder 15 MB.

Wenn Sie bei Linux-Kernel 5.4 oder höher die Mountinghilfe nicht zum Mounten Ihrer Dateisysteme verwenden, sollten Sie erwägen, `read_ahead_kb` manuell auf 15 MB einzustellen, um die Leistung zu verbessern. Nach dem Mounten des Dateisystems können Sie den `read_ahead_kb`-Wert mithilfe des folgenden Befehls zurücksetzen. Ersetzen Sie die folgenden Werte, bevor Sie diesen Befehl verwenden:

- Ersetzen Sie *read-ahead-value-kb* durch die gewünschte Größe in Kilobyte.
- Ersetzen Sie *efs-mount-point* durch den Mountingpunkt des Dateisystems.

```
device_number=$(stat -c '%d' efs-mount-point)  
((major = ($device_number & 0xFFF00) >> 8))  
((minor = ($device_number & 0xFF) | (($device_number >> 12) & 0xFFF00)))  
sudo bash -c "echo read-ahead-value-kb > /sys/class/bdi/$major:$minor/read_ahead_kb"
```

Im Folgenden wird beispielsweise die `read_ahead_kb`-Größe auf 1 MB festgelegt.

```
device_number=$(stat -c '%d' efs)  
((major = ($device_number & 0xFFF00) >> 8))  
((minor = ($device_number & 0xFF) | (($device_number >> 12) & 0xFFF00)))  
sudo bash -c "echo 15000 > /sys/class/bdi/$major:$minor/read_ahead_kb"
```

Behebung von Amazon EFS-Leistungsproblemen

Wenn Sie Probleme mit Amazon EFS haben, die sich nur schwer beheben lassen, sollten Sie sicherstellen, dass Sie einen aktuellen Linux-Kernel verwenden. Wenn Sie eine Linux-Unternehmensdistribution verwenden, empfehlen wir Folgendes:

- Amazon Linux 2 mit Kernel 4.3 oder neuer
- Amazon Linux 2015.09 oder neuer
- RHEL 7.3 oder neuer
- Alle Versionen von Ubuntu 16.04
- Ubuntu 14.04 mit Kernel 3.13.0-83 oder neuer
- SLES 12 Sp2 oder höher

Wenn Sie eine andere Verteilung oder einen benutzerdefinierten Kernel verwenden, empfehlen wir Kernel-Version 4.3 oder neuer.

Note

RHEL 6.9 könnte für bestimmte Workloads suboptimal sein aufgrund von [Schlechte Leistung beim Öffnen vieler Dateien gleichzeitig](#).

Themen

- [Ein EFS-Dateisystem kann nicht erstellt werden](#)
- [Zugriff auf zulässige Dateien im NFS-Dateisystem verweigert](#)
- [Fehler beim Zugriff auf die Amazon-EFS-Konsole](#)
- [Amazon EC2-Instance hängt sich auf](#)
- [Anwendung, die große Datenmengen schreibt, bleibt hängen](#)
- [Schlechte Leistung beim Öffnen vieler Dateien gleichzeitig](#)
- [Benutzerdefinierte NFS-Einstellungen verursachen Schreibverzögerungen](#)
- [Die Erstellung von Sicherungen mit Oracle Recovery Manager ist langsam](#)

Ein EFS-Dateisystem kann nicht erstellt werden

Eine Anfrage zur Erstellung eines EFS-Dateisystems schlägt mit der folgenden Meldung fehl:

```
User: arn:aws:iam::111122223333:user/username is not authorized to
perform: elasticfilesystem:CreateFileSystem on the specified resource.
```

Maßnahme

Überprüfen Sie Ihre AWS Identity and Access Management (IAM-) Richtlinie, um sicherzustellen, dass Sie berechtigt sind, EFS-Dateisysteme mit den angegebenen Ressourcenbedingungen zu erstellen. Weitere Informationen finden Sie unter [Identitäts- und Zugriffsmanagement für Amazon EFS](#).

Zugriff auf zulässige Dateien im NFS-Dateisystem verweigert

Wenn ein Benutzer, dem mehr als 16 Zugriffsgruppen IDs (GIDs) zugewiesen sind, versucht, einen Vorgang in einem NFS-Dateisystem auszuführen, kann ihm der Zugriff auf zulässige Dateien im Dateisystem verweigert werden. [Dieses Problem tritt auf, weil das NFS-Protokoll maximal 16 GIDs pro Benutzer unterstützt und alle weiteren Daten aus der NFS-Client-Anfrage gekürzt GIDs werden, wie in RFC 5531 definiert.](#)

Maßnahme

Strukturieren Sie Ihre NFS-Benutzer- und Gruppenzuordnungen neu, sodass jedem Benutzer nicht mehr als 16 Zugriffsgruppen () zugewiesen werden. GIDs

Fehler beim Zugriff auf die Amazon-EFS-Konsole

Dieser Abschnitt beschreibt Fehler, die beim Zugriff auf die Amazon-EFS-Management Console auftreten können.

Fehler bei der Authentifizierung der Anmeldeinformationen für **ec2:DescribeVPCs**

Die folgende Fehlermeldung wird beim Zugriff auf die Amazon-EFS-Konsole angezeigt:

```
AuthFailure: An error occurred authenticating your credentials for ec2:DescribeVPCs.
```

Dieser Fehler weist darauf hin, dass Ihre Anmeldeinformationen beim Amazon EC2-Service nicht erfolgreich authentifiziert wurden. Die Amazon-EFS-Konsole ruft den Amazon EC2-Service in Ihrem Namen auf, wenn Sie EFS-Dateisysteme in der von Ihnen ausgewählten VPC erstellen.

Maßnahme

Stellen Sie sicher, dass die Uhrzeit auf dem Client, der auf die Amazon-EFS-Konsole zugreift, korrekt eingestellt ist.

Amazon EC2-Instance hängt sich auf

Eine Amazon EC2-Instance kann hängen bleiben, weil Sie ein Mounting-Ziel für ein Dateisystem gelöscht haben, ohne das Dateisystem vorher auszuhängen.

Maßnahme

Bevor Sie ein Dateisystem-Mounting-Ziel löschen, heben Sie das Mounting des Dateisystems auf. Weitere Informationen zum Unmounten Ihres Amazon-EFS-Dateisystems finden Sie unter [Aufheben des Mountings von Dateisystemen](#).

Anwendung, die große Datenmengen schreibt, bleibt hängen

Eine Anwendung, die eine große Menge an Daten in Amazon EFS schreibt, hängt sich auf und verursacht einen Neustart der Instance.

Maßnahme

Wenn eine Anwendung zu lange braucht, um alle Daten in Amazon EFS zu schreiben, kann es sein, dass Linux neu startet, weil es den Anschein hat, dass der Vorgang nicht mehr

reagiert. Diese Verhaltensweise wird von zwei Kernel-Konfigurationsparametern definiert, `kernel.hung_task_panic` und `kernel.hung_task_timeout_secs`.

Im folgenden Beispiel wird der Status des hängengebliebenen Prozesses vom `ps`- Befehl mit `D` vor dem Neustart der Instance gemeldet; dies bedeutet, dass der Prozess auf einen E/A-Vorgang wartet.

```
$ ps aux | grep large_io.py
root 33253 0.5 0.0 126652 5020 pts/3 D+ 18:22 0:00 python large_io.py
/efs/large_file
```

Um einen Neustart zu verhindern, verlängern Sie den Timeout-Zeitraum, oder deaktivieren Sie die Kernel-Panik, wenn eine Aufgabe hängenbleibt. Der folgende Befehl deaktiviert die Kernel-Panik für hängengebliebene Aufgaben in den meisten Linux-Systemen.

```
$ sudo sysctl -w kernel.hung_task_panic=0
```

Schlechte Leistung beim Öffnen vieler Dateien gleichzeitig

Bei Anwendungen, die mehrere Dateien parallel öffnen, wird die erwartete Leistungssteigerung durch I/O Parallelisierung nicht erreicht.

Maßnahme

Dieses Problem tritt auf Network File System Version 4 (NFSv4) -Clients und auf RHEL 6-Clients auf, die NFSv4 .1 verwenden, weil diese NFS-Clients die NFS OPEN- und CLOSE-Operationen serialisieren. Verwenden Sie das NFS-Protokoll Version 4.1 und eine der vorgeschlagenen [Linux-Distributionen](#), die dieses Problem nicht haben.

Wenn Sie NFSv4 .1 nicht verwenden können, beachten Sie, dass der Linux 4.0-Client NFSv4 Öffnungs- und Schließenanfragen nach Benutzer-ID und Gruppe serialisiert. IDs Diese Serialisierung geschieht auch dann, wenn mehrere Prozesse oder mehrere Threads gleichzeitig Anforderungen ausgeben. Der Client sendet jeweils nur einen Öffnungs- oder Schließvorgang an einen NFS-Server, wenn alle übereinstimmen. IDs Um diese Probleme zu umgehen, können Sie eine der folgenden Aktionen durchführen:

- Sie können jeden Vorgang von einer anderen Benutzer-ID auf derselben Amazon EC2-Instance ausführen.
- Sie können IDs den Benutzer für alle offenen Anfragen unverändert lassen und IDs stattdessen die Gruppe ändern.

- Sie können jeden Vorgang auf einer separaten Amazon EC2-Instance ausführen.

Benutzerdefinierte NFS-Einstellungen verursachen Schreibverzögerungen

Sie haben benutzerdefinierte NFS-Client-Einstellungen, und es dauert bis zu drei Sekunden, bis eine Amazon EC2-Instance einen Schreibvorgang auf einem Dateisystem von einer anderen Amazon EC2-Instance sieht.

Maßnahme

Wenn dieses Problem auftritt, können Sie sie auf eine der folgenden Weisen lösen:

- Wenn der NFS-Client auf der Amazon EC2-Instance, die die Daten liest, das Attribut-Caching aktiviert hat, unmounten Sie Ihr Dateisystem. Mounten Sie es dann erneut mit der Option `noac`, um die Attributzwischenspeicherung zu deaktivieren. Das Zwischenspeichern von Attributen in NFSv4 .1 ist standardmäßig aktiviert.

Note

Die Deaktivierung der clientseitigen Zwischenspeicherung kann möglicherweise die Leistung Ihrer Anwendung beeinträchtigen.

- Sie können auch bei Bedarf den Attributzwischenspeicher leeren, indem Sie eine mit den NFS-Vorgängen kompatible Computersprache verwenden. Zu diesem Zweck können Sie ACCESS-Vorgangsanforderung unmittelbar vor einer Leseanforderung senden.

Beispielsweise können Sie mit der Programmiersprache Python den folgenden Aufruf konstruieren.

```
# Does an NFS ACCESS procedure request to clear the attribute cache, given a path to
the file
import os
os.access(path, os.W_OK)
```

Die Erstellung von Sicherungen mit Oracle Recovery Manager ist langsam

Die Erstellung von Sicherungen mit Oracle Recovery Manager kann langsam sein, wenn Oracle Recovery Manager für 120 Sekunden pausiert, bevor er einen Sicherungsauftrag startet.

Maßnahme

Wenn dieses Problem auftritt, deaktivieren Sie Oracle Direct NFS, wie unter [Enabling and Disabling Direct NFS Client Control of NFS](#) im Oracle Help Center beschrieben.

 Note

Amazon EFS unterstützt kein Oracle Direct NFS.

Beheben von AMI- und Kernel-Problemen

Nachfolgend finden Sie Informationen zur Fehlerbehebung bei Problemen im Zusammenhang mit bestimmten Amazon Machine Image (AMI) oder Kernel-Versionen bei der Verwendung von Amazon EFS von einer Amazon EC2-Instance aus.

Themen

- [Eigentümerschaft kann nicht geändert werden](#)
- [Aufgrund des Client-Bug wiederholt das Dateisystem Vorgänge immer wieder](#)
- [Blockierter Client](#)
- [Das Auflisten von Dateien in einem großen Verzeichnis dauert zu lange](#)

Eigentümerschaft kann nicht geändert werden

Sie können den Besitzer eines nicht file/directory mit dem chown Linux-Befehl ändern.

Kernel-Versionen mit diesem Bug

2.6.32

Maßnahme

Sie können dieses Problem lösen, indem Sie folgende Schritte ausführen:

- Wenn Sie chown für den einmaligen Einrichtungsschritt durchführen, der zur Änderung der Eigentümerschaft des EFS-Stammverzeichnisses erforderlich ist, können Sie den Befehl chown von einer Instance ausführen, auf der ein neuer Kernel ausgeführt wird. Verwenden Sie zum Beispiel die neueste Version von Amazon Linux.
- Wenn chown Teil Ihrer Produktionsabläufe ist, müssen Sie die Kernel-Version aktualisieren, um chown zu verwenden.

Aufgrund des Client-Bug wiederholt das Dateisystem Vorgänge immer wieder

Aufgrund eines Client-Bugs wiederholt ein Dateisystem ständig Vorgänge.

Maßnahme

Aktualisieren Sie die Client-Software auf die neueste Version.

Blockierter Client

Ein Client ist blockiert.

Kernel-Versionen mit diesem Bug

- CentOS-7 mit Kernel Linux 3.10.0-229.20.1.el7.x86_64
- Ubuntu 15.10 mit Kernel Linux 4.2.0-18-generic

Maßnahme

Führen Sie eine der folgenden Aktionen aus:

- Upgrade auf eine neuere Kernel-Version. Für CentOS-7 enthält Kernel-Version Linux 3.10.0-327 oder neuer die Fehlerbehebung.
- Downgrade auf eine ältere Kernel-Version.

Das Auflisten von Dateien in einem großen Verzeichnis dauert zu lange

Dies kann geschehen, wenn das Verzeichnis geändert wird, während Ihr NFS-Client das Verzeichnis durchläuft, um den Listenvorgang abzuschließen. Wenn der NFS-Client feststellt, dass die Inhalte des Verzeichnisses während dieses Vorgangs geändert wurden, beginnt er mit dem Vorgang von vorn. Dadurch kann es sein, dass der ls-Befehl für ein großes Verzeichnis mit häufig geänderten Dateien zu lange dauert.

Kernel-Versionen mit diesem Bug

CentOS- und RHEL-Kernel-Versionen unter 2.6.32-696.el6

Maßnahme

Um dieses Problem zu lösen, führen Sie ein Upgrade auf eine neuere Kernel-Version durch.

Daten in Amazon EFS sichern und replizieren

Um Ihre Daten zu schützen, sichert Amazon EFS automatisch Ihre EFS-Dateisysteme. Für mehr Stabilität und Datenschutz können Sie Ihr EFS-Dateisystem in einem AWS-Region replizieren. Durch das Sichern und Replizieren Ihrer EFS-Dateisysteme stellen Sie sicher, dass Sie weiterhin Operationen oder Dienste bereitstellen können, falls etwas mit den EFS-Dateisystemdaten passiert. Zum Beispiel im Fall von Datenbeschädigung oder Datenverlust.

Themen

- [Sicherung von EFS-Dateisystemen](#)
- [EFS-Dateisysteme replizieren](#)

Sicherung von EFS-Dateisystemen

Amazon EFS ist nativ integriert mit AWS Backup, ein vollständig verwalteter, richtlinienbasierter Service, mit dem Sie Backup-Richtlinien zum Schutz Ihrer Daten in Amazon EFS erstellen und verwalten können. Dateisysteme, die Sie mit der Amazon EFS-Konsole erstellen, werden AWS Backup standardmäßig automatisch gesichert. Wenn Sie die CLI oder API verwenden, um ein Dateisystem zu erstellen, sind automatische Backups standardmäßig nur für One Zone-Dateisysteme aktiviert.

Mit AWS Backup Amazon EFS können Sie die folgenden Aktionen ausführen:

- Verwalten Sie die automatische Planung und Aufbewahrung von Backups, indem Sie Backup-Pläne konfigurieren. Sie geben die Backup-Häufigkeit, den Zeitpunkt der Sicherung und die Aufbewahrungsdauer von Backups sowie eine Lebenszyklusrichtlinie für Backups an.
- Stellen Sie Backups von Amazon EFS-Daten wieder her. Sie können Dateisystemdaten entweder in einem neuen oder einem vorhandenen Dateisystem wiederherstellen. Sie können auch wählen, ob Sie eine vollständige Wiederherstellung oder eine Wiederherstellung auf Elementebene durchführen möchten.

Weitere Informationen zur Verwendung finden Sie AWS Backup unter [Erste Schritte mit AWS Backup](#) im AWS Backup Entwicklerhandbuch.

Themen

- [Wie AWS Backup funktioniert mit Amazon EFS](#)
- [Erforderliche IAM-Berechtigungen](#)

- [Backup-Leistung](#)
- [Verwaltung automatischer Backups von EFS-Dateisystemen](#)

Wie AWS Backup funktioniert mit Amazon EFS

Dateisysteme, die Sie mit der Amazon EFS-Konsole erstellen, werden AWS Backup standardmäßig automatisch gesichert. Wenn Sie die CLI oder API verwenden, um ein Dateisystem zu erstellen, sind automatische Backups standardmäßig nur für One Zone-Dateisysteme aktiviert. Sie können automatische Backups aktivieren, nachdem Sie Ihr EFS-Dateisystem mithilfe der API AWS CLI oder erstellt haben. Der standardmäßige EFS-Backupplan verwendet die AWS Backup empfohlenen Einstellungen für automatische Backups — tägliche Backups mit einer Aufbewahrungsfrist von 35 Tagen. Die mit dem standardmäßigen EFS-Backupplan erstellten Backups werden in einem standardmäßigen EFS-Backup-Tresor gespeichert, der ebenfalls von Amazon EFS in Ihrem Namen erstellt wird. Der Standard-Sicherungsplan und Sicherungstresor kann normalerweise nicht gelöscht werden.

Alle Daten in einem EFS-Dateisystem werden gesichert, unabhängig davon, in welcher Speicherklasse sich die Daten befinden. Es fallen keine Kosten für den Datenzugriff an, wenn ein EFS-Dateisystem mit aktivierter Lebenszyklusverwaltung und Daten in der Infrequent Access (IA)- oder Archive-Speicherklasse gesichert wird. Bei der Wiederherstellung eines Wiederherstellungspunkts werden alle Dateien in der Standardspeicherklasse wiedergestellt.

Inkrementelle Sicherungen

AWS Backup führt inkrementelle Backups von EFS-Dateisystemen durch. Während der ersten Sicherung wird eine Kopie des gesamten Dateisystems erstellt. Bei nachfolgenden Sicherungen dieses Dateisystems werden nur Dateien und Verzeichnisse kopiert, die geändert, hinzugefügt oder entfernt wurden. AWS Backup behält bei jeder inkrementellen Sicherung die erforderlichen Referenzdaten bei, um eine vollständige Wiederherstellung zu ermöglichen. Durch dieses Verfahren wird die zum Vollenden der Sicherung erforderliche Zeit verringert und es werden Speicherkosten eingespart, weil keine Datenduplikate angelegt werden.

Backup-Konsistenz

Amazon EFS ist so konzipiert, dass es hochverfügbar ist. Sie können auf Ihre EFS-Dateisysteme zugreifen und diese ändern, während Ihre Sicherung in erfolgt AWS Backup. Wenn Sie jedoch während der Sicherung Änderungen an Ihrem Dateisystem vornehmen, können Unregelmäßigkeiten, wie z.°B. Duplikate, Verzerrungen oder Datenverlust auftreten. Diese Änderungen umfassen das

Schreiben, Umbenennen, Verschieben oder Löschen. Um konsistente Sicherungen zu garantieren, empfehlen wir, dass Sie Anwendungen oder Prozesse, die Änderungen an dem Dateisystem vornehmen, für die Dauer des Sicherungsvorgangs anhalten. Oder Sie planen Ihre Sicherungen so, dass sie in Zeiten durchgeführt werden, in denen das Dateisystem nicht geändert wird.

Zeitfenster für den Abschluss der Sicherung

Sie können optional ein bestimmtes Fertigstellungsfenster für eine Sicherung angeben. Dieses Fenster definiert den Zeitraum, in dem eine Sicherung ausgeführt werden muss. Wenn Sie ein Fenster angeben, berücksichtigen Sie die erwartete Performance sowie die Größe und Zusammensetzung Ihres Dateisystems. Auf diese Weise stellen Sie sicher, dass Ihre Sicherung während des Fensters fertiggestellt werden kann.

Sicherungen, die nicht während des angegebenen Fensters fertiggestellt werden können, werden mit dem Status „unvollständig“ gekennzeichnet. Wird beim nächsten geplanten Backup an der Stelle AWS Backup fortgesetzt, an der es unterbrochen wurde. Sie können den Status all Ihrer Backups in der AWS Backup Management Console einsehen.

Bewährte Methoden für die Zeitfenster zum Abschluss von Backups

Beachten Sie bei der Konfiguration von Zeitfenstern für den Abschluss von Backups für Ihre EFS-Dateisysteme die folgenden bewährten Methoden, um zuverlässige Sicherungsvorgänge sicherzustellen:

- Überwachen Sie die Dauer der Sicherung — Wenn Sie feststellen, dass Ihre Backups einen längeren Zeitraum in Anspruch nehmen (z. B. 20 Tage oder mehr), sollten Sie das Abschlussfenster auf 3 Tage verkürzen. Dieser Ansatz AWS Backup ermöglicht es, mehrere kürzere Teilsicherungen durchzuführen, anstatt zu versuchen, das gesamte Backup in einem einzigen erweiterten Fenster abzuschließen.
- Berücksichtigen Sie die Eigenschaften des Dateisystems — Große Dateisysteme mit Millionen von Dateien oder großen Datenmengen können von kürzeren Abschlussfenstern profitieren. Die Empfehlung für ein Zeitfenster von 3 Tagen trägt dazu bei, dass die Sicherungsvorgänge erfolgreich abgeschlossen und bei Bedarf effizient wieder aufgenommen werden können.
- Planen Sie den inkrementellen Fortschritt ein — Wenn Sie kürzere Zeitfenster für die Fertigstellung verwenden, AWS Backup wird das System beim nächsten geplanten Backup automatisch an der Stelle fortgesetzt, an der es unterbrochen wurde. Dieser inkrementelle Ansatz kann bei großen Dateisystemen zuverlässiger sein als der Versuch, alles in einem einzigen langen Fenster abzuschließen.

- Testen Sie Ihre Backup-Strategie — Bevor Sie sich in der Produktion auf Ihre Backup-Konfiguration verlassen, testen Sie sie anhand Ihrer spezifischen Dateisystemgröße und -eigenschaften, um das optimale Abschlussfenster für Ihren Anwendungsfall zu ermitteln.

Note

Kürzere Zeitfenster für die Fertigstellung bedeuten nicht, dass die AWS Backup Fähigkeit, große Dateisysteme zu verarbeiten, eingeschränkt ist. Stattdessen bieten sie einen besser vorhersehbaren und besser verwaltbaren Backup-Prozess, der sich an unterschiedliche Dateisystemgrößen und Netzwerkbedingungen anpassen kann.

On-Demand-Sicherungen

Mit AWS Backup können Sie bei Bedarf eine einzelne Ressource in einem Backup-Tresor speichern. Im Gegensatz zu geplanten Sicherungen müssen Sie keinen Sicherungsplan erstellen, um eine On-Demand-Sicherung zu initiieren. Sie können Ihrer Sicherung nach wie vor einen Lebenszyklus zuweisen, der den Wiederherstellungspunkt automatisch in das selten genutzte Speicher-Tier verschiebt und beim Löschen darauf hinweist.

Darüber hinaus AWS Backup werden Daten nur für Daten, die im letzten warmen Backup nicht mehr vorhanden sind, automatisch in einen kalten Speicher übertragen. Ihr Dateisystem hat beispielsweise 100 Dateien, wenn Sie eine Sicherung erstellen, und Sie löschen zwei Dateien am Tag nach der Erstellung der Sicherung (100 Dateien — 2 Dateien = 98 Dateien am zweiten Tag). Wenn Sie die Daten in einen kalten Speicher übertragen, werden nur die beiden gelöschten Dateien in den kalten Speicher verschoben, und die verbleibenden 98 Dateien werden als warmer Speicher abgerechnet.

Gleichzeitige Sicherungen

AWS Backup beschränkt Backups auf ein gleichzeitiges Backup pro Ressource. Daher können geplante oder On-Demand-Sicherungen fehlschlagen, wenn bereits ein Sicherungsauftrag ausgeführt wird. Weitere Informationen zu AWS Backup Grenzwerten finden Sie unter [AWS Backup Kontingente](#) im AWS Backup Entwicklerhandbuch.

Löschungen von Backup

Die Standard-Zugriffsrichtlinie für den EFS-Sicherungstresore ist so eingestellt, dass das Löschen von Wiederherstellungspunkten verweigert wird. Um bestehende Sicherungen Ihrer EFS-

Dateisysteme zu löschen, müssen Sie die Tresorzugriffsrichtlinie ändern. Wenn Sie versuchen, einen EFS-Wiederherstellungspunkt zu löschen, ohne die Tresorzugriffsrichtlinie zu ändern, wird die folgende Fehlermeldung angezeigt:

```
"Access Denied: Insufficient privileges to perform this action. Please consult with the account administrator for necessary permissions."
```

Um die Standard-Zugriffsrichtlinie für den Sicherungstresor zu bearbeiten, müssen Sie über die erforderlichen Berechtigungen zum Bearbeiten von Richtlinien verfügen. Weitere Informationen finden Sie unter [Erlauben aller IAM-Aktionen \(Administratorzugriff\)](#) im IAM-Benutzerhandbuch.

Erforderliche IAM-Berechtigungen

AWS Backup erstellt in Ihrem Namen eine dienstbezogene Rolle in Ihrem Konto. Diese Rolle hat die erforderlichen Berechtigungen für die Durchführung von Amazon-EFS-Backups.

Sie können die Aktionen `elasticfilesystem:backup` und `elasticfilesystem:restore` verwenden, um einer IAM-Entität (z. B. Benutzer, Gruppe oder Rolle) die Erstellung oder Wiederherstellung von Sicherungen eines EFS-Dateisystems zu gewähren oder zu verweigern. Sie können diese Aktionen in einer Dateisystemrichtlinie oder in einer identitätsbasierten IAM-Richtlinie verwenden. Weitere Informationen erhalten Sie unter [Identitäts- und Zugriffsmanagement für Amazon EFS](#) und [Verwendung von IAM zur Steuerung des Zugriffs auf Dateisysteme](#).

Backup-Leistung

Im Allgemeinen können Sie mit den folgenden Sicherungs- und Wiederherstellungsraten rechnen AWS Backup. Bei einigen Workloads, z. B. solchen, die eine große Datei oder ein großes Verzeichnis enthalten, können die Raten geringer sein.

- Backup-Rate von 2.000 Dateien pro Sekunde oder 400 Megabyte pro Sekunde (MBps), je nachdem, welcher Wert langsamer ist.
- Wiederherstellungsrate von 1.500 Dateien pro Sekunde oder 200 MBps, je nachdem, welcher Wert langsamer ist.

Die maximale Dauer für einen Sicherungsvorgang AWS Backup beträgt 30 Tage.

Durch die Verwendung werden AWS Backup keine angesammelten Summen-Credits verbraucht und sie wird auch nicht auf die Obergrenzen für Dateioperationen im allgemeinen Leistungsmodus angerechnet. Weitere Informationen finden Sie unter [Kontingente für Amazon-EFS-Dateisysteme](#).

Verwaltung automatischer Backups von EFS-Dateisystemen

Wenn Sie ein Dateisystem mit der Amazon-EFS-Konsole erstellen, sind automatische Sicherungen standardmäßig aktiviert. Sie können automatische Backups aktivieren, nachdem Sie Ihr Dateisystem mithilfe der API AWS CLI oder erstellt haben.

Sie können die Standardeinstellungen für den Backup-Plan über die AWS Backup Konsole bearbeiten. Weitere Informationen finden Sie im AWS Backup Entwicklerhandbuch unter [Backup-Pläne verwalten](#). Sie können alle Ihre automatischen Sicherungen anzeigen und die Standardeinstellungen für den EFS-Sicherungsplan mithilfe der [AWS Backup -Konsole](#) bearbeiten.

Amazon EFS wendet den `aws:elasticfilesystem:default-backup-System-Tag`-Schlüssel mit einem Wert von `enabled` auf EFS-Dateisysteme an, wenn automatische Sicherungen aktiviert sind.

Nachdem Sie ein Dateisystem erstellt haben, können Sie automatische Backups mithilfe der Konsole, der oder der AWS CLI EFS-API ein- oder ausschalten.

Verwenden der Konsole

1. Öffnen Sie die Amazon Elastic File System-Konsole unter <https://console.aws.amazon.com/efs/>.
2. Wählen Sie auf der Seite Dateisysteme das Dateisystem aus, für das Sie automatische Sicherungen ein- oder ausschalten möchten, und zeigen Sie die Seite mit den Dateisystemdetails an.
3. Wählen Sie Bearbeiten im Bereich der Allgemeinen Einstellungen aus.
4.
 - Um automatische Sicherungen zu aktivieren, wählen Sie Automatische Backups aktivieren aus.
 - Um automatische Sicherungen zu deaktivieren, wählen Sie Automatische Backups deaktivieren aus.
5. Wählen Sie Änderungen speichern aus.

Mit dem AWS CLI

- Verwenden Sie den `put-backup-policy`-CLI-Befehl (der entsprechende API-Vorgang ist [PutBackupPolicy](#)), um automatische Sicherungen für ein vorhandenes Dateisystem ein- oder auszuschalten.
 - Verwenden Sie den folgenden Befehl, um automatische Sicherungen zu aktivieren.

```
$ aws efs put-backup-policy --file-system-id fs-01234567 \  
--backup-policy Status="ENABLED"
```

Amazon EFS reagiert mit der neuen Backup-Richtlinie.

```
{  
  "BackupPolicy": {  
    "Status": "ENABLING"  
  }  
}
```

- Verwenden Sie den folgenden Befehl, um automatische Sicherungen zu deaktivieren.

```
$ aws efs put-backup-policy --file-system-id fs-01234567 \  
--backup-policy Status="DISABLED"
```

Amazon EFS reagiert mit der neuen Backup-Richtlinie.

```
{  
  "BackupPolicy": {  
    "Status": "DISABLING"  
  }  
}
```

EFS-Dateisysteme replizieren

Für mehr Stabilität und Datenschutz können Sie Ihr EFS-Dateisystem in einem AWS-Region replizieren. Wenn Sie die Replikation auf einem EFS-Dateisystem aktivieren, repliziert Amazon EFS die Daten und Metadaten auf dem Quelldateisystem automatisch und transparent in ein Zieldateisystem. Im Katastrophenfall oder bei der Durchführung von Übungen am Spieltag können Sie ein Failover auf Ihr Replikat-Dateisystem durchführen. Um den Betrieb wieder aufzunehmen, können Sie dann auf das primäre Dateisystem zurückgreifen.

Um den Prozess der Erstellung des Zieldateisystems und dessen Synchronisation mit dem Quelldateisystem zu verwalten, verwendet Amazon EFS eine Replikationskonfiguration.

Nachdem Sie die Replikationskonfiguration erstellt haben, synchronisiert Amazon EFS die Quell- und Zieldateisysteme automatisch. Am Quelldateisystem vorgenommene Änderungen werden nicht

point-in-time konsistent in das Zielsystem übertragen. Stattdessen werden sie auf der Grundlage der Uhrzeit der letzten Synchronisierung für die Replikation übertragen. Die Uhrzeit der letzten Synchronisierung gibt an, wann die letzte erfolgreiche Synchronisierung zwischen der Quelle und dem Ziel abgeschlossen wurde. Änderungen, die zum Zeitpunkt der letzten Synchronisierung an Ihrem Quellsystem vorgenommen wurden, werden in das Zielsystem repliziert, während Änderungen, die nach der letzten Synchronisierung am Quellsystem vorgenommen wurden, möglicherweise nicht repliziert werden. Weitere Informationen finden Sie unter [Replikationsdetails anzeigen](#).

Die Replikation ist in allen Bereichen verfügbar, AWS-Regionen in denen Amazon EFS verfügbar ist. Um ein EFS-Dateisystem in einer Region zu replizieren, die standardmäßig deaktiviert ist, müssen Sie sich zunächst für die Region anmelden. Weitere Informationen finden Sie im AWS Allgemeinen Referenzhandbuch unter „[Geben Sie an, was AWS-Regionen Ihr Konto verwenden kann](#)“. Wenn Sie sich später von einer Region abmelden, unterbricht Amazon EFS alle Replikationsaktivitäten für die Region. Um die Replikationsaktivitäten für die Region wieder aufzunehmen, melden Sie sich AWS-Region erneut für die an.

 Note

Die Replikation unterstützt die Verwendung von Tags für die attributbasierte Zugriffskontrolle (ABAC) nicht.

Topics

- [Kosten](#)
- [Replikationsleistung](#)
- [Erforderliche IAM-Berechtigungen](#)
- [Konfiguration der Replikation auf ein neues EFS-Dateisystem](#)
- [Konfiguration der Replikation auf ein vorhandenes EFS-Dateisystem](#)
- [Kontenübergreifende AWS Replikation von EFS-Dateisystemen](#)
- [Replikationsdetails anzeigen](#)
- [Löschen von Replikationskonfigurationen](#)
- [Das Replikat wird verwendet](#)

Kosten

Um die Replikation zu erleichtern, erstellt Amazon EFS versteckte Verzeichnisse und Metadaten im Zieldateisystem. Dies entspricht ungefähr 12 Mebibyte (MiB) an gemessenen Daten, die Ihnen in Rechnung gestellt werden. Weitere Informationen über die Ermittlung des Dateisystemspeichers finden Sie unter [Wie Amazon EFS Dateisystem- und Objektgrößen meldet](#).

Replikationsleistung

Wenn Sie während des Failback-Prozesses neue Replikationen erstellen oder die Richtung vorhandener Replikationen umkehren, führt Amazon EFS eine erste Synchronisierung durch, die eine Reihe von einmaligen Einrichtungsaktionen zur Unterstützung der Replikation umfasst. Auf replizierte Daten kann im Zieldateisystem erst zugegriffen werden, nachdem die erste Synchronisierung abgeschlossen ist. Wie lange es dauert, bis die erste Synchronisierung abgeschlossen ist, hängt von Faktoren wie der Größe des Quelldateisystems und der Anzahl der darin enthaltenen Dateien ab.

Nach Abschluss der ersten Replikation behält Amazon EFS für die meisten Dateisysteme ein Recovery Point Objective (RPO) von 15 Minuten bei. Wenn das Quelldateisystem jedoch Dateien enthält, die sich sehr häufig ändern und entweder mehr als 100 Millionen Dateien oder Dateien mit einer Größe von mehr als 100 GB enthalten, kann die Replikation länger als 15 Minuten dauern. Hinweise zur Überwachung, wann die letzte Replikation erfolgreich abgeschlossen wurde, finden Sie unter [Replikationsdetails anzeigen](#).

Sie können mithilfe der Konsole, der AWS Command Line Interface (AWS CLI), der API und Amazon überwachen, wann die letzte erfolgreiche Synchronisierung stattgefunden hat CloudWatch. Verwenden Sie in CloudWatch die [TimeSinceLastSync](#)EFS-Metrik. Weitere Informationen finden Sie unter [Replikationsdetails anzeigen](#).

Erforderliche IAM-Berechtigungen

Amazon EFS verwendet entweder die mit dem EFS-Dienst verknüpfte Rolle mit dem Namen `AWSServiceRoleForAmazonElasticFileSystem` oder die von Ihnen angegebene IAM-Rolle, um die Replikation zwischen den Quell- und Zieldateisystemen zu synchronisieren. Um eine IAM-Rolle bereitzustellen, muss der IAM-Benutzer oder die IAM-Rolle, die die Replikationskonfiguration erstellt hat, über entsprechende Berechtigungen verfügen. `iam:PassRole` Weitere Informationen finden Sie im Benutzerhandbuch unter [Gewähren von Benutzerberechtigungen zur Übergabe einer Rolle an einen AWS Dienst](#).AWS Identity and Access Management

- Weitere Informationen zu finden Sie im Beispiel unter [Verwendung von Service-gebundenen Rollen für Amazon EFS](#). `iam:CreateServiceLinkedRole`
- Weitere Informationen zu einer benutzerdefinierten IAM-Rolle finden Sie unter [Erstellen Sie eine IAM-Rolle mit einer benutzerdefinierten Vertrauensrichtlinie](#).

Note

Wenn Sie eine kontenübergreifende Replikation durchführen, müssen Sie bei der Erstellung der Replikationskonfiguration eine IAM-Rolle angeben. Die Verwendung der serviceverknüpften Rolle ist nicht zulässig. Weitere Informationen finden Sie unter [Kontenübergreifende AWS Replikation von EFS-Dateisystemen](#).

Die dienstgebundene Rolle oder IAM-Rolle, die Sie bei der Erstellung der Replikationskonfiguration angeben, muss über die folgenden Berechtigungen für die Replikation verfügen.

- `elasticfilesystem:DescribeFileSystems`
- `elasticfilesystem:CreateFileSystem`
- `elasticfilesystem:CreateReplicationConfiguration`
- `elasticfilesystem>DeleteReplicationConfiguration`
- `elasticfilesystem:DescribeReplicationConfigurations`

Sie können die `AmazonElasticFileSystemFullAccess` verwaltete Richtlinie verwenden, um automatisch alle erforderlichen EFS-Berechtigungen zu erhalten. Weitere Informationen finden Sie unter [AWS verwaltete Richtlinie: AmazonElasticFileSystemFullAccess](#).

Konfiguration der Replikation auf ein neues EFS-Dateisystem

Amazon EFS erstellt automatisch ein neues Dateisystem und kopiert die Daten und Metadaten auf dem Quelldateisystem in ein neues schreibgeschütztes Zieldateisystem in dem von AWS-Region Ihnen ausgewählt. Wenn Sie in ein neues Dateisystem replizieren, wählen Sie den Dateisystemtyp und den Schlüssel AWS Key Management Service (AWS KMS), der für die Verschlüsselung verwendet werden soll. Darüber hinaus erstellt Amazon EFS bei der Erstellung des Zieldateisystems keine Mount-Ziele. Nachdem Sie die Replikationskonfiguration erstellt haben, müssen Sie [ein oder mehrere Mount-Ziele erstellen](#), um [ein Zieldateisystem zu mounten](#).

 Note

Ein Dateisystem kann nur Teil einer Replikationskonfiguration sein. Das Quelldateisystem kann in einer anderen Replikationskonfiguration kein Zieldateisystem sein.

- **Dateisystemtyp** – Der Dateisystemtyp bestimmt die Verfügbarkeit und Haltbarkeit, mit der ein Amazon-EFS-Dateisystem Daten in einer AWS-Region speichert.
 - Wählen Sie **Regional** aus, um ein Dateisystem zu erstellen, das Daten und Metadaten redundant in allen Availability Zones innerhalb der AWS-Region speichert.
 - Wählen Sie **One Zone** aus, um ein Dateisystem zu erstellen, das Daten und Metadaten redundant innerhalb einer Availability Zone speichert.

Weitere Informationen über Dateisystemtypen finden Sie unter [EFS-Dateisystemtypen](#).

- **Verschlüsselung** – Alle Zieldateisysteme werden mit aktivierter Verschlüsselung im Ruhezustand erstellt. Sie können den AWS KMS Schlüssel angeben, der zum Verschlüsseln des Zieldateisystems verwendet wird. Wenn Sie keinen KMS-Schlüssel angeben, wird der vom Service verwaltete KMS-Schlüssel für Amazon EFS verwendet.

 Important

Der KMS-Schlüssel kann nicht geändert werden, nachdem das Zieldateisystem erstellt wurde.

Das Zieldateisystem wird mit Standardeinstellungen erstellt, die auf Ihrem Quelldateisystem basieren. Zusätzliche Einstellungen können nach der Erstellung geändert werden.

- **Automatische Sicherungen** – Für Zieldateisysteme, die One-Zone-Speicher verwenden, sind automatische Sicherungen standardmäßig aktiviert. Die Einstellung für automatische Sicherungen kann nicht geändert werden, nachdem das Dateisystem erstellt wurde. Weitere Informationen finden Sie unter [Verwaltung automatischer Backups von EFS-Dateisystemen](#).
- **Leistungsmodus** — Der Leistungsmodus des Zieldateisystems entspricht dem des Quelldateisystems, es sei denn, das Zieldateisystem verwendet One Zone Storage. In diesem Fall wird der Allzweckmodus verwendet. Der Leistungsmodus kann nicht geändert werden.
- **Durchsatzmodus** — Der Durchsatzmodus des Zieldateisystems entspricht dem des Quelldateisystems. Nachdem das Dateisystem erstellt wurde, können Sie den Modus ändern.

Wenn der Durchsatzmodus des Quelldateisystems bereitgestellt ist, entspricht der bereitgestellte Durchsatzbetrag des Zieldateisystems dem des Quelldateisystems, es sei denn, die bereitgestellte Menge der Quelldatei überschreitet den Grenzwert für die Region des Zieldateisystems. Wenn die vom Quelldateisystem bereitgestellte Menge das Limit der Region für das Zieldateisystem überschreitet, entspricht die bereitgestellte Durchsatzmenge des Zieldateisystems dem Limit der Region. Weitere Informationen finden Sie unter [Amazon-EFS-Kontingente, die Sie erhöhen können](#).

- **Lebenszyklusmanagement** – Das Lebenszyklusmanagement ist auf dem Zieldateisystem nicht aktiviert. Nachdem das Zieldateisystem erstellt wurde, können Sie es aktivieren. Weitere Informationen finden Sie unter [Verwaltung des Speicherlebenszyklus](#).

Schritt 1: Erstellen Sie die Replikationskonfiguration

Der erste Schritt bei der Replikation auf ein neues Dateisystem besteht darin, die Replikationskonfiguration zu erstellen. Auf Daten, die in das Zieldateisystem repliziert wurden, kann erst nach Abschluss der ersten Synchronisierung zugegriffen werden. Die Dauer der anfänglichen Synchronisierung hängt von Faktoren wie der Größe des Quelldateisystems und der Anzahl der darin enthaltenen Dateien ab. Weitere Hinweise zur Replikationsleistung finden Sie unter [Replikationsleistung](#).

Verwenden der Konsole

1. Melden Sie sich bei der an AWS-Managementkonsole und öffnen Sie die Amazon EFS-Konsole unter <https://console.aws.amazon.com/efs/>.
2. Öffnen Sie das Dateisystem, das Sie replizieren möchten:
 - a. Wählen Sie im linken Navigationsbereich die Option Dateisysteme aus.
 - b. Wählen Sie in der Liste Dateisysteme das Dateisystem aus, das Sie replizieren möchten. Das von Ihnen gewählte Dateisystem kann in einer vorhandenen Replikationskonfiguration kein Quell- oder Zieldateisystem sein.
3. Wählen Sie die Registerkarte Replikation.
4. Wählen Sie im Abschnitt Replikation die Option Replikation erstellen aus.
5. Definieren Sie im Abschnitt Replikationseinstellungen die Replikationseinstellungen:

- a. Wählen Sie für die Replikationskonfiguration aus, ob auf ein neues oder ein vorhandenes Dateisystem repliziert werden soll.
 - b. Wählen Sie unter Ziel das aus AWS-Region, AWS-Region in das das Dateisystem repliziert werden soll.
6. Definieren Sie im Abschnitt Einstellungen für das Zieldateisystem die Einstellungen für das Zieldateisystem.
- a. Wählen Sie unter Dateisystemtyp eine Speicheroption für das Dateisystem aus:
 - Um ein Dateisystem zu erstellen, das Daten redundant in mehreren geografisch getrennten Availability Zones innerhalb einer speichert AWS-Region, wählen Sie Regional.
 - Um ein Dateisystem zu erstellen, das Daten redundant innerhalb einer einzigen Availability Zone in einer speichert AWS-Region, wählen Sie One Zone und dann Availability Zone aus.

Weitere Informationen finden Sie unter [EFS-Dateisystemtypen](#).

 Note

One-Zone-Dateisysteme sind nicht in allen Availability Zones in der AWS-Regionen verfügbar, in denen Amazon EFS verfügbar ist.

- b. Bei der Verschlüsselung wird die Verschlüsselung von Daten im Ruhezustand automatisch auf dem Zieldateisystem aktiviert. Standardmäßig verwendet Amazon EFS Ihren AWS Key Management Service (AWS KMS) Serviceschlüssel (aws/elasticfilesystem). Um einen anderen KMS-Schlüssel zu verwenden, wählen Sie den KMS-Schlüssel oder geben Sie den Amazon-Ressourcennamen (ARN) des Schlüssels ein.

 Important

Der KMS-Schlüssel kann nicht geändert werden, nachdem das Dateisystem erstellt wurde.

Erstellen Sie die Replikationskonfiguration (AWS CLI)

Dieser Abschnitt enthält Beispiele für die Erstellung einer Replikationskonfiguration AWS CLI mithilfe des `create-replication-configuration` Befehls. Der äquivalente API-Befehl lautet [CreateReplicationConfiguration](#).

Example: Erstellen Sie eine Replikationskonfiguration für ein regionales Zielsystem

Im folgenden Beispiel wird eine Replikationskonfiguration für das Dateisystem `fs-0123456789abcdef1` erstellt. Das Beispiel verwendet den `Region` Parameter, um ein Zielsystem in der zu erstellen `eu-west-2` AWS-Region. Der `KmsKeyId` Parameter gibt die KMS-Schlüssel-ID an, die beim Verschlüsseln des Zielsystems verwendet werden soll:

```
aws efs create-replication-configuration \
--source-file-system-id fs-0123456789abcdef1 \
--destinations "[{\"Region\":\"eu-west-2\", \"KmsKeyId\":\"arn:aws:kms:us-east-2:111122223333:key/abcd1234-ef56-ab78-cd90-1111abcd2222\"}]"
```

Der AWS CLI reagiert wie folgt:

```
{
  "SourceFileSystemArn": "arn:aws:elasticfilesystem:us-east-1:111122223333:file-system/fs-0123456789abcdef1",
  "SourceFileSystemRegion": "us-east-1",
  "Destinations": [
    {
      "Status": "ENABLING",
      "FileSystemId": "fs-0123456789abcde22",
      "Region": "eu-west-2"
    }
  ],
  "SourceFileSystemId": "fs-0123456789abcdef1",
  "CreationTime": 1641491892.0,
  "OriginalSourceFileSystemArn": "arn:aws:elasticfilesystem:us-east-1:111122223333:file-system/fs-0123456789abcdef1"
}
```

Example: Erstellen Sie eine Replikationskonfiguration für ein One-Zone-Zieldateisystem

Im folgenden Beispiel wird eine Replikationskonfiguration für das Dateisystem

fs-0123456789abcdef1 erstellt. In dem Beispiel wird der `AvailabilityZoneName` Parameter verwendet, um ein One Zone-Zieldateisystem in der *us-west-2a* Availability Zone zu erstellen. Da kein KMS-Schlüssel angegeben ist, wird das Zieldateisystem mit dem AWS KMS Standard-Serviceschlüssel (`aws/elasticfilesystem`) des Kontos verschlüsselt.

```
aws efs create-replication-configuration \  
--source-file-system-id fs-0123456789abcdef1 \  
--destinations AvailabilityZoneName=us-west-2a
```

Schritt 2: Mounten Sie das Zieldateisystem

Amazon EFS erstellt keine Mount-Ziele, wenn es das Zieldateisystem erstellt. Um das Zieldateisystem zu mounten, müssen Sie ein oder mehrere Mount-Ziele erstellen. Weitere Informationen finden Sie unter [Mounting von EFS-Dateisystemen](#).

Konfiguration der Replikation auf ein vorhandenes EFS-Dateisystem

Amazon EFS repliziert die Daten und Metadaten des Quelldateisystems in das von Ihnen gewählte AWS-Region Zieldateisystem. Während der Replikation identifiziert Amazon EFS Datenunterschiede zwischen den Dateisystemen und wendet die Unterschiede auf das Zieldateisystem an.

Um auf ein vorhandenes Dateisystem zu replizieren, führen Sie die folgenden Schritte aus.

Topics

- [Schritt 1: Deaktivieren Sie den Replikationsschutz des Dateisystems vor Überschreibung](#)
- [Schritt 2: Erstellen Sie die Replikationskonfiguration](#)

Note

Ein Dateisystem kann nur Teil einer Replikationskonfiguration sein. Das Quelldateisystem kann in einer anderen Replikationskonfiguration kein Zieldateisystem sein.

Schritt 1: Deaktivieren Sie den Replikationsschutz des Dateisystems vor Überschreibung

Wenn Sie ein Amazon EFS-Dateisystem erstellen, ist dessen Replikationsüberschreibschutz standardmäßig aktiviert. Der Replikationsüberschreibschutz verhindert, dass das Dateisystem als Ziel in einer Replikationskonfiguration verwendet wird. Bevor Sie das Dateisystem als Ziel in einer Replikationskonfiguration verwenden können, müssen Sie den Schutz deaktivieren. Wenn die Replikationskonfiguration gelöscht wird, wird der Replikationsüberschreibschutz des Dateisystems wieder aktiviert und das Dateisystem wird beschreibbar.

Der Status des Replikationsüberschreibschutzes für ein Amazon-EFS-Dateisystem kann einen der in der folgenden Tabelle beschriebenen Statuswerte haben.

Status des Dateisystems	Description
ENABLED (AKTIVIERT)	Das Dateisystem kann nicht als Zieldateisystem in einer Replikationskonfiguration verwendet werden. Das Dateisystem ist beschreibbar. Der Überschreibschutz für die Replikation ist standardmäßig ENABLED.
DISABLED (DEAKTIVIERT)	Das Dateisystem kann als Zieldateisystem in einer Replikationskonfiguration verwendet werden.
REPLICATING	Das Dateisystem wird als Zieldateisystem in einer Replikationskonfiguration verwendet. Das Dateisystem ist schreibgeschützt und wird nur durch die Amazon-EFS-Replikation geändert.

Erforderliche Berechtigung

Für die Deaktivierung des Replikationsüberschreibschutzes sind Berechtigungen für die Aktion `elasticfilesystem:UpdateFileSystemProtection` erforderlich. Weitere Informationen finden Sie unter [AWS verwaltete Richtlinie: AmazonElasticFileSystemFullAccess](#).

Verwenden der Konsole

1. Melden Sie sich bei der an AWS-Managementkonsole und öffnen Sie die Amazon EFS-Konsole unter <https://console.aws.amazon.com/efs/>.
2. Wählen Sie im linken Navigationsbereich die Option Dateisysteme aus.

3. Wählen Sie in der Liste Dateisysteme das Amazon-EFS-Dateisystem aus, das Sie als Zieldateisystem in einer Replikationskonfiguration verwenden möchten.
4. Deaktivieren Sie im Abschnitt Dateisystemschutz die Option Überschreibschutz bei der Replikation.

Um den Replikationsschutz vor Überschreibung zu deaktivieren ()AWS CLI

Im folgenden Beispiel deaktiviert der `update-file-system-protection` CLI-Befehl den Replikationsüberschreibschutz für das angegebene Dateisystem. Der entsprechende API-Befehl lautet. [UpdateFileSystemProtection](#)

```
aws efs update-file-system-protection
--file-system-id fs-0a8b2be428114d97c
--replication-overwrite-protection DISABLED
```

Der AWS CLI reagiert wie folgt.

```
{
  "ReplicationOverwriteProtection": "DISABLED"
}
```

Schritt 2: Erstellen Sie die Replikationskonfiguration

Nachdem Sie den Replikationsüberschreibschutz auf dem Zieldateisystem deaktiviert haben, können Sie die Replikationskonfiguration erstellen. Bei der Replikation in ein vorhandenes Dateisystem kann sich das Zieldateisystem im selben Konto oder in einem anderen Konto als das Quelldateisystem befinden.

Bevor Sie eine Replikationskonfiguration für Amazon EFS erstellen, sollten Sie die folgenden wichtigen Anforderungen und Überlegungen überprüfen:

- Wenn das Quelldateisystem verschlüsselt ist, muss auch das Zieldateisystem verschlüsselt werden. Wenn die Quelldatei unverschlüsselt und das Zieldateisystem verschlüsselt ist, können Sie außerdem nach dem Failover kein Failback zum Quellziel durchführen. Weitere Informationen zur Verschlüsselung finden Sie unter [Datenverschlüsselung in Amazon EFS](#).
- Wenn Sie die Replikation zu einem vorhandenen Dateisystem anfänglich konfigurieren, schreibt Amazon EFS Daten in das Zieldateisystem oder entfernt bestehende Daten aus dem Zieldateisystem, sodass sie den Daten im Quelldateisystem entsprechen. Wenn Sie Daten

im Zieldateisystem nicht ändern möchten, sollten Sie stattdessen in ein neues Dateisystem replizieren. Weitere Informationen finden Sie unter [Konfiguration der Replikation auf ein neues EFS-Dateisystem](#).

- Auf Daten, die in das Zieldateisystem repliziert wurden, kann erst nach Abschluss der ersten Synchronisierung zugegriffen werden. Die Dauer der Synchronisierung hängt von Faktoren wie der Größe des Quelldateisystems und der Anzahl der darin enthaltenen Dateien ab. Weitere Hinweise zur Replikationsleistung finden Sie unter [Replikationsleistung](#).

Voraussetzungen

Halten Sie eine Kopie der Zieldateisystem-ID (für Replikation mit demselben Konto) oder des Zieldateisystem-ARN (für kontenübergreifende Replikation) bereit, die Sie verwenden möchten.

Wenn sich das Zieldateisystem in einem anderen AWS-Konto als dem Quelldateisystem befindet, erstellen Sie eine IAM-Rolle, die es Amazon EFS ermöglicht, die Replikation durchzuführen und den Dateisystemen Ressourcenrichtlinien zuzuweisen. Weitere Informationen finden Sie unter [Kontenübergreifende AWS Replikation von EFS-Dateisystemen](#).

Verwenden der Konsole

1. Melden Sie sich bei der an AWS-Managementkonsole und öffnen Sie die Amazon EFS-Konsole unter <https://console.aws.amazon.com/efs/>.
2. Öffnen Sie das Dateisystem, das Sie replizieren möchten:
 - a. Wählen Sie im linken Navigationsbereich die Option Dateisysteme aus.
 - b. Wählen Sie das Amazon-EFS-Dateisystem, das Sie replizieren möchten, aus der Liste der Dateisysteme aus. Das von Ihnen gewählte Dateisystem kann in einer vorhandenen Replikationskonfiguration kein Quell- oder Zieldateisystem sein.
3. Wählen Sie die Registerkarte Replikation.
4. Wählen Sie im Abschnitt Replikation die Option Replikation erstellen aus.
5. Wählen Sie für die Replikationskonfiguration das vorhandene Dateisystem aus.
6. Wählen Sie das Zieldateisystem aus.
 - Um in ein Dateisystem zu replizieren, das sich im selben Dateisystem AWS-Konto wie das Quelldateisystem befindet:
 1. Wählen Sie In diesem Konto ein Dateisystem auswählen aus und wählen Sie unter Ziel das Dateisystem aus AWS-Region, in das AWS-Region das Dateisystem repliziert werden soll.

2. Wählen Sie „EFS durchsuchen“ und wählen Sie dann das Dateisystem aus. Der Pfad zu Ihrem Zieldateisystem wird im Feld Ziel angezeigt.
- Um in ein Dateisystem zu replizieren, das sich in einem anderen Dateisystem AWS-Konto als dem Quelldateisystem befindet:
 1. Wählen Sie „Dateisystem in einem anderen Konto angeben“.
 2. Geben Sie für Zieldateisystem ARN den Amazon-Ressourcennamen (ARN) des Zieldateisystems ein.

 Note

Wenn der Schutz vor dem Überschreiben der Replikation auf dem Dateisystem aktiviert ist, wird eine Warnung angezeigt. Wählen Sie Schutz deaktivieren, um das Dateisystem auf einer neuen Registerkarte zu öffnen und den Replikationsüberschreibschutz zu deaktivieren. Kehren Sie nach dem Deaktivieren des Schutzes zur Registerkarte Replikation erstellen zurück und klicken Sie auf die Schaltfläche Aktualisieren, um die Meldung zu löschen.

7. Geben Sie für die IAM-Rolle den ARN der IAM-Rolle ein, mit der Amazon EFS in das Zieldateisystem replizieren kann. Dies ist für die Replikation mit demselben Konto optional, für die kontenübergreifende Replikation jedoch erforderlich. Weitere Informationen finden Sie unter [Kontenübergreifende AWS Replikation von EFS-Dateisystemen](#).
8. Wählen Sie Replikation erstellen aus, geben Sie Confirm in das Eingabefeld für die Bestätigungsnachricht ein, und wählen Sie dann Replikation erstellen aus. Im Abschnitt Replikation werden die Replikationsdetails angezeigt.

Um die Replikationskonfiguration zu erstellen (AWS CLI)

Dieser Abschnitt enthält Beispiele für die Erstellung einer Replikationskonfiguration AWS CLI mithilfe des `create-replication-configuration` Befehls. Der äquivalente API-Befehl lautet [CreateReplicationConfiguration](#).

Example: Erstellen Sie eine Replikationskonfiguration für ein vorhandenes Zieldateisystem in einer anderen Region

Im folgenden Beispiel wird eine Replikationskonfiguration erstellt, bei der die Dateisystem-ID auf die Dateisystem-ID *fs-0a8b2be428114d97c* in der *eu-west-2* AWS-Region repliziert *fs-0123456789abcdef1* wird.

```
aws efs create-replication-configuration \
--source-file-system-id fs-0123456789abcdef1 \
--destinations "[{\"Region\":\"eu-west-2\",\"FileSystemId\":\"fs-0a8b2be428114d97c\"}]"
```

Der AWS CLI reagiert wie folgt:

```
{
  "SourceFileSystemId": "fs-0123456789abcdef1",
  "SourceFileSystemRegion": "us-east-1",
  "SourceFileSystemArn": "arn:aws:elasticfilesystem:us-east-1:111122223333:file-system/fs-0123456789abcdef1",
  "OriginalSourceFileSystemArn": "arn:aws:elasticfilesystem:us-east-1:111122223333:file-system/fs-0123456789abcdef1",
  "CreationTime": "2024-10-20T20:40:13+00:00",
  "Destinations": [
    {
      "Status": "ENABLING",
      "FileSystemId": "fs-0a8b2be428114d97c",
      "Region": "eu-west-2",
      "OwnerId": "123456789012",
    }
  ],
  "SourceFileSystemOwnerId": "123456789012"
}
```

Example: Erstellen Sie eine kontenübergreifende Replikationskonfiguration

Im folgenden Beispiel wird eine Replikationskonfiguration erstellt, bei der sich das Quell- und das Zieldateisystem unterscheiden AWS-Konten. Die Quelldateisystem-ID *fs-0123456789abcdef1* im Konto *555666777888* wird auf die Dateisystem-ID *fs-0a8b2be428114d97c* im Konto *123456789012* repliziert. Das Beispiel gibt den Amazon-Ressourcennamen (ARN) des Zieldateisystems und den ARN der IAM-Rolle im Quellkonto an, sodass Amazon EFS die Replikation in seinem Namen durchführen kann. Da kein KMS-Schlüssel angegeben ist, wird das Zieldateisystem

mit dem AWS KMS Standard-Serviceschlüssel (aws/elasticfilesystem) des Kontos verschlüsselt.

```
aws efs
--region $REGION
--endpoint $ENDPOINT create-replication-configuration
--source-file-system-id fs-0123456789abcdef1
--destinations Region=eu-west-2,FileSystemId=arn:aws:elasticfilesystem:eu-
west-2:123456789012:file-system/
fs-0a8b2be428114d97c,RoleArn=arn:aws:iam::555666777888:role/cross-account-replication
```

Der AWS CLI reagiert wie folgt:

```
{
  "SourceFileSystemId": "fs-0123456789abcdef1",
  "SourceFileSystemRegion": "us-east-1",
  "SourceFileSystemArn": "arn:aws:elasticfilesystem:us-east-1:555666777888:file-
system/fs-0123456789abcdef1",
  "OriginalSourceFileSystemArn": "arn:aws:elasticfilesystem:us-
east-1:555666777888:file-system/fs-0123456789abcdef1",
  "CreationTime": "2024-10-20T20:40:13+00:00",
  "Destinations": [
    {
      "Status": "ENABLING",
      "FileSystemId": "fs-0a8b2be428114d97c",
      "Region": "eu-west-2",
      "OwnerId": "123456789012",
      "RoleArn": "arn:aws:iam::555666777888:role/cross-account-replication"
    }
  ],
  "SourceFileSystemOwnerId": "555666777888"
}
```

Kontenübergreifende AWS Replikation von EFS-Dateisystemen

Sie können EFS-Dateisysteme überall AWS-Konten replizieren. Die kontenübergreifende Replikation verbessert die allgemeine Widerstandsfähigkeit und Zuverlässigkeit Ihrer Disaster-Recovery-Strategien (DR) und kann Ihnen helfen, die Compliance-Anforderungen Ihres Unternehmens zu erfüllen.

Beispielsweise könnten Sie aufgrund von Compliance-Richtlinien verpflichtet sein, unterschiedliche Konten für unterschiedliche Umgebungen (z. B. Produktion, Staging und Disaster Recovery

(DR)) zu verwenden. Oder Sie stellen möglicherweise fest, dass die Replikation zwischen verschiedenen Systemen AWS-Konten eine stärkere Isolierung, eine detailliertere Kontrolle über Berechtigungen und Zugriffsrichtlinien und eine einfachere Prüfung von Ressourcen ermöglicht. Wenn das Produktionskonto kompromittiert ist (z. B. durch Sicherheitslücken, Fehlkonfigurationen oder Bedrohungen von innen), kann ein separater Zugriff auf die DR-Server verhindern, den Explosionsradius von Sicherheitsvorfällen verringern und das Risiko unbefugter Änderungen minimieren.

Für die Replikation zwischen diesen Systemen AWS-Konten sind zusätzliche Sicherheits- und Richtlinieneinstellungen erforderlich. Anstatt serviceverknüpfte Rollen für die kontenübergreifende Replikation zu verwenden, müssen Sie eine IAM-Rolle erstellen, die Amazon EFS die Berechtigung erteilt, die Replikation im Zielkonto durchzuführen. Sie müssen auch Richtlinien für die Dateisysteme erstellen, die Sie für mehrere Konten gemeinsam nutzen möchten. Nachdem die IAM-Rollen- und Dateisystemrichtlinien erstellt wurden, erstellen Sie die Replikationskonfiguration.

Topics

- [Erstellen Sie eine IAM-Rolle mit einer benutzerdefinierten Vertrauensrichtlinie](#)
- [Erstellen Sie Richtlinien für die Quell- und Zieldateisysteme](#)
- [Erstellen Sie die Replikationskonfiguration](#)

Erstellen Sie eine IAM-Rolle mit einer benutzerdefinierten Vertrauensrichtlinie

Damit Amazon EFS die kontoübergreifende Replikation im Namen des Quellkontos durchführen kann, muss eine IAM-Rolle für das Quellkonto erstellt werden. Die Rolle muss über die `elasticfilesystem.amazonaws.com` Vertrauensrichtlinie verfügen, damit Amazon EFS die Rolle übernehmen und als Service Principal agieren kann. Die Rolle muss alle IAM-Berechtigungen enthalten, die für die Durchführung der Replikation erforderlich sind (siehe [Erforderliche IAM-Berechtigungen](#)) und die ausdrückliche Berechtigung zur Replikation in das Dateisystem im Zielkonto gewähren.

Voraussetzungen

Sie müssen sowohl das Quelldateisystem als auch das Zieldateisystem in der Replikationskonfiguration erstellen, bevor Sie die IAM-Rolle für das Quellkonto erstellen können. Amazon EFS kann das Zieldateisystem während der Replikation nicht für Sie erstellen. Darüber hinaus müssen Sie den Amazon-Ressourcennamen (ARN) für jedes Dateisystem kennen und angeben.

Um die IAM-Rolle für die kontenübergreifende Replikation zu erstellen

Im Folgenden finden Sie die allgemeinen Schritte zum Erstellen einer IAM-Rolle mit benutzerdefinierten Vertrauensrichtlinien für die kontenübergreifende Replikation mit Amazon EFS. step-by-stepAnweisungen zum Erstellen einer IAM-Rolle finden Sie unter [Erstellen einer Rolle mithilfe benutzerdefinierter Vertrauensrichtlinien](#) im AWS Identity and Access Management Benutzerhandbuch.

1. Erstellen Sie in der AWS Identity and Access Management Konsole für das Quellkonto eine IAM-Rolle, die die folgende Vertrauensrichtlinie verwendet. Anweisungen finden Sie unter [Erstellen einer Rolle mithilfe benutzerdefinierter Vertrauensrichtlinien](#) im AWS Identity and Access Management-Benutzerhandbuch.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "elasticfilesystem.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

2. Nachdem Sie die Rolle erstellt haben, weisen Sie ihr die folgenden Berechtigungen zu. `arn:aws:elasticfilesystem:us-east-1:111122223333:file-system/fs-0123456789abcdef1` Ersetzen Sie durch den ARN des Zieldateisystems und `arn:aws:elasticfilesystem:us-east-1:444455556666:file-system/fs-5678910112hijkqr1` ersetzen Sie ihn durch den ARN des Quelldateisystems. Anweisungen zum Zuweisen von Berechtigungen zur Rolle finden Sie unter [Richtlinien mit dem JSON-Editor erstellen](#).

JSON

```
{
```

```

    "Version": "2012-10-17",
    "Statement": [
      {
        "Effect": "Allow",
        "Action": [
          "elasticfilesystem:DescribeFileSystems",
          "elasticfilesystem:CreateReplicationConfiguration",
          "elasticfilesystem:DescribeReplicationConfigurations",
          "elasticfilesystem>DeleteReplicationConfiguration",
          "elasticfilesystem:ReplicationWrite"
        ],
        "Resource": "arn:aws:elasticfilesystem:us-  
east-1:111122223333:file-system/fs-0123456789abcdef1"
      },
      {
        "Effect": "Allow",
        "Action": [
          "elasticfilesystem:ReplicationRead",
          "elasticfilesystem:DescribeFileSystems"
        ],
        "Resource": "arn:aws:elasticfilesystem:us-  
east-1:444455556666:file-system/fs-5678910112hijkqr1"
      }
    ]
  }
}

```

3. Kopieren Sie den ARN für die IAM-Rolle oder notieren Sie ihn. Sie müssen den ARN angeben, wenn Sie die Replikationskonfiguration erstellen.

Erstellen Sie Richtlinien für die Quell- und Zieldateisysteme

Um Dateisysteme kontenübergreifend in Amazon EFS gemeinsam zu nutzen, müssen Sie Richtlinien sowohl dem Ziel- als auch dem Quelldateisystem zuweisen. Die Richtlinien gewähren oder beschränken kontenübergreifend den Zugriff auf das Dateisystem, auf das sie angewendet werden. Nur Kontoinhaber mit der Berechtigung, Dateisysteme zu bearbeiten, können dem Dateisystem in ihrem Konto Richtlinien zuweisen.

Die Richtlinien müssen nicht nur den Zugriff auf Konten gewähren oder einschränken, sondern auch andere Berechtigungen gewähren, die Clients für die Arbeit mit den Dateisystemen benötigen, wie `elasticfilesystem:ClientMount` z. Andernfalls können Clients möglicherweise nicht auf das Dateisystem zugreifen.

⚠ Important

Sie können den Zugriff auf Ressourcen über eine TLS-Verbindung nicht einschränken. Wenn Sie die "aws:SecureTransport": "false" Bedingung in Ihre Anweisung aufnehmen, schlägt die NFS-Client-Verbindung fehl.

Richtlinie für das Zielsystem

Damit das Quellkonto berechtigt ist, in das Zielsystem zu replizieren und die Replikationskonfiguration aus dem Zielkonto zu löschen, muss die folgende Richtlinie auf dem Zielsystem erstellt werden. `arn:aws:iam::444455556666:root` Ersetzen Sie es durch die ID des Kontos, dem das Quellsystem gehört. Durch `arn:aws:elasticfilesystem:us-east-1:111122223333:file-system/fs-0123456789abcdef1` den ARN des Zielsystems ersetzen

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowSourceAccountReplicationActions",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::444455556666:root"
      },
      "Action": [
        "elasticfilesystem:DescribeFileSystems",
        "elasticfilesystem:CreateReplicationConfiguration",
        "elasticfilesystem:DescribeReplicationConfigurations",
        "elasticfilesystem>DeleteReplicationConfiguration",
        "elasticfilesystem:ReplicationWrite"
      ],
      "Resource": "arn:aws:elasticfilesystem:us-east-1:111122223333:file-system/fs-0123456789abcdef1"
    },
    {
      "Sid": "AllowReadOnlyClientAccess",
      "Effect": "Allow",
      "Principal": {
```



```

        "AWS": "arn:aws:iam::111122223333:role/EfsReadOnly"
    },
    "Action": [
        "elasticfilesystem:ClientMount"
    ],
    "Resource": "arn:aws:elasticfilesystem:us-east-1:111122223333:file-
system/fs-0123456789abcdef1"
    }
]
}

```

Richtlinie für das Quelldateisystem

Damit das Zielkonto berechtigt ist, die Replikationskonfiguration aus dem Quellkonto zu löschen, müssen Sie dem Quelldateisystem die folgende Richtlinie zuweisen.

arn:aws:iam::111122223333:root Ersetzen Sie es durch die ID des Kontos, dem das Zieldateisystem gehört. arn:aws:elasticfilesystem:us-east-1:444455556666:file-system/fs-5678910112hijkqr1 Ersetzen Sie durch den ARN des Quelldateisystems.

JSON

```

{
    "Version": "2012-10-17",
    "Id": "efs-policy-wizard-15ad9567-2546-4bbb-8168-5541b6fc0e55",
    "Statement": [
        {
            "Sid": "AllowDestinationAccountToDeleteReplication",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:root"
            },
            "Action": "elasticfilesystem:DeleteReplicationConfiguration",
            "Resource": "arn:aws:elasticfilesystem:us-east-1:444455556666:file-
system/fs-5678910112hijkqr1"
        },
        {
            "Sid": "AllowClientAccess",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:role/EfsReadOnly"
            },
        },
    ]
}

```

```

    "Action": [
        "elasticfilesystem:ClientMount",
        "elasticfilesystem:ClientWrite",
        "elasticfilesystem:ClientRootAccess"
    ],
    "Resource": "arn:aws:elasticfilesystem:us-east-1:444455556666:file-
system/fs-5678910112hijkqr1",
    "Condition": {
        "Bool": {
            "elasticfilesystem:AccessedViaMountTarget": "true"
        }
    }
}

```

Um die Dateisystemrichtlinie zu erstellen

Führen Sie die folgenden Schritte sowohl für das Ziel- als auch für das Quelldateisystem aus und verwenden Sie dabei die Richtlinien aus dem vorherigen Abschnitt.

1. Melden Sie sich AWS-Managementkonsole mit dem Konto an, dem das Dateisystem gehört, und öffnen Sie dann die Amazon EFS-Konsole unter [Amazon EFS Console](#).
2. Öffnen Sie das Dateisystem:
 - a. Wählen Sie im linken Navigationsbereich die Option Dateisysteme aus.
 - b. Wählen Sie in der Liste Dateisysteme das Dateisystem aus.
3. Wählen Sie auf der Registerkarte Dateisystemrichtlinie die Option Bearbeiten aus.
4. Fügen Sie die Richtlinie in den Richtlinieneditor {Json} ein und wählen Sie dann Speichern.

Erstellen Sie die Replikationskonfiguration

Nachdem Sie die IAM-Rolle erstellt und die Dateisystemrichtlinien zu den Quell- und Zieldateisystemen hinzugefügt haben, folgen Sie den Anweisungen unter [Konfiguration der Replikation auf ein vorhandenes EFS-Dateisystem](#). So erstellen Sie die Replikationskonfiguration.

Replikationsdetails anzeigen

In einer Replikationskonfiguration können Sie den Zeitpunkt überwachen, zu dem die letzte erfolgreiche Synchronisierung abgeschlossen wurde. Alle Änderungen an Daten im Quelldateisystem, die vor diesem Zeitpunkt vorgenommen wurden, wurden erfolgreich in das Zieldateisystem repliziert. Alle Änderungen, die nach diesem Zeitpunkt vorgenommen wurden, werden möglicherweise nicht vollständig repliziert. Um zu überwachen, wann die letzte Replizierung erfolgreich abgeschlossen wurde, können Sie die Amazon EFS-Konsole AWS CLI, API oder Amazon verwenden CloudWatch.

- In der EFS-Konsole — Die Eigenschaft Zuletzt synchronisiert im Abschnitt Dateisystemdetails > Replikation zeigt den Zeitpunkt an, zu dem die letzte erfolgreiche Synchronisierung zwischen Quelle und Ziel abgeschlossen wurde.
- In der API AWS CLI oder — Die LastReplicatedTimestamp Eigenschaft im Destination Objekt zeigt den Zeitpunkt an, zu dem die letzte erfolgreiche Synchronisierung abgeschlossen wurde. Verwenden Sie den describe-replication-configurations-CLI-Befehl, um auf diese Eigenschaft zuzugreifen. [DescribeReplicationConfigurations](#) ist die entsprechende API-Operation.
- In CloudWatch — Die TimeSinceLastSync CloudWatch Metrik für Amazon EFS zeigt die Zeit, die seit Abschluss der letzten erfolgreichen Synchronisierung vergangen ist. Weitere Informationen finden Sie unter [CloudWatch Metriken für Amazon EFS](#).

Eine Replikationskonfiguration kann einen der in der folgenden Tabelle beschriebenen Statuswerte haben.

Replikationsstatus	Description
ENABLED	Die Replikationskonfiguration befindet sich in einem fehlerfreien Zustand und kann verwendet werden.
ENABLING	Amazon EFS ist dabei, die Replikationskonfiguration zu erstellen.
DELETING	Amazon EFS löscht die Replikationskonfiguration als Antwort auf eine vom Benutzer initiierte Löschanforderung.
PAUSING	Amazon EFS ist dabei, die Replikation anzuhalten.

Replikationsstatus	Description
PAUSED	Die Replikation wurde aufgrund eines Problems mit der Konfiguration angehalten. Zusätzliche Informationen zu dem Problem werden bereitgestellt. Zu den Problemen, die dazu führen, dass die Replikation unterbrochen wird, gehören: • Autorisierungsfehler. Berechtigungsprobleme verhindern, dass Amazon EFS eine Replikation durchführt. Stellen Sie sicher, dass die IAM-Rolle, die zur Erstellung der Replikationskonfiguration verwendet wurde, berechtigt ist, die Replikation durchzuführen. Stellen Sie außerdem sicher, dass die Dateisystemrichtlinien korrekt sind. • AWS Konto ist nicht verfügbar. Vergewissern Sie sich, dass die Quell- und Zielkonten für die entsprechenden Konten AWS-Regionen aktiviert und nicht gesperrt sind. Weitere Informationen finden Sie im AWS Allgemeinen Referenzhandbuch unter „Geben AWS-Regionen Sie an, welches Konto verwendet werden kann“. • Auf den KMS-Schlüssel für das Quell- oder Zieldateisystem kann nicht zugegriffen werden. Stellen Sie sicher, dass auf den jedem Dateisystem zugewiesenen KMS-Schlüssel zugegriffen werden kann. Weitere Informationen finden Sie unter AWS KMS Schlüssel für Amazon EFS verwenden.
ERROR	Die Replikationskonfiguration befindet sich in einem fehlerhaften Zustand und kann nicht wiederhergestellt werden. Sie müssen die Replikationskonfiguration löschen und eine neue erstellen. Zusätzliche Informationen zu dem Problem werden bereitgestellt. Bei der konto- oder regionsübergreifenden Replikation kann der Fehler dadurch verursacht werden, dass die Replikationskonfiguration aus dem anderen AWS-Konto oder gelöscht wurde. AWS-Region

Verwenden der Konsole

1. Öffnen Sie die Amazon Elastic File System-Konsole unter <https://console.aws.amazon.com/efs/>.
2. Wählen Sie im linken Navigationsbereich die Option Dateisysteme aus.

3. Wählen Sie ein Dateisystem aus der Liste aus.
4. Wählen Sie die Registerkarte Replikation aus, um den Abschnitt Replikation anzuzeigen.

Im Abschnitt Replikation finden Sie die folgenden Informationen zur Replikationskonfiguration:

- Der Replikationsstatus kann Aktivieren, Aktiviert, Löschen, Wird angehalten, Angehalten oder Fehler lauten. Amazon EFS zeigt Details zur Ursache für den Status „Angehalten“ oder „Fehler“ an.
- Die Replikationsrichtung gibt die Richtung an, in die Daten repliziert werden. Das erste aufgelistete Dateisystem ist die Quelle, und die Daten werden auf das zweite aufgelistete Dateisystem repliziert, das das Ziel ist.
- Zuletzt synchronisiert zeigt an, wann die letzte erfolgreiche Synchronisierung im Zieldateisystem stattgefunden hat. Alle Änderungen an Daten im Quelldateisystem, die vor diesem Zeitpunkt vorgenommen wurden, wurden erfolgreich in das Zieldateisystem repliziert. Alle Änderungen, die nach diesem Zeitpunkt vorgenommen wurden, werden möglicherweise nicht vollständig repliziert.
- Replikationsdateisysteme listet jedes Dateisystem in der Replikationskonfiguration nach seiner Dateisystem-ID, der Rolle, die es in der Replikationskonfiguration spielt (entweder Quelle oder Ziel), dem Ort, AWS-Region in dem es sich befindet, und seiner Berechtigung auf. Ein Quelldateisystem hat die Berechtigung Schreibbar und ein Zieldateisystem hat die Berechtigung Schreibgeschützt.

Unter Verwendung der AWS CLI

Verwenden Sie den `describe-replication-configurations` Befehl, um eine Replikationskonfiguration anzuzeigen. Sie können die Replikationskonfiguration entweder für ein bestimmtes Dateisystem oder alle Replikationskonfigurationen für ein bestimmtes Dateisystem AWS-Konto in einer bestimmten AWS-Region. Der äquivalente API-Befehl lautet [DescribeReplicationConfigurations](#).

Wenn der Status der Replikationskonfiguration `PAUSED` oder `ERROR` lautet, werden im `StatusMessage` Parameter Informationen zur Ursache des Problems und zur Behebung des Problems zurückgegeben.

Example: Zeigt die Replikationskonfiguration für ein bestimmtes Dateisystem an

Das folgende Beispiel beschreibt die Replikationskonfiguration für das Dateisystem `fs-0123456789abcdef1`.

```
aws efs describe-replication-configurations --file-system-id fs-0123456789abcdef1
```

The AWS CLI reagiert wie folgt.

```
{
  "Replications": [
    {
      "SourceFileSystemArn": "arn:aws:elasticfilesystem:eu-
west-1:111122223333:file-system/fs-abcdef0123456789a",
      "CreationTime": 1641491892.0,
      "SourceFileSystemRegion": "eu-west-1",
      "OriginalSourceFileSystemArn": "arn:aws:elasticfilesystem:eu-
west-1:111122223333:file-system/fs-abcdef0123456789a",
      "SourceFileSystemId": "fs-abcdef0123456789a",
      "Destinations": [
        {
          "Status": "ENABLED",
          "FileSystemId": "fs-0123456789abcdef1",
          "Region": "us-east-1"
        }
      ]
    }
  ]
}
```

Example: Die Replikationskonfiguration für alle Replikationskonfigurationen in einem Konto anzeigen

Das folgende Beispiel beschreibt die Replikationskonfiguration für alle Replikationskonfigurationen für ein Konto AWS-Region im Dateisystem.

```
aws efs describe-replication-configurations
```

Der AWS CLI reagiert wie folgt.

```
{
  "Replications": [
    {
```

```

    "SourceFileSystemArn": "arn:aws:elasticfilesystem:eu-
west-1:555555555555:file-system/fs-0123456789abcdef1",
    "CreationTime": 1641491892.0,
    "SourceFileSystemRegion": "eu-west-1",
    "OriginalSourceFileSystemArn": "arn:aws:elasticfilesystem:eu-
west-1:555555555555:file-system/fs-0123456789abcdef1",
    "SourceFileSystemId": "fs-0123456789abcdef1",
    "Destinations": [
      {
        "Status": "ENABLED",
        "FileSystemId": "fs-abcdef0123456789a",
        "Region": "us-east-1",
        "LastReplicatedTimestamp": 1641491802.375
      }
    ]
  },
  {
    "SourceFileSystemArn": "arn:aws:elasticfilesystem:eu-
west-1:555555555555:file-system/fs-021345abcdef6789a",
    "CreationTime": 1641491822.0,
    "SourceFileSystemRegion": "eu-west-1",
    "OriginalSourceFileSystemArn": "arn:aws:elasticfilesystem:eu-
west-1:555555555555:file-system/fs-021345abcdef6789a",
    "SourceFileSystemId": "fs-021345abcdef6789a",
    "Destinations": [
      {
        "Status": "ENABLED",
        "FileSystemId": "fs-012abc3456789def1",
        "Region": "us-east-1",
        "LastReplicatedTimestamp": 1641491823.575
      }
    ]
  }
]
}

```

Löschen von Replikationskonfigurationen

Wenn Sie ein Failover auf das Zielsystem durchführen müssen, löschen Sie die Replikationskonfiguration, zu der es gehört. Nachdem Sie eine Replikationskonfiguration gelöscht haben, ist das Zielsystem schreibbar und der Replikationsüberschreibschutz ist wieder aktiviert. Weitere Informationen finden Sie unter [Das Replikat wird verwendet](#).

Das Löschen einer Replikationskonfiguration und das Ändern des Zielfilesystems, sodass es schreibbar ist, kann mehrere Minuten dauern. Nachdem die Konfiguration gelöscht wurde, schreibt Amazon EFS möglicherweise einige Daten in ein `lost+found`-Verzeichnis im Stammverzeichnis des Zielfilesystems und verwendet dabei die folgende Namenskonvention:

```
efs-replication-lost+found-source-file-system-id-TIMESTAMP
```

 Note

Dateisysteme, die Teil einer Replikationskonfiguration sind, können nicht gelöscht werden. Sie müssen die Replikationskonfiguration löschen, bevor Sie das Dateisystem löschen.

Sie können eine bestehende Replikationskonfiguration entweder aus dem Quell- oder dem Zielfilesystem löschen, indem Sie die Amazon EFS-Konsole, AWS CLI, die oder die API verwenden.

Für konto- oder regionsübergreifende Replizierungen löscht Amazon EFS die Replikationskonfiguration sowohl aus den Quell- als auch aus den Zielkonten oder Regionen. Wenn es ein Konfigurations- oder Berechtigungsproblem gibt, das Amazon EFS daran hindert, die Replikationskonfiguration von beiden Seiten zu löschen, können Sie die Konfiguration nur von der lokalen Seite löschen (dem Konto oder der Region, von der aus das Löschen durchgeführt wird). Wenn Sie die lokale Konfiguration löschen, kann die Konfiguration im anderen Konto oder in der anderen Region nicht wiederhergestellt werden.

Verwenden der Konsole

1. Öffnen Sie die Amazon Elastic File System-Konsole unter <https://console.aws.amazon.com/efs/>.
2. Wählen Sie im linken Navigationsbereich die Option Dateisysteme aus.
3. Wählen Sie entweder das Quell- oder das Zielfilesystem aus, das sich in der Replikationskonfiguration befindet, die Sie löschen möchten.
4. Wählen Sie die Registerkarte Replikation aus, um den Abschnitt Replikation anzuzeigen.
5. Wählen Sie Replikation löschen aus, um die Replikationskonfiguration zu löschen. Wenn Sie dazu aufgefordert werden, bestätigen Sie Ihre Auswahl.

Wenn Sie eine Konfiguration für die kontenübergreifende Replikation löschen und ein Problem auftritt, das Sie daran hindert, die Konfiguration sowohl von der Quell- als auch von der Zielseite zu löschen, können Sie die Option wählen, nur die Konfiguration dieses Dateisystems zu löschen.

 Note

Löschen Sie die Konfiguration des Dateisystems nur, wenn Amazon EFS nicht in der Lage ist, die Replikationskonfiguration sowohl im Quell- als auch im Zielkonto oder in der Region zu löschen. Wenn Sie die lokale Konfiguration löschen, kann die Konfiguration im anderen Konto oder in der anderen Region nicht wiederhergestellt werden.

Mit dem AWS CLI

Verwenden Sie die `delete-replication-configuration` CLI, um eine Replikationskonfiguration zu löschen. Der äquivalente API-Befehl lautet [DeleteReplicationConfiguration](#).

Im folgenden Beispiel wird die Replikationskonfiguration für das Quelldateisystem `fs-0123456789abcdef1` gelöscht.

```
aws efs --region us-west-2 delete-replication-configuration \  
--source-file-system-id fs-0123456789abcdef1
```

Wenn Amazon EFS aufgrund eines Konfigurations- oder Berechtigungsproblems die Replikationskonfiguration nicht von beiden Seiten löscht, können Sie die Konfiguration nur von der lokalen Seite löschen (dem Konto oder der Region, von der aus das Löschen durchgeführt wird). Wenn Sie die lokale Konfiguration löschen, kann die Konfiguration im anderen Konto oder in der anderen Region nicht wiederhergestellt werden. Der entsprechende API-Parameter ist `DeletionMode` und der Wert ist `LOCAL_CONFIGURATION_ONLY`.

Im folgenden Beispiel wird die Replikationskonfiguration für das Quelldateisystem nur `fs-0123456789abcdef1` von der lokalen Seite gelöscht.

```
aws efs --region us-west-2 delete-replication-configuration \  
--source-file-system-id fs-0123456789abcdef1 \  
--deletion-mode LOCAL_CONFIGURATION_ONLY
```

Das Replikat wird verwendet

Im Notfall oder bei der Durchführung von Übungen am Spieltag können Sie ein Failover auf Ihr Replikat-Dateisystem durchführen, indem Sie dessen Replikationskonfiguration löschen. Nachdem

die Replikationskonfiguration gelöscht wurde, ist das Replikat schreibbar und Sie können es in Ihrem Anwendungsworkflow verwenden. Wenn der Notfall abgemildert ist oder die Übung am Spieltag vorbei ist, können Sie das Replikat weiterhin als primäres Dateisystem verwenden oder Sie können ein Failback durchführen, um den Betrieb auf Ihrem ursprünglichen primären Dateisystem wieder aufzunehmen.

Während des Failback-Vorgangs können Sie auswählen, ob Sie die an Ihrem Replikat-Dateisystem vorgenommenen Änderungen verwerfen oder sie beibehalten möchten, indem Sie sie zurück auf Ihr primäres Dateisystem kopieren.

- Um die während des Failovers an Ihrem Replikat vorgenommenen Änderungen zu verwerfen, erstellen Sie die ursprüngliche Replikationskonfiguration auf Ihrem primären Dateisystem neu, wobei das Replikatdateisystem das Replikationsziel ist. Während der Replikation synchronisiert Amazon EFS die Dateisysteme, indem es die Daten Ihres Replikatdateisystems aktualisiert, sodass sie mit denen Ihres primären Dateisystems übereinstimmen.
- Um die während des Failovers an Ihrem Replikat vorgenommenen Änderungen zu replizieren, erstellen Sie eine Replikationskonfiguration auf Ihrem primären Replikat-Dateisystem neu, wobei das primäre Dateisystem das Replikationsziel ist. Während der Replikation identifiziert Amazon EFS die Unterschiede von Ihrem Replikat-Dateisystem und überträgt sie zurück in das primäre Dateisystem. Sobald die Replikation abgeschlossen ist, können Sie mit der Replizierung des primären Dateisystems fortfahren, indem Sie die ursprüngliche Replikationskonfiguration erneut erstellen oder eine neue Konfiguration erstellen.

Die Zeit, die Amazon EFS benötigt, um den Replikationsprozess abzuschließen, ist unterschiedlich und hängt von Faktoren wie der Größe des Dateisystems und der Anzahl der darin enthaltenen Dateien ab. Weitere Informationen finden Sie unter [Replikationsleistung](#).

Sicherung Ihrer Daten in Amazon EFS

Cloud-Sicherheit AWS hat höchste Priorität. Als AWS Kunde profitieren Sie von einer Rechenzentrums- und Netzwerkarchitektur, die darauf ausgelegt sind, die Anforderungen der sicherheitssensibelsten Unternehmen zu erfüllen.

Sicherheit ist eine gemeinsame Verantwortung von Ihnen AWS und Ihnen. Das [Modell der geteilten Verantwortung](#) beschreibt dies als Sicherheit der Cloud und Sicherheit in der Cloud:

- Sicherheit der Cloud — AWS ist verantwortlich für den Schutz der Infrastruktur, auf der AWS Dienste in der ausgeführt AWS Cloud werden. AWS bietet Ihnen auch Dienste, die Sie sicher nutzen können. Externe Prüfer testen und verifizieren regelmäßig die Wirksamkeit unserer Sicherheitsmaßnahmen im Rahmen der [AWS](#). Weitere Informationen zu den Compliance-Programmen, die für Amazon Elastic File System gelten, finden Sie unter [AWS Services in Scope by Compliance Program AWS](#).
- Sicherheit in der Cloud — Ihre Verantwortung richtet sich nach dem AWS Service, den Sie nutzen. Sie sind auch für andere Faktoren verantwortlich, etwa für die Vertraulichkeit Ihrer Daten, für die Anforderungen Ihres Unternehmens und für die geltenden Gesetze und Vorschriften.

Diese Dokumentation hilft Ihnen zu verstehen, wie Sie das Modell der gemeinsamen Verantwortung bei der Verwendung von Amazon EFS anwenden können. In den folgenden Themen erfahren Sie, wie Sie Amazon EFS konfigurieren, um Ihre Sicherheits- und Compliance-Ziele zu erreichen. Sie lernen auch, wie Sie andere AWS Dienste nutzen können, die Sie bei der Überwachung und Sicherung Ihrer Amazon EFS-Ressourcen unterstützen.

Topics

- [Datenschutz in Amazon EFS](#)
- [Identitäts- und Zugriffsmanagement für Amazon EFS](#)
- [Verwendung von IAM zur Steuerung des Zugriffs auf Dateisysteme](#)
- [Konformitätsprüfung für Amazon EFS](#)
- [Resilienz in Amazon EFS](#)
- [Steuern des Netzwerkzugriffs auf EFS-Dateisysteme für NFS-Clients](#)
- [Benutzer, Gruppen und Berechtigungen auf NFS-Ebene \(Network File System\)](#)
- [Arbeiten mit Zugriffspunkten](#)
- [Sperren des öffentlichen Zugriffs auf EFS-Dateisysteme](#)

- [Netzwerkisolierung für Amazon EFS](#)

Datenschutz in Amazon EFS

Das AWS [Modell](#) der gilt für den Datenschutz in Amazon EFS. Wie in diesem Modell beschrieben, AWS ist es für den Schutz der globalen Infrastruktur verantwortlich, auf der alle Systeme laufen AWS Cloud. Sie sind dafür verantwortlich, die Kontrolle über Ihre in dieser Infrastruktur gehosteten Inhalte zu behalten. Sie sind auch für die Sicherheitskonfiguration und die Verwaltungsaufgaben für die von Ihnen verwendeten AWS-Services verantwortlich. Weitere Informationen zum Datenschutz finden Sie unter [Häufig gestellte Fragen zum Datenschutz](#). Informationen zum Datenschutz in Europa finden Sie im Blog-Beitrag [AWS -Modell der geteilten Verantwortung und in der DSGVO](#) im AWS - Sicherheitsblog.

Aus Datenschutzgründen empfehlen wir, dass Sie AWS-Konto Anmeldeinformationen schützen und einzelne Benutzer mit AWS IAM Identity Center oder AWS Identity and Access Management (IAM) einrichten. So erhält jeder Benutzer nur die Berechtigungen, die zum Durchführen seiner Aufgaben erforderlich sind. Außerdem empfehlen wir, die Daten mit folgenden Methoden schützen:

- Verwenden Sie für jedes Konto die Multi-Faktor-Authentifizierung (MFA).
- Wird verwendet SSL/TLS , um mit AWS Ressourcen zu kommunizieren. Wir benötigen TLS 1.2 und empfehlen TLS 1.3.
- Richten Sie die API und die Protokollierung von Benutzeraktivitäten mit ein AWS CloudTrail. Informationen zur Verwendung von CloudTrail Pfaden zur Erfassung von AWS Aktivitäten finden Sie unter [Arbeiten mit CloudTrail Pfaden](#) im AWS CloudTrail Benutzerhandbuch.
- Verwenden Sie AWS Verschlüsselungslösungen zusammen mit allen darin enthaltenen Standardsicherheitskontrollen AWS-Services.
- Verwenden Sie erweiterte verwaltete Sicherheitsservices wie Amazon Macie, die dabei helfen, in Amazon S3 gespeicherte persönliche Daten zu erkennen und zu schützen.
- Wenn Sie für den Zugriff AWS über eine Befehlszeilenschnittstelle oder eine API FIPS 140-3-validierte kryptografische Module benötigen, verwenden Sie einen FIPS-Endpunkt. Weitere Informationen über verfügbare FIPS-Endpunkte finden Sie unter [Federal Information Processing Standard \(FIPS\) 140-3](#).

Wir empfehlen dringend, in Freitextfeldern, z. B. im Feld Name, keine vertraulichen oder sensiblen Informationen wie die E-Mail-Adressen Ihrer Kunden einzugeben. Dies gilt auch, wenn Sie mit Amazon EFS oder anderen AWS-Services über die Konsole AWS CLI, API oder arbeiten AWS SDKs.

Alle Daten, die Sie in Tags oder Freitextfelder eingeben, die für Namen verwendet werden, können für Abrechnungs- oder Diagnoseprotokolle verwendet werden. Wenn Sie eine URL für einen externen Server bereitstellen, empfehlen wir dringend, keine Anmeldeinformationen zur Validierung Ihrer Anforderung an den betreffenden Server in die URL einzuschließen.

Themen

- [Datenverschlüsselung in Amazon EFS](#)
- [Datenschutz zwischen Netzwerken](#)

Datenverschlüsselung in Amazon EFS

Amazon EFS bietet umfassende Verschlüsselungsfunktionen, um Ihre Daten sowohl im Speicher als auch bei der Übertragung zu schützen.

- Verschlüsselung im Ruhezustand — Verschlüsselt Daten, die in Ihrem Dateisystem gespeichert sind.
- Verschlüsselung bei der Übertragung — Verschlüsselt Daten, während sie zwischen Ihren Clients und dem Dateisystem übertragen werden.

Wenn Ihr Unternehmen Unternehmens- oder behördlichen Richtlinien unterliegt, die die Verschlüsselung von Daten und Metadaten vorschreiben, empfehlen wir, ein Dateisystem zu erstellen, das im Ruhezustand verschlüsselt wird, und Ihr Dateisystem mithilfe der Verschlüsselung von Daten bei der Übertragung zu mounten.

Themen

- [Verschlüsseln von Daten im Ruhezustand](#)
- [Verschlüsseln von Daten während der Übertragung](#)
- [AWS KMS Schlüssel für Amazon EFS verwenden](#)
- [Fehlerbehebung bei der Verschlüsselung](#)

Verschlüsseln von Daten im Ruhezustand

Die Verschlüsselung im Ruhezustand verschlüsselt Daten, die auf Ihrem EFS-Dateisystem gespeichert sind. Auf diese Weise können Sie Compliance-Anforderungen erfüllen und sensible Daten vor unbefugtem Zugriff schützen. Ihr Unternehmen verlangt möglicherweise die

Verschlüsselung aller Daten, die einer bestimmten Klassifizierung entsprechen oder mit einer bestimmten Anwendung, einem Workload oder einer bestimmten Umgebung verknüpft sind.

 Note

Die Infrastruktur AWS für die Schlüsselverwaltung verwendet von den Federal Information Processing Standards (FIPS) 140-3 zugelassene kryptografische Algorithmen. Die Infrastruktur entspricht den Empfehlungen der National Institute of Standards and Technology (NIST) 800-57.

Wenn Sie mit der Amazon EFS-Konsole ein Dateisystem erstellen, ist die Verschlüsselung im Ruhezustand standardmäßig aktiviert. Wenn Sie die AWS CLI API oder SDKs zum Erstellen eines Dateisystems verwenden, müssen Sie die Verschlüsselung ausdrücklich aktivieren.

Nachdem Sie ein EFS-Dateisystem erstellt haben, können Sie dessen Verschlüsselungseinstellung nicht ändern. Das bedeutet, dass Sie ein unverschlüsseltes Dateisystem nicht so ändern können, dass es verschlüsselt wird. [Replizieren Sie stattdessen das Dateisystem](#), um Daten aus dem unverschlüsselten Dateisystem in ein neues verschlüsseltes Dateisystem zu kopieren. Weitere Informationen finden Sie unter [Wie aktiviere ich die Verschlüsselung im Ruhezustand für ein vorhandenes EFS-Dateisystem?](#)

Funktionsweise der Verschlüsselung im Ruhezustand

In einem verschlüsselten Dateisystem werden Daten und Metadaten standardmäßig verschlüsselt, bevor sie in den Speicher geschrieben werden, und beim Lesen automatisch entschlüsselt. Diese Prozesse werden von Amazon EFS transparent abgewickelt, sodass Sie Ihre Anwendungen nicht ändern müssen.

Amazon EFS verwendet AWS KMS für die Schlüsselverwaltung Folgendes:

- Verschlüsselung von Dateidaten — Der Inhalt Ihrer Dateien wird mit dem von Ihnen angegebenen KMS-Schlüssel verschlüsselt. Dies kann entweder sein:
 - Die Standardoption AWS-eigener Schlüssel für Amazon EFS (aws/elasticfilesystem), keine zusätzlichen Gebühren.
 - Ein vom Kunden verwalteter Schlüssel, den Sie erstellen und verwalten — Bietet zusätzliche Kontroll- und Prüffunktionen.
- Metadatenverschlüsselung — Dateinamen, Verzeichnisnamen und Verzeichnisinhalte werden mit einem Schlüssel verschlüsselt, den Amazon EFS intern verwaltet.

Verschlüsselungsprozess

Wenn ein Dateisystem erstellt oder in ein Dateisystem desselben Kontos repliziert wird, verwendet Amazon EFS eine [Forward Access Session \(FAS\)](#), um KMS-Aufrufe mit den Anmeldeinformationen des Anrufers durchzuführen. In den CloudTrail Protokollen scheint der `kms:CreateGrant` Aufruf von derselben Benutzeridentität getätigt worden zu sein, mit der das Dateisystem oder die Replikation erstellt wurde. Sie können eingegangene Amazon EFS-Serviceaufrufe identifizieren, CloudTrail indem Sie nach dem `invokedBy` Feld mit dem Wert `suchenelasticfilesystem.amazonaws.com`. Die Ressourcenrichtlinie für den KMS-Schlüssel muss die `CreateGrant` Aktion zulassen, damit FAS den Aufruf tätigen kann.

Important

Sie haben die Kontrolle über den Zuschuss und können ihn jederzeit widerrufen. Durch den Widerruf der Gewährung kann Amazon EFS nicht mehr auf den KMS-Schlüssel für future Operationen zugreifen. Weitere Informationen finden Sie unter [Zuschüsse zurückziehen und widerrufen im Entwicklerhandbuch](#).AWS Key Management Service .

Bei der Verwendung von kundenverwalteten KMS-Schlüsseln muss die Ressourcenrichtlinie auch den Amazon EFS-Service Principal zulassen und die `kms:ViaService` Bedingung enthalten, den Zugriff auf den jeweiligen Service-Endpunkt einzuschränken. Beispiel:

```
"kms:ViaService":  
  "elasticfilesystem.us-east-2.amazonaws.com"
```

Amazon EFS verwendet den branchenüblichen AES-256-Verschlüsselungsalgorithmus, um Daten und Metadaten im Ruhezustand zu verschlüsseln.

Weitere Informationen zu KMS-Schlüsselrichtlinien für Amazon EFS finden Sie unter [AWS KMS Schlüssel für Amazon EFS verwenden](#).

Erzwingen der Verschlüsselung im Ruhezustand für neue Dateisysteme

Sie können den `elasticfilesystem:Encrypted` IAM-Bedingungsschlüssel in AWS Identity and Access Management (IAM-) identitätsbasierten Richtlinien verwenden, um die Erstellung im Ruhezustand zu erzwingen, wenn Benutzer EFS-Dateisysteme erstellen. Weitere Informationen zum Verwenden des Bedingungsschlüssels finden Sie unter [Beispiel: Erzwingen der Erstellung verschlüsselter Dateisysteme](#).

Sie können auch interne Service-Kontrollrichtlinien (SCPs) definieren AWS Organizations , um die Amazon EFS-Verschlüsselung für alle AWS-Konten in Ihrer Organisation durchzusetzen. Weitere Informationen zu den Richtlinien zur Servicesteuerung finden Sie unter [Servicesteuerungsrichtlinien](#) im AWS Organizations Benutzerhandbuch. AWS Organizations

Verschlüsseln von Daten während der Übertragung

Amazon EFS unterstützt die Verschlüsselung von Daten während der Übertragung mit Transport Layer Security (TLS). Wenn die Verschlüsselung von Daten während der Übertragung als Mount-Option für Ihr EFS-Dateisystem deklariert ist, stellt Amazon EFS beim Mounten Ihres Dateisystems eine sichere TLS-Verbindung mit Ihrem EFS-Dateisystem her. Der gesamte NFS-Verkehr wird über diese verschlüsselte Verbindung geleitet.

Funktionsweise der Verschlüsselung während der Übertragung

Wir empfehlen die Verwendung der EFS-Mountinghilfe zum Mounten Ihres Dateisystems, da sie den Einhängvorgang im Vergleich zum Einhängen mit `NFS mount` vereinfacht. Der EFS-Mount-Helper verwaltet den Prozess, indem er entweder `efs-proxy` (für `efs-utils` Version 2.0.0 und höher) oder `stunnel` (für frühere Versionen von `efs-utils`) verwendet, um eine sichere TLS-Verbindung mit Ihrem EFS-Dateisystem herzustellen.

Sie können die Datenverschlüsselung während der Übertragung aber auch ohne die Mountinghilfe aktivieren. Im Folgenden finden Sie die dafür erforderlichen Schritte.

Um die Verschlüsselung von Daten während der Übertragung zu aktivieren, ohne den Mount Helper zu verwenden

1. Laden Sie `stunnel` herunter und installieren Sie es, und notieren Sie sich den Port, auf dem die Anwendung lauscht. Weitere Informationen finden Sie unter [Upgraden von stunnel](#).
2. Führen Sie `stunnel` den Befehl aus, um mithilfe von TLS eine Verbindung zu Ihrem EFS-Dateisystem auf Port 2049 herzustellen.
3. Mounten Sie mithilfe des NFS-Clients `localhost:port`, wobei `port` der Port ist, den Sie sich im ersten Schritt notiert haben.

Da die Verschlüsselung von Daten während der Übertragung für jede einzelne Verbindung konfiguriert wird, läuft für jede konfigurierte Verbindung ein eigener `stunnel`-Vorgang auf der Instance. Standardmäßig überwacht der vom Mount Helper verwendete Stunnel-Prozess einen lokalen Port zwischen 20049 und 20449 und stellt auf Port 2049 eine Verbindung zu Amazon EFS her.

 Note

Wenn der EFS-Mount-Helper mit TLS verwendet wird, erzwingt er standardmäßig die Verwendung des Online Certificate Status Protocol (OCSP) und die Überprüfung von Zertifikats-Hostnamen. Der EFS-Mount-Helper verwendet das Stunnel-Programm für seine TLS-Funktionalität. In manchen Linux-Versionen ist keine Stunnel-Version enthalten, die diese TLS-Features standardmäßig unterstützt. Wenn Sie eine dieser Linux-Versionen verwenden, schlägt das Mounten eines EFS-Dateisystems mithilfe von TLS fehl.

Wenn Sie das `amazon-efs-utils` Paket installiert haben, erfahren Sie unter [Upgraden von stunnel](#), wie Sie die Version von `stunnel` auf Ihrem System aktualisieren.

Informationen zu Problemen mit der Verschlüsselung finden Sie unter [Fehlerbehebung bei der Verschlüsselung](#).

Wenn Sie Datenverschlüsselung während der Übertragung verwenden, ändert sich die Einrichtung des NFS-Clients. Wenn Sie Ihre aktiv gemounteten Dateisysteme untersuchen, sehen Sie eines, das an `127.0.0.1` oder `localhost` gemountet ist, wie im folgenden Beispiel.

```
$ mount | column -t
127.0.0.1:/ on /home/ec2-user/efs          type nfs4
(rw,relatime,vers=4.1,rsiz=1048576,wsiz=1048576,namlen=255,hard,proto=tcp,port=20127,timeo=600)
```

Beim Mounten mit TLS und dem EFS-Mount-Helper konfigurieren Sie Ihren NFS-Client neu, sodass er an einem lokalen Port gemountet wird. Die EFS-Mountinghilfe startet einen Client-Vorgang `stunnel`, der auf diesem lokalen Port lauscht, und `stunnel` öffnet eine verschlüsselte Verbindung zum EFS-Dateisystem über TLS. Die EFS-Mountinghilfe ist für die Einrichtung und Pflege dieser verschlüsselten Verbindung und der zugehörigen Konfiguration zuständig.

Um festzustellen, welche Amazon-EFS-Dateisystem-ID zu welchem lokalen Mounting-Punkt gehört, können Sie den folgenden Befehl verwenden. Denken Sie daran, es `efs-mount-point` durch den lokalen Pfad zu ersetzen, in dem Sie Ihr Dateisystem bereitgestellt haben.

```
grep -E "Successfully mounted.*efs-mount-point" /var/log/amazon/efs/mount.log | tail -1
```

Wenn Sie den EFS-Mount-Helper für die Verschlüsselung von Daten während der Übertragung verwenden, erstellt er auch einen Prozess namens `amazon-efs-mount-watchdog`. Dieser Prozess stellt sicher, dass der Stunnel-Prozess jedes Mounts ausgeführt wird, und stoppt den Stunnel,

wenn das EFS-Dateisystem unmountet wird. Wenn der Stunnel-Vorgang aus irgendeinem Grund unerwartet beendet wird, wird er vom Watchdog-Vorgang neu gestartet.

AWS KMS Schlüssel für Amazon EFS verwenden

Amazon EFS ist für die Schlüsselverwaltung in AWS Key Management Service (AWS KMS) integriert. Amazon EFS verwendet vom Kunden verwaltete Schlüssel, um Ihr Dateisystem auf folgende Weise zu verschlüsseln:

- Verschlüsselung von Metadaten im Ruhezustand – Amazon EFS verwendet das Von AWS verwaltete Schlüssel für Amazon EFS, `aws/elasticfilesystem`, um Metadaten des Dateisystems (d.h. Dateinamen, Verzeichnisnamen und Verzeichnisinhalte) zu ver- und entschlüsseln.
- Verschlüsselung von Dateidaten im Ruhezustand – Sie wählen den vom Kunden verwalteten Schlüssel (CMK), der zur Ver- und Entschlüsselung von Dateidaten (d. h. dem Inhalt Ihrer Dateien) verwendet wird. Sie können Berechtigungen für diesen vom Kunden verwalteten Schlüssel (CMK) aktivieren, deaktivieren oder widerrufen. Dieser von Kunden gemanagte Schlüssel kann einer der beiden folgenden Typen sein:
 - Von AWS verwalteter Schlüssel für Amazon EFS — Dies ist der vom Kunden verwaltete Standardschlüssel, `aws/elasticfilesystem`. Für die Erstellung und Speicherung eines von Kunden gemanagten Schlüssels fallen keine Gebühren an, aber es fallen Nutzungsgebühren an. Weitere Informationen finden Sie auf der Seite über [AWS Key Management Service – Preise](#).
 - Kundenverwalteter Schlüssel – Dies ist der flexibelste KMS-Schlüssel, da Sie seine Schlüsselrichtlinien und Berechtigungen für mehrere Benutzer oder Dienste konfigurieren können. Weitere Informationen zum Erstellen von kundenverwalteten Schlüsseln finden Sie unter [Creating Keys](#) im AWS Key Management Service Developer Guide.

Wenn Sie einen vom Kunden verwalteten Schlüssel (CMK) für die Ver- und Entschlüsselung von Dateidaten verwenden, können Sie die Schlüsselrotation aktivieren. Wenn Sie die Schlüsselrotation aktivieren, AWS KMS wird Ihr Schlüssel automatisch einmal pro Jahr rotiert. Darüber hinaus können Sie bei einem vom Kunden verwalteten Schlüssel (CMK) jederzeit entscheiden, wann Sie den Zugriff auf Ihren vom Kunden verwalteten Schlüssel deaktivieren, wieder aktivieren, löschen oder widerrufen möchten. Weitere Informationen finden Sie unter [AWS KMS Schlüssel für Amazon EFS verwenden](#).

 Important

Amazon EFS akzeptiert nur symmetrische, vom Kunden verwaltete Schlüssel. Sie können keine asymmetrischen, vom Kunden verwalteten Schlüssel (CMK) mit Amazon EFS verwenden.

Die Datenverschlüsselung und -entschlüsselung im Ruhezustand erfolgt transparent. Amazon IDs EFS-spezifische AWS Konten erscheinen jedoch in Ihren AWS CloudTrail Protokollen, die sich auf AWS KMS Aktionen beziehen. Weitere Informationen finden Sie unter [Amazon EFS-Protokolldateieinträge für encrypted-at-rest Dateisysteme](#).

Die wichtigsten Richtlinien von Amazon EFS für AWS KMS

Schlüsselrichtlinien sind die wichtigste Methode zur Kontrolle des Zugriffs auf vom Kunden verwaltete Schlüssel. Weitere Informationen zu Schlüsselrichtlinien finden Sie unter [Schlüsselrichtlinien in AWS KMS](#) im AWS Key Management Service -Entwicklerhandbuch. Die folgende Liste beschreibt alle AWS KMS-bezogenen Berechtigungen, die von Amazon EFS für verschlüsselte Dateisysteme im Ruhezustand benötigt oder anderweitig unterstützt werden:

- kms:Encrypt – (Optional) Verschlüsselt Klartext in Geheimtext. Diese Berechtigung ist in der Standard-Schlüsselrichtlinie enthalten.
- kms:Decrypt – (Erforderlich) Entschlüsselt Geheimtext. Geheimtext ist Klartext, der zuvor verschlüsselt wurde. Diese Berechtigung ist in der Standard-Schlüsselrichtlinie enthalten.
- kms: ReEncrypt — (Optional) Verschlüsselt Daten auf der Serverseite mit einem neuen, vom Kunden verwalteten Schlüssel, ohne dass der Klartext der Daten auf der Clientseite offengelegt wird. Die Daten werden zuerst entschlüsselt und dann neu verschlüsselt. Diese Berechtigung ist in der Standard-Schlüsselrichtlinie enthalten.
- kms: GenerateDataKeyWithoutPlaintext — (Erforderlich) Gibt einen Datenverschlüsselungsschlüssel zurück, der mit einem vom Kunden verwalteten Schlüssel verschlüsselt wurde. Diese Berechtigung ist in der Standardschlüsselrichtlinie unter kms: GenerateDataKey * enthalten.
- kms: CreateGrant — (Erforderlich) Fügt einem Schlüssel einen Zuschuss hinzu, um anzugeben, wer den Schlüssel verwenden kann und unter welchen Bedingungen. Erteilungen sind eine alternative Berechtigungsmethode zu Schlüsselrichtlinien. Weitere Informationen zu Zuschüssen finden Sie unter [Grants AWS KMS im AWS Key Management Service](#) Developer Guide. Diese Berechtigung ist in der Standard-Schlüsselrichtlinie enthalten.

- kms: DescribeKey — (Erforderlich) Stellt detaillierte Informationen zum angegebenen vom Kunden verwalteten Schlüssel bereit. Diese Berechtigung ist in der Standard-Schlüsselrichtlinie enthalten.
- kms: ListAliases — (Optional) Listet alle Schlüsselalias im Konto auf. Wenn Sie die Konsole verwenden, um ein verschlüsseltes Dateisystem zu erstellen, wird mit dieser Berechtigung die Liste KMS-Schlüssel auswählen ausgefüllt. Wir empfehlen für eine optimale Benutzererfahrung diese Berechtigung. Diese Berechtigung ist in der Standard-Schlüsselrichtlinie enthalten.

Von AWS verwalteter Schlüssel für die Amazon EFS KMS-Richtlinie

Die KMS-Richtlinie JSON Von AWS verwalteter Schlüssel für Amazon EFS `aws/elasticfilesystem` lautet wie folgt:

```
{
  "Version": "2012-10-17",
  "Id": "auto-elasticfilesystem-1",
  "Statement": [
    {
      "Sid": "Allow access to EFS for all principals in the account that are
authorized to use EFS",
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
      "Action": [
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:ReEncrypt*",
        "kms:GenerateDataKey*",
        "kms:CreateGrant",
        "kms:DescribeKey"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "kms:ViaService": "elasticfilesystem.us-east-2.amazonaws.com",
          "kms:CallerAccount": "111122223333"
        }
      }
    },
    {
      "Sid": "Allow direct access to key metadata to the account",
```

```
    "Effect": "Allow",
    "Principal": {
      "AWS": "arn:aws:iam::111122223333:root"
    },
    "Action": [
      "kms:Describe*",
      "kms:Get*",
      "kms:List*",
      "kms:RevokeGrant"
    ],
    "Resource": "*"
  }
]
```

Die wichtigsten Staaten und ihre Auswirkungen

Der Status Ihres KMS-Schlüssels wirkt sich direkt auf den Zugriff auf Ihr verschlüsseltes Dateisystem aus:

Aktiviert

Normaler Betrieb — voller Lese- und Schreibzugriff auf das Dateisystem

Disabled

Auf das Dateisystem kann nach einer kurzen Zeit nicht mehr zugegriffen werden. Kann wieder aktiviert werden.

Löschen ausstehend

Auf das Dateisystem kann nicht mehr zugegriffen werden. Das Löschen kann während der Wartezeit abgebrochen werden.

Gelöscht

Auf das Dateisystem kann dauerhaft nicht zugegriffen werden. Diese Aktion kann nicht rückgängig gemacht werden.

Warning

Wenn Sie den für Ihr Dateisystem verwendeten KMS-Schlüssel deaktivieren oder löschen oder Amazon EFS den Zugriff auf den Schlüssel entziehen, kann nicht mehr auf Ihr Dateisystem zugegriffen werden. Dies kann zu Datenverlust führen, wenn Sie keine Backups

haben. Stellen Sie immer sicher, dass Sie über die richtigen Backup-Verfahren verfügen, bevor Sie Änderungen an den Verschlüsselungsschlüsseln vornehmen.

Fehlerbehebung bei der Verschlüsselung

Im Folgenden finden Sie Informationen zur Fehlerbehebung bei Verschlüsselungsproblemen für Amazon EFS.

- [Mounting mit Verschlüsselung der Daten während der Übertragung schlägt fehl](#)
- [Mounting mit Verschlüsselung der Daten während der Übertragung wird unterbrochen](#)
- [Encrypted-at-rest Das Dateisystem kann nicht erstellt werden](#)
- [Nicht verwendbares verschlüsseltes Dateisystem](#)

Mounting mit Verschlüsselung der Daten während der Übertragung schlägt fehl

Wenn Sie die Amazon-EFS-Mountinghilfe mit Transport Layer Security (TLS) verwenden, erzwingt sie standardmäßig eine Hostnamenprüfung. Einige Systeme unterstützen diese Funktion nicht, beispielsweise, wenn Sie Red Hat Enterprise Linux oder CentOS verwenden. In solchen Fällen schlägt das Mounten eines EFS-Dateisystems mit TLS fehl.

Maßnahme

Wir empfehlen ein Upgrade der Stunnel-Version auf Ihrem Client, um die Überprüfung des Hostnamens zu unterstützen. Weitere Informationen finden Sie unter [Upgraden von stunnel](#).

Mounting mit Verschlüsselung der Daten während der Übertragung wird unterbrochen

Es ist möglich, wenn auch unwahrscheinlich, dass Ihre verschlüsselte Verbindung zu Ihrem Amazon-EFS-Dateisystem durch clientseitige Ereignisse hängen bleibt oder unterbrochen wird.

Maßnahme

Wenn Ihre Verbindung zu Ihrem Amazon-EFS-Dateisystem mit Verschlüsselung der Daten während der Übertragung unterbrochen wird, führen Sie die folgenden Schritte aus:

1. Stellen Sie sicher, dass der Stunnel-Service auf dem Client ausgeführt wird.
2. Vergewissern Sie sich, dass die Watchdog-Anwendung `amazon-efs-mount-watchdog` auf dem Client ausgeführt wird. Sie können mit dem folgenden Befehl herausfinden, ob diese Anwendung ausgeführt wird:

```
ps aux | grep [a]mazon-efs-mount-watchdog
```

3. Überprüfen Sie Ihre Support-Protokolle. Weitere Informationen finden Sie unter [Abrufen von Support-Protokollen](#).
4. Optional können Sie Ihre Stunnel-Protokolle aktivieren und auch dort die Informationen prüfen. Sie können die Konfiguration Ihrer Protokolle unter `/etc/amazon/efs/efs-utils.conf` ändern, um die Stunnel-Protokolle zu aktivieren. Hierfür müssen Sie das Dateisystem jedoch mit der Mountinghilfe ausbinden und erneut mounten, um die Änderungen zu übernehmen.

Important

Die Aktivierung der Stunnel-Protokolle kann erheblichen Speicherplatz auf Ihrem Dateisystem beanspruchen.

Wenn die Unterbrechungen weiterhin bestehen, wenden Sie sich an den AWS Support.

Encrypted-at-rest Das Dateisystem kann nicht erstellt werden

Sie haben versucht, ein neues encrypted-at-rest Dateisystem zu erstellen. Sie erhalten jedoch eine Fehlermeldung, dass das Produkt nicht verfügbar AWS KMS ist.

Maßnahme

Dieser Fehler kann in dem seltenen Fall auftreten, dass AWS KMS er in Ihrem vorübergehend nicht verfügbar ist AWS-Region. Warten Sie in diesem Fall, bis AWS KMS die volle Verfügbarkeit wiederhergestellt ist, und versuchen Sie dann erneut, das Dateisystem zu erstellen.

Nicht verwendbares verschlüsseltes Dateisystem

Ein verschlüsseltes Dateisystem gibt ständig NFS-Serverfehler zurück. Diese Fehler können auftreten, wenn EFS Ihren Masterschlüssel aus AWS KMS einem der folgenden Gründe nicht abrufen kann:

- Der Schlüssel wurde deaktiviert.
- Der Schlüssel wurde gelöscht.
- Die Erlaubnis für Amazon EFS, den Schlüssel zu verwenden, wurde widerrufen.
- AWS KMS ist vorübergehend nicht verfügbar.

Maßnahme

Vergewissern Sie sich zunächst, dass der AWS KMS Schlüssel aktiviert ist. Zeigen Sie sich dazu die Schlüssel in der Konsole an. Weitere Informationen finden Sie unter [Schlüssel anzeigen](#) im AWS Key Management Service -Entwicklerhandbuch.

Wenn der Schlüssel nicht aktiviert ist, aktivieren Sie ihn. Weitere Informationen finden Sie unter [Aktivieren und Deaktivieren von Schlüsseln](#) im AWS Key Management Service -Entwicklerhandbuch.

Wenn der Schlüssel zur Löschung ansteht, wird er durch diesen Status deaktiviert. Sie können die Löschung eines Schlüssels abbrechen und den Schlüssel erneut aktivieren. Weitere Informationen finden Sie unter [Planen und Abbrechen des Löschens von Schlüsseln](#) im AWS Key Management Service -Entwicklerhandbuch.

Wenn der Schlüssel aktiviert ist und Sie immer noch ein Problem haben oder wenn bei der erneuten Aktivierung Ihres Schlüssels ein Problem auftritt, wenden Sie sich an den AWS Support.

Datenschutz zwischen Netzwerken

In diesem Thema wird beschrieben, wie Amazon EFS Verbindungen vom Service zu anderen Standorten sichert.

Datenverkehr zwischen Service und On-Premises-Clients und -Anwendungen

Sie haben zwei Verbindungsoptionen zwischen Ihrem privaten Netzwerk und AWS:

- Eine AWS Site-to-Site VPN Verbindung. Weitere Informationen finden Sie unter [Was ist AWS Site-to-Site VPN?](#)
- Eine Direct Connect Verbindung. Weitere Informationen finden Sie unter [Was ist Direct Connect?](#)

Der Zugriff auf Amazon EFS über das Netzwerk erfolgt über AWS Published APIs. Kunden müssen Transport Layer 1.2 oder höher unterstützen. Wir empfehlen TLS 1.3 oder höher. Clients müssen außerdem Cipher Suites mit PFS (Perfect Forward Secrecy) wie DHE (Ephemeral Diffie-Hellman) oder ECDHE (Elliptic Curve Ephemeral Diffie-Hellman) unterstützen. Die meisten modernen Systeme wie Java 7 und höher unterstützen diese Modi. Außerdem müssen Sie die Anforderungen mit einer Zugriffsschlüssel-ID und einem geheimen Zugriffsschlüssel signieren, die einem IAM-Prinzipal zugeordnet sind. Sie können auch [AWS -Security-Token-Service \(AWS STS\)](#) verwenden um temporäre Sicherheitsanmeldeinformationen zu generieren.

Verkehr zwischen VPC und Amazon EFS API

Um eine private Verbindung zwischen Ihrer Virtual Private Cloud (VPC) und der Amazon-EFS-API herzustellen, können Sie einen Schnittstellen-VPC-Endpunkt erstellen. Sie können diese Verbindung verwenden, um die Amazon EFS-API von Ihrer VPC aus aufzurufen, ohne Datenverkehr über das Internet zu senden. Der Endpunkt bietet eine sichere Verbindung zur Amazon-EFS-API, ohne dass ein Internet-Gateway, eine NAT-Instance oder ein Virtual Private Network (VPN) erforderlich ist. Weitere Informationen finden Sie unter [Arbeiten mit VPC-Endpunkten mit Schnittstellen in Amazon EFS](#).

Verkehr zwischen AWS Ressourcen in derselben Region

Ein Amazon Virtual Private Cloud (Amazon VPC) -Endpunkt für Amazon EFS ist eine logische Einheit innerhalb einer VPC, die nur Konnektivität zu Amazon EFS ermöglicht. Die Amazon VPC leitet Anfragen an Amazon EFS weiter und leitet Antworten zurück an die VPC. Weitere Informationen finden Sie unter [VPC-Endpunkte](#) im Amazon VPC-Benutzerhandbuch.

Identitäts- und Zugriffsmanagement für Amazon EFS

AWS Identity and Access Management (IAM) hilft einem Administrator AWS-Service, den Zugriff auf AWS Ressourcen sicher zu kontrollieren. IAM-Administratoren steuern, wer authentifiziert (angemeldet) und autorisiert (im Besitz von Berechtigungen) ist, Amazon-EFS-Ressourcen zu nutzen. IAM ist ein Programm AWS-Service, das Sie ohne zusätzliche Kosten nutzen können.

Themen

- [Zielgruppe](#)
- [Authentifizierung mit Identitäten](#)
- [Verwalten des Zugriffs mit Richtlinien](#)
- [Funktionsweise von Amazon Elastic File System mit IAM](#)
- [Beispiele für identitätsbasierte Richtlinien für Amazon Elastic File System](#)
- [Beispiele für ressourcenbasierte Richtlinien für Amazon EFS](#)
- [AWS verwaltete Richtlinien für Amazon EFS](#)
- [Verwendung von Tags mit Amazon EFS](#)
- [Verwendung von Service-gebundenen Rollen für Amazon EFS](#)

- [Fehlerbehebung für Amazon-Elastic-File-System-Identität und -Zugriff](#)

Zielgruppe

Wie Sie AWS Identity and Access Management (IAM) verwenden, hängt von Ihrer Rolle ab:

- Servicebenutzer – Fordern Sie von Ihrem Administrator Berechtigungen an, wenn Sie nicht auf Features zugreifen können (siehe [Fehlerbehebung für Amazon-Elastic-File-System-Identität und -Zugriff](#)).
- Serviceadministrator – Bestimmen Sie den Benutzerzugriff und stellen Sie Berechtigungsanfragen (siehe [Funktionsweise von Amazon Elastic File System mit IAM](#)).
- IAM-Administrator – Schreiben Sie Richtlinien zur Zugriffsverwaltung (siehe [Beispiele für identitätsbasierte Richtlinien für Amazon Elastic File System](#)).

Authentifizierung mit Identitäten

Authentifizierung ist die Art und Weise, wie Sie sich AWS mit Ihren Identitätsdaten anmelden. Sie müssen sich als IAM-Benutzer authentifizieren oder eine IAM-Rolle annehmen. Root-Benutzer des AWS-Kontos

Sie können sich als föderierte Identität anmelden, indem Sie Anmeldeinformationen aus einer Identitätsquelle wie AWS IAM Identity Center (IAM Identity Center), Single Sign-On-Authentifizierung oder Anmeldeinformationen verwenden. Google/Facebook Weitere Informationen zum Anmelden finden Sie unter [So melden Sie sich bei Ihrem AWS-Konto an](#) im Benutzerhandbuch für AWS-Anmeldung .

AWS Bietet für den programmatischen Zugriff ein SDK und eine CLI zum kryptografischen Signieren von Anfragen. Weitere Informationen finden Sie unter [AWS Signature Version 4 for API requests](#) im IAM-Benutzerhandbuch.

AWS-Konto Root-Benutzer

Wenn Sie einen erstellen AWS-Konto, beginnen Sie mit einer Anmeldeidentität, dem sogenannten AWS-Konto Root-Benutzer, der vollständigen Zugriff auf alle AWS-Services Ressourcen hat. Wir raten ausdrücklich davon ab, den Root-Benutzer für Alltagsaufgaben zu verwenden. Eine Liste der Aufgaben, für die Sie sich als Root-Benutzer anmelden müssen, finden Sie unter [Tasks that require root user credentials](#) im IAM-Benutzerhandbuch.

Verbundidentität

Es hat sich bewährt, dass menschliche Benutzer für den Zugriff AWS-Services mithilfe temporärer Anmeldeinformationen einen Verbund mit einem Identitätsanbieter verwenden müssen.

Eine föderierte Identität ist ein Benutzer aus Ihrem Unternehmensverzeichnis, Ihrem Directory Service Web-Identitätsanbieter oder der AWS-Services mithilfe von Anmeldeinformationen aus einer Identitätsquelle zugreift. Verbundene Identitäten übernehmen Rollen, die temporäre Anmeldeinformationen bereitstellen.

Für die zentrale Zugriffsverwaltung empfehlen wir AWS IAM Identity Center. Weitere Informationen finden Sie unter [Was ist IAM Identity Center?](#) im AWS IAM Identity Center -Benutzerhandbuch.

IAM-Benutzer und -Gruppen

Ein [IAM-Benutzer](#) ist eine Identität mit bestimmten Berechtigungen für eine einzelne Person oder Anwendung. Wir empfehlen die Verwendung temporärer Anmeldeinformationen anstelle von IAM-Benutzern mit langfristigen Anmeldeinformationen. Weitere Informationen finden Sie im IAM-Benutzerhandbuch unter [Erfordern, dass menschliche Benutzer den Verbund mit einem Identitätsanbieter verwenden müssen, um AWS mithilfe temporärer Anmeldeinformationen darauf zugreifen zu](#) können.

Eine [IAM-Gruppe](#) spezifiziert eine Sammlung von IAM-Benutzern und erleichtert die Verwaltung von Berechtigungen für große Gruppen von Benutzern. Weitere Informationen finden Sie unter [Anwendungsfälle für IAM-Benutzer](#) im IAM-Benutzerhandbuch.

IAM-Rollen

Eine [IAM-Rolle](#) ist eine Identität mit spezifischen Berechtigungen, die temporäre Anmeldeinformationen bereitstellt. Sie können eine Rolle übernehmen, indem Sie [von einer Benutzer- zu einer IAM-Rolle \(Konsole\) wechseln](#) AWS CLI oder einen AWS API-Vorgang aufrufen. Weitere Informationen finden Sie unter [Methoden, um eine Rolle zu übernehmen](#) im IAM-Benutzerhandbuch.

IAM-Rollen sind nützlich für den Verbundbenutzer-Zugriff, temporäre IAM-Benutzerberechtigungen, kontoübergreifenden Zugriff, serviceübergreifenden Zugriff und Anwendungen, die auf Amazon EC2 laufen. Weitere Informationen finden Sie unter [Kontoübergreifender Ressourcenzugriff in IAM](#) im IAM-Benutzerhandbuch.

Verwalten des Zugriffs mit Richtlinien

Sie kontrollieren den Zugriff, AWS indem Sie Richtlinien erstellen und diese an AWS Identitäten oder Ressourcen anhängen. Eine Richtlinie definiert Berechtigungen, wenn sie mit einer Identität oder Ressource verknüpft sind. AWS bewertet diese Richtlinien, wenn ein Principal eine Anfrage stellt. Die meisten Richtlinien werden AWS als JSON-Dokumente gespeichert. Weitere Informationen zu JSON-Richtliniendokumenten finden Sie unter [Übersicht über JSON-Richtlinien](#) im IAM-Benutzerhandbuch.

Mit Hilfe von Richtlinien legen Administratoren fest, wer Zugriff auf was hat, indem sie definieren, welches Prinzipal welche Aktionen auf welchen Ressourcen und unter welchen Bedingungen durchführen darf.

Standardmäßig haben Benutzer, Gruppen und Rollen keine Berechtigungen. Ein IAM-Administrator erstellt IAM-Richtlinien und fügt sie zu Rollen hinzu, die die Benutzer dann übernehmen können. IAM-Richtlinien definieren Berechtigungen unabhängig von der Methode, die zur Ausführung der Operation verwendet wird.

Identitätsbasierte Richtlinien

Identitätsbasierte Richtlinien sind JSON-Berechtigungsrichtliniendokumente, die Sie einer Identität (Benutzer, Gruppe oder Rolle) anfügen können. Diese Richtlinien steuern, welche Aktionen Identitäten für welche Ressourcen und unter welchen Bedingungen ausführen können. Informationen zum Erstellen identitätsbasierter Richtlinien finden Sie unter [Definieren benutzerdefinierter IAM-Berechtigungen mit vom Kunden verwalteten Richtlinien](#) im IAM-Benutzerhandbuch.

Identitätsbasierte Richtlinien können Inline-Richtlinien (direkt in eine einzelne Identität eingebettet) oder verwaltete Richtlinien (eigenständige Richtlinien, die mit mehreren Identitäten verbunden sind) sein. Informationen dazu, wie Sie zwischen verwalteten und Inline-Richtlinien wählen, finden Sie unter [Choose between managed policies and inline policies](#) im IAM-Benutzerhandbuch.

Ressourcenbasierte Richtlinien

Ressourcenbasierte Richtlinien sind JSON-Richtliniendokumente, die Sie an eine Ressource anfügen. Beispiele hierfür sind Vertrauensrichtlinien für IAM-Rollen und Amazon S3-Bucket-Richtlinien. In Services, die ressourcenbasierte Richtlinien unterstützen, können Service-Administratoren sie verwenden, um den Zugriff auf eine bestimmte Ressource zu steuern. Sie müssen in einer ressourcenbasierten Richtlinie [einen Prinzipal angeben](#).

Ressourcenbasierte Richtlinien sind Richtlinien innerhalb dieses Diensts. Sie können AWS verwaltete Richtlinien von IAM nicht in einer ressourcenbasierten Richtlinie verwenden.

Weitere Richtlinientypen

AWS unterstützt zusätzliche Richtlinientypen, mit denen die maximalen Berechtigungen festgelegt werden können, die durch gängigere Richtlinientypen gewährt werden:

- **Berechtigungsgrenzen** – Eine Berechtigungsgrenze legt die maximalen Berechtigungen fest, die eine identitätsbasierte Richtlinie einer IAM-Entität erteilen kann. Weitere Informationen finden Sie unter [Berechtigungsgrenzen für IAM-Entitäten](#) im IAM-Benutzerhandbuch.
- **Richtlinien zur Dienstkontrolle (SCPs)** — Geben Sie die maximalen Berechtigungen für eine Organisation oder Organisationseinheit in an AWS Organizations. Weitere Informationen finden Sie unter [Service-Kontrollrichtlinien](#) im AWS Organizations -Benutzerhandbuch.
- **Richtlinien zur Ressourcenkontrolle (RCPs)** — Legen Sie die maximal verfügbaren Berechtigungen für Ressourcen in Ihren Konten fest. Weitere Informationen finden Sie im AWS Organizations Benutzerhandbuch unter [Richtlinien zur Ressourcenkontrolle \(RCPs\)](#).
- **Sitzungsrichtlinien** – Sitzungsrichtlinien sind erweiterte Richtlinien, die als Parameter übergeben werden, wenn Sie eine temporäre Sitzung für eine Rolle oder einen Verbundbenutzer erstellen. Weitere Informationen finden Sie unter [Sitzungsrichtlinien](#) im IAM-Benutzerhandbuch.

Mehrere Richtlinientypen

Wenn für eine Anfrage mehrere Arten von Richtlinien gelten, sind die daraus resultierenden Berechtigungen schwieriger zu verstehen. Informationen darüber, wie AWS bestimmt wird, ob eine Anfrage zulässig ist, wenn mehrere Richtlinientypen betroffen sind, finden Sie unter [Bewertungslogik für Richtlinien](#) im IAM-Benutzerhandbuch.

Funktionsweise von Amazon Elastic File System mit IAM

Bevor Sie mit IAM den Zugriff auf Amazon EFS verwalten können, sollten Sie sich darüber informieren, welche IAM-Features Sie mit Amazon EFS verwenden können.

IAM-Features, die Sie mit Amazon Elastic File System verwenden können

IAM-Feature	Amazon-EFS-Support
Identitätsbasierte Richtlinien	Ja

IAM-Feature	Amazon-EFS-Support
Ressourcenbasierte Richtlinien	Ja
Richtlinienaktionen	Ja
Richtlinienressourcen	Ja
Richtlinienbedingungsschlüssel (servicespezifisch)	Ja
ACLs	Nein
ABAC (Tags in Richtlinien)	Teilweise
Temporäre Anmeldeinformationen	Ja
Prinzipalberechtigungen	Ja
Servicerollen	Ja
Service-verknüpfte Rollen	Ja

Einen allgemeinen Überblick darüber, wie Amazon EFS und andere AWS Services mit den meisten IAM-Funktionen funktionieren, finden Sie im [IAM-Benutzerhandbuch unter AWS Services, die mit IAM funktionieren](#).

Identitätsbasierte Richtlinien für Amazon EFS

Unterstützt Richtlinien auf Identitätsbasis: Ja

Identitätsbasierte Richtlinien sind JSON-Berechtigungsrichtliniendokumente, die Sie einer Identität anfügen können, wie z. B. IAM-Benutzern, -Benutzergruppen oder -Rollen. Diese Richtlinien steuern, welche Aktionen die Benutzer und Rollen für welche Ressourcen und unter welchen Bedingungen ausführen können. Informationen zum Erstellen identitätsbasierter Richtlinien finden Sie unter [Definieren benutzerdefinierter IAM-Berechtigungen mit vom Kunden verwalteten Richtlinien](#) im IAM-Benutzerhandbuch.

Mit identitätsbasierten IAM-Richtlinien können Sie angeben, welche Aktionen und Ressourcen zugelassen oder abgelehnt werden. Darüber hinaus können Sie die Bedingungen festlegen, unter

denen Aktionen zugelassen oder abgelehnt werden. Informationen zu sämtlichen Elementen, die Sie in einer JSON-Richtlinie verwenden, finden Sie in der [IAM-Referenz für JSON-Richtlinienelemente](#) im IAM-Benutzerhandbuch.

Beispiele für identitätsbasierte Richtlinien für Amazon EFS

Beispiele für identitätsbasierte Amazon-EFS-Richtlinien finden Sie unter [Beispiele für identitätsbasierte Richtlinien für Amazon Elastic File System](#).

Ressourcenbasierte Richtlinien in Amazon EFS

Unterstützt ressourcenbasierte Richtlinien: Ja

Ressourcenbasierte Richtlinien sind JSON-Richtliniendokumente, die Sie an eine Ressource anfügen. Beispiele für ressourcenbasierte Richtlinien sind IAM-Rollen-Vertrauensrichtlinien und Amazon-S3-Bucket-Richtlinien. In Services, die ressourcenbasierte Richtlinien unterstützen, können Service-Administratoren sie verwenden, um den Zugriff auf eine bestimmte Ressource zu steuern. Für die Ressource, an welche die Richtlinie angehängt ist, legt die Richtlinie fest, welche Aktionen ein bestimmter Prinzipal unter welchen Bedingungen für diese Ressource ausführen kann. Sie müssen in einer ressourcenbasierten Richtlinie [einen Prinzipal angeben](#). Zu den Prinzipalen können Konten, Benutzer, Rollen, Verbundbenutzer oder gehören. AWS-Services

Um kontoübergreifenden Zugriff zu ermöglichen, können Sie ein gesamtes Konto oder IAM-Entitäten in einem anderen Konto als Prinzipal in einer ressourcenbasierten Richtlinie angeben. Weitere Informationen finden Sie unter [Kontoübergreifender Ressourcenzugriff in IAM](#) im IAM-Benutzerhandbuch.

Weitere Informationen zur Verwendung einer Ressourcenrichtlinie zur Steuerung des Datenzugriffs auf Dateisysteme finden Sie unter [Verwendung von IAM zur Steuerung des Zugriffs auf Dateisysteme](#). Weitere Informationen zum Hinzufügen einer ressourcenbasierten Richtlinie zu einem Dateisystem finden Sie unter [Erstellen von Dateisystemrichtlinien](#).

Ressourcenbasierte Richtlinien in Amazon EFS

Beispiele für ressourcenbasierte Amazon-EFS-Richtlinien finden Sie unter [Beispiele für ressourcenbasierte Richtlinien für Amazon EFS](#).

Richtlinienaktionen für Amazon EFS

Unterstützt Richtlinienaktionen: Ja

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer Zugriff auf was hat. Das heißt, welcher Prinzipal Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen kann.

Das Element `Action` einer JSON-Richtlinie beschreibt die Aktionen, mit denen Sie den Zugriff in einer Richtlinie zulassen oder verweigern können. Nehmen Sie Aktionen in eine Richtlinie auf, um Berechtigungen zur Ausführung des zugehörigen Vorgangs zu erteilen.

Eine Liste von Amazon-EFS-Aktionen finden Sie unter [Von Amazon Elastic File System definierte Aktionen](#) in der Service-Autorisierungs-Referenz.

Richtlinienaktionen in Amazon EFS verwenden das folgende Präfix vor der Aktion:

```
elasticfilesystem
```

Um mehrere Aktionen in einer einzigen Anweisung anzugeben, trennen Sie sie mit Kommata:

```
"Action": [  
    "elasticfilesystem:action1",  
    "elasticfilesystem:action2"  
]
```

Beispiele für identitätsbasierte Amazon-EFS-Richtlinien finden Sie unter [Beispiele für identitätsbasierte Richtlinien für Amazon Elastic File System](#).

Richtlinienressourcen für Amazon EFS

Unterstützt Richtlinienressourcen: Ja

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer Zugriff auf was hat. Das heißt, welcher Prinzipal Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen kann.

Das JSON-Richtlinienelement `Resource` gibt die Objekte an, auf welche die Aktion angewendet wird. Als Best Practice geben Sie eine Ressource mit dem zugehörigen [Amazon-Ressourcennamen \(ARN\)](#) an. Verwenden Sie für Aktionen, die keine Berechtigungen auf Ressourcenebene unterstützen, einen Platzhalter (*), um anzugeben, dass die Anweisung für alle Ressourcen gilt.


```
"Resource": "*"
```

Eine Liste der Amazon EFS-Ressourcentypen und ihrer ARNs Eigenschaften finden Sie unter [Von Amazon Elastic File System definierte Ressourcen](#) in der Service Authorization Reference. Sie erfahren, mit welchen Aktionen Sie den ARN einer jeden Ressource angeben können, unter [Von Amazon Elastic File System definierte Aktionen](#).

Beispiele für identitätsbasierte Amazon-EFS-Richtlinien finden Sie unter [Beispiele für identitätsbasierte Richtlinien für Amazon Elastic File System](#).

Richtlinien-Bedingungsschlüssel für Amazon EFS

Unterstützt servicespezifische Richtlinienbedingungsschlüssel: Ja

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer auf was Zugriff hat. Das heißt, welcher Prinzipal Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen kann.

Das Element `Condition` gibt an, wann Anweisungen auf der Grundlage definierter Kriterien ausgeführt werden. Sie können bedingte Ausdrücke erstellen, die [Bedingungsoperatoren](#) verwenden, z. B. ist gleich oder kleiner als, damit die Bedingung in der Richtlinie mit Werten in der Anforderung übereinstimmt. Eine Übersicht aller AWS globalen Bedingungsschlüssel finden Sie unter [Kontextschlüssel für AWS globale Bedingungen](#) im IAM-Benutzerhandbuch.

Eine Liste von Amazon-EFS-Bedingungsschlüsseln finden Sie unter [Bedingungsschlüssel für Amazon Elastic File System](#) in der Service-Autorisierungs-Referenz. Informationen dazu, mit welchen Aktionen und Ressourcen Sie einen Bedingungsschlüssel verwenden können, finden Sie unter [Von Amazon Elastic File System definierte Aktionen](#).

Beispiele für identitätsbasierte Amazon-EFS-Richtlinien finden Sie unter [Beispiele für identitätsbasierte Richtlinien für Amazon Elastic File System](#).

ACLs in Amazon EFS

Unterstützt ACLs: Nein

Zugriffskontrolllisten (ACLs) steuern, welche Principals (Kontomitglieder, Benutzer oder Rollen) über Zugriffsberechtigungen für eine Ressource verfügen. ACLs ähneln ressourcenbasierten Richtlinien, verwenden jedoch nicht das JSON-Richtliniendokumentformat.

ABAC mit Amazon EFS

Unterstützt ABAC (Tags in Richtlinien): Teilweise

Die attributbasierte Zugriffskontrolle (ABAC) ist eine Autorisierungsstrategie, bei der Berechtigungen basierend auf Attributen, auch als Tags bezeichnet, definiert werden. Sie können Tags an IAM-Entitäten und AWS -Ressourcen anhängen und dann ABAC-Richtlinien so entwerfen, dass Operationen möglich sind, wenn das Tag des Prinzipals mit dem Tag auf der Ressource übereinstimmt.

Um den Zugriff auf der Grundlage von Tags zu steuern, geben Sie im Bedingungelement einer [Richtlinie Tag-Informationen](#) an, indem Sie die Schlüssel `aws:ResourceTag/key-name`, `aws:RequestTag/key-name`, oder Bedingung `aws:TagKeys` verwenden.

Wenn ein Service alle drei Bedingungsschlüssel für jeden Ressourcentyp unterstützt, lautet der Wert für den Service Ja. Wenn ein Service alle drei Bedingungsschlüssel für nur einige Ressourcentypen unterstützt, lautet der Wert Teilweise.

Weitere Informationen zu ABAC finden Sie unter [Definieren von Berechtigungen mit ABAC-Autorisierung](#) im IAM-Benutzerhandbuch. Um ein Tutorial mit Schritten zur Einstellung von ABAC anzuzeigen, siehe [Attributbasierte Zugriffskontrolle \(ABAC\)](#) verwenden im IAM-Benutzerhandbuch.

Verwenden temporärer Anmeldeinformationen mit Amazon EFS

Unterstützt temporäre Anmeldeinformationen: Ja

Temporäre Anmeldeinformationen ermöglichen den kurzfristigen Zugriff auf AWS Ressourcen und werden automatisch erstellt, wenn Sie einen Verbund verwenden oder die Rollen wechseln. AWS empfiehlt, temporäre Anmeldeinformationen dynamisch zu generieren, anstatt langfristige Zugriffsschlüssel zu verwenden. Weitere Informationen finden Sie unter [Temporäre Anmeldeinformationen in IAM](#) und [AWS-Services , die mit IAM funktionieren](#) im IAM-Benutzerhandbuch.

Serviceübergreifende Prinzipalberechtigungen für Amazon EFS

Unterstützt Forward Access Sessions (FAS): Ja

Forward Access Sessions (FAS) verwenden die Berechtigungen des Prinzipals, der einen aufruft AWS-Service, in Kombination mit der Anfrage, Anfragen AWS-Service an nachgeschaltete Dienste

zu stellen. Einzelheiten zu den Richtlinien für FAS-Anforderungen finden Sie unter [Zugriffssitzungen weiterleiten](#).

Servicerollen für Amazon EFS

Unterstützt Servicerollen: Ja

Eine Servicerolle ist eine [IAM-Rolle](#), die ein Service annimmt, um Aktionen in Ihrem Namen auszuführen. Ein IAM-Administrator kann eine Servicerolle innerhalb von IAM erstellen, ändern und löschen. Weitere Informationen finden Sie unter [Erstellen einer Rolle zum Delegieren von Berechtigungen an einen AWS-Service](#) im IAM-Benutzerhandbuch.

Warning

Das Ändern der Berechtigungen für eine Servicerolle könnte die Funktionalität von Amazon EFS beeinträchtigen. Bearbeiten Sie Servicerollen nur, wenn Amazon EFS dazu Anleitungen gibt.

Service-verknüpfte Rollen für Amazon EFS

Unterstützt serviceverknüpfte Rollen: Ja

Eine serviceverknüpfte Rolle ist eine Art von Servicerolle, die mit einer verknüpft ist. AWS-Service Der Service kann die Rolle übernehmen, um eine Aktion in Ihrem Namen auszuführen. Dienstbezogene Rollen werden in Ihrem Dienst angezeigt AWS-Konto und gehören dem Dienst. Ein IAM-Administrator kann die Berechtigungen für Service-verknüpfte Rollen anzeigen, aber nicht bearbeiten.

Details zum Erstellen oder Verwalten von serviceverknüpften Amazon-EFS-Rollen finden Sie unter [Verwendung von Service-gebundenen Rollen für Amazon EFS](#).

Beispiele für identitätsbasierte Richtlinien für Amazon Elastic File System

Benutzer und Rollen besitzen standardmäßig keine Berechtigungen zum Erstellen oder Ändern von Amazon-EFS-Ressourcen. Ein IAM-Administrator muss IAM-Richtlinien erstellen, die Benutzern die Berechtigung erteilen, Aktionen für die Ressourcen auszuführen, die sie benötigen.

Informationen dazu, wie Sie unter Verwendung dieser beispielhaften JSON-Richtliniendokumente eine identitätsbasierte IAM-Richtlinie erstellen, finden Sie unter [Erstellen von IAM-Richtlinien \(Konsole\)](#) im IAM-Benutzerhandbuch.

Einzelheiten zu den von Amazon EFS definierten Aktionen und Ressourcentypen, einschließlich des Formats ARNs für die einzelnen Ressourcentypen, finden Sie unter [Aktionen, Ressourcen und Bedingungsschlüssel für Amazon Elastic File System](#) in der Service Authorization Reference.

Themen

- [Best Practices für Richtlinien](#)
- [Verwenden der Amazon-EFS-Konsole](#)
- [Beispiel: Erteilen der Berechtigung zur Anzeige der eigenen Berechtigungen für Benutzer](#)
- [Beispiel: Erzwingen der Erstellung verschlüsselter Dateisysteme](#)
- [Beispiel: Erzwingen der Erstellung unverschlüsselter Dateisysteme](#)

Best Practices für Richtlinien

Identitätsbasierte Richtlinien können festlegen, ob jemand Amazon-EFS-Ressourcen in Ihrem Konto erstellen, darauf zugreifen oder löschen kann. Dies kann zusätzliche Kosten für Ihr verursachen AWS-Konto. Beachten Sie beim Erstellen oder Bearbeiten identitätsbasierter Richtlinien die folgenden Richtlinien und Empfehlungen:

- Erste Schritte mit AWS verwalteten Richtlinien und Umstellung auf Berechtigungen mit den geringsten Rechten — Verwenden Sie die AWS verwalteten Richtlinien, die Berechtigungen für viele gängige Anwendungsfälle gewähren, um damit zu beginnen, Ihren Benutzern und Workloads Berechtigungen zu gewähren. Sie sind in Ihrem verfügbar. AWS-Konto Wir empfehlen Ihnen, die Berechtigungen weiter zu reduzieren, indem Sie vom AWS Kunden verwaltete Richtlinien definieren, die speziell auf Ihre Anwendungsfälle zugeschnitten sind. Weitere Informationen finden Sie unter [Von AWS verwaltete Richtlinien](#) oder [Von AWS verwaltete Richtlinien für Auftragsfunktionen](#) im IAM-Benutzerhandbuch.
- Anwendung von Berechtigungen mit den geringsten Rechten – Wenn Sie mit IAM-Richtlinien Berechtigungen festlegen, gewähren Sie nur die Berechtigungen, die für die Durchführung einer Aufgabe erforderlich sind. Sie tun dies, indem Sie die Aktionen definieren, die für bestimmte Ressourcen unter bestimmten Bedingungen durchgeführt werden können, auch bekannt als die geringsten Berechtigungen. Weitere Informationen zur Verwendung von IAM zum Anwenden von Berechtigungen finden Sie unter [Richtlinien und Berechtigungen in IAM](#) im IAM-Benutzerhandbuch.
- Verwenden von Bedingungen in IAM-Richtlinien zur weiteren Einschränkung des Zugriffs – Sie können Ihren Richtlinien eine Bedingung hinzufügen, um den Zugriff auf Aktionen und Ressourcen zu beschränken. Sie können beispielsweise eine Richtlinienbedingung schreiben,

um festzulegen, dass alle Anforderungen mithilfe von SSL gesendet werden müssen. Sie können auch Bedingungen verwenden, um Zugriff auf Serviceaktionen zu gewähren, wenn diese für einen bestimmten Zweck verwendet werden AWS-Service, z. CloudFormation B. Weitere Informationen finden Sie unter [IAM-JSON-Richtlinienelemente: Bedingung](#) im IAM-Benutzerhandbuch.

- Verwenden von IAM Access Analyzer zur Validierung Ihrer IAM-Richtlinien, um sichere und funktionale Berechtigungen zu gewährleisten – IAM Access Analyzer validiert neue und vorhandene Richtlinien, damit die Richtlinien der IAM-Richtliniensprache (JSON) und den bewährten IAM-Methoden entsprechen. IAM Access Analyzer stellt mehr als 100 Richtlinienprüfungen und umsetzbare Empfehlungen zur Verfügung, damit Sie sichere und funktionale Richtlinien erstellen können. Weitere Informationen finden Sie unter [Richtlinienvvalidierung mit IAM Access Analyzer](#) im IAM-Benutzerhandbuch.
- Multi-Faktor-Authentifizierung (MFA) erforderlich — Wenn Sie ein Szenario haben, das IAM-Benutzer oder einen Root-Benutzer in Ihrem System erfordert AWS-Konto, aktivieren Sie MFA für zusätzliche Sicherheit. Um MFA beim Aufrufen von API-Vorgängen anzufordern, fügen Sie Ihren Richtlinien MFA-Bedingungen hinzu. Weitere Informationen finden Sie unter [Sicherer API-Zugriff mit MFA](#) im IAM-Benutzerhandbuch.

Weitere Informationen zu bewährten Methoden in IAM finden Sie unter [Best Practices für die Sicherheit in IAM](#) im IAM-Benutzerhandbuch.

Verwenden der Amazon-EFS-Konsole

Um auf die Konsole von Amazon Elastic File System zugreifen zu können, müssen Sie über eine Mindestanzahl von Berechtigungen verfügen. Diese Berechtigungen müssen es Ihnen ermöglichen, Details zu den Amazon EFS-Ressourcen in Ihrem aufzulisten und anzuzeigen AWS-Konto. Wenn Sie eine identitätsbasierte Richtlinie erstellen, die strenger ist als die mindestens erforderlichen Berechtigungen, funktioniert die Konsole nicht wie vorgesehen für Entitäten (Benutzer oder Rollen) mit dieser Richtlinie.

Sie müssen Benutzern, die nur die API AWS CLI oder die AWS API aufrufen, keine Mindestberechtigungen für die Konsole gewähren. Stattdessen sollten Sie nur Zugriff auf die Aktionen zulassen, die der API-Operation entsprechen, die die Benutzer ausführen möchten.

Um sicherzustellen, dass Benutzer und Rollen die Amazon EFS-Konsole weiterhin verwenden können, fügen Sie den Entitäten auch die `AmazonElasticFileSystemReadOnlyAccess` AWS verwaltete Amazon EFS-Richtlinie hinzu. Weitere Informationen finden Sie unter [Hinzufügen von Berechtigungen zu einem Benutzer](#) im IAM-Benutzerhandbuch.

Sie finden die `AmazonElasticFileSystemReadOnlyAccess` und andere Amazon EFS Managed Service-Richtlinien unter [AWS verwaltete Richtlinien für Amazon EFS](#).

Beispiel: Erteilen der Berechtigung zur Anzeige der eigenen Berechtigungen für Benutzer

In diesem Beispiel wird gezeigt, wie Sie eine Richtlinie erstellen, die IAM-Benutzern die Berechtigung zum Anzeigen der eingebundenen Richtlinien und verwalteten Richtlinien gewährt, die ihrer Benutzeridentität angefügt sind. Diese Richtlinie umfasst Berechtigungen zum Ausführen dieser Aktion auf der Konsole oder programmgesteuert mithilfe der API AWS CLI oder AWS .

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ViewOwnUserInfo",
      "Effect": "Allow",
      "Action": [
        "iam:GetUserPolicy",
        "iam:ListGroupsForUser",
        "iam:ListAttachedUserPolicies",
        "iam:ListUserPolicies",
        "iam:GetUser"
      ],
      "Resource": ["arn:aws:iam::*:user/${aws:username}"]
    },
    {
      "Sid": "NavigateInConsole",
      "Effect": "Allow",
      "Action": [
        "iam:GetGroupPolicy",
        "iam:GetPolicyVersion",
        "iam:GetPolicy",
        "iam:ListAttachedGroupPolicies",
        "iam:ListGroupPolicies",
        "iam:ListPolicyVersions",
        "iam:ListPolicies",
        "iam:ListUsers"
      ],
      "Resource": "*"
    }
  ]
}
```

```
}
```

Beispiel: Erzwingen der Erstellung verschlüsselter Dateisysteme

Das folgende Beispiel zeigt eine identitätsbasierte Richtlinie, die Prinzipals autorisiert, nur verschlüsselte Dateisysteme zu erstellen.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "elasticfilesystem:CreateFileSystem",
      "Condition": {
        "Bool": {
          "elasticfilesystem:Encrypted": "true"
        }
      },
      "Resource": "*"
    }
  ]
}
```

Wenn diese Richtlinie einem Benutzer zugewiesen wird, der versucht, ein unverschlüsseltes Dateisystem zu erstellen, schlägt die Anforderung fehl. Dem Benutzer wird eine Meldung ähnlich der folgenden angezeigt, unabhängig davon, ob er die AWS-Managementkonsole, die oder die AWS CLI AWS API oder das SDK verwendet:

```
User: arn:aws:iam::111122223333:user/username is not authorized to
perform: elasticfilesystem:CreateFileSystem on the specified resource.
```

Beispiel: Erzwingen der Erstellung unverschlüsselten Dateisysteme

Das folgende Beispiel zeigt eine identitätsbasierte Richtlinie, die Prinzipals autorisiert, nur unverschlüsselte Dateisysteme zu erstellen.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "elasticfilesystem:CreateFileSystem",
      "Condition": {
```

```
        "Bool": {
            "elasticfilesystem:Encrypted": "false"
        },
        "Resource": "*"
    }
}
```

Wenn diese Richtlinie einem Benutzer zugewiesen wird, der versucht, ein verschlüsseltes Dateisystem zu erstellen, schlägt die Anforderung fehl. Dem Benutzer wird eine Meldung ähnlich der folgenden angezeigt, unabhängig davon, ob er die AWS-Managementkonsole AWS CLI, die oder die AWS API oder das SDK verwendet:

```
User: arn:aws:iam::111122223333:user/username is not authorized to
perform: elasticfilesystem:CreateFileSystem on the specified resource.
```

Sie können auch die Erstellung verschlüsselter oder unverschlüsselter EFS-Dateisysteme erzwingen, indem Sie eine AWS Organizations Service Control Policy (SCP) erstellen. Weitere Informationen zu Dienststeuerungsrichtlinien finden Sie unter [Dienststeuerungsrichtlinien](#) im AWS Organizations Benutzerhandbuch. AWS Organizations

Beispiele für ressourcenbasierte Richtlinien für Amazon EFS

In diesem Abschnitt finden Sie Beispielrichtlinien für Dateisysteme, die Berechtigungen für verschiedene Amazon-EFS-Aktionen erteilen oder verweigern. Die Zeichenbeschränkung von EFS-Dateisystemrichtlinien liegt bei 20.000. Hinweise zu den Elementen einer ressourcenbasierten Richtlinie finden Sie unter [Ressourcenbasierte Richtlinien in Amazon EFS](#).

Important

Wenn Sie einem einzelnen IAM-Benutzer oder einer einzelnen IAM-Rolle in einer Dateisystemrichtlinie Berechtigungen erteilen, sollten Sie diesen Benutzer oder diese Rolle nicht löschen oder neu erstellen, solange die Richtlinie noch auf dem Dateisystem gültig ist. Wenn dies der Fall ist, wird dieser Benutzer oder diese Rolle effektiv für das Dateisystem gesperrt und kann nicht darauf zugreifen. Weitere Informationen finden Sie unter [Angaben eines Prinzipals](#) im IAM-Benutzerhandbuch.

Informationen zum Erstellen von Richtlinien für ein Dateisystem finden Sie unter [Erstellen von Dateisystemrichtlinien](#).

Themen

- [Beispiel: Erteilen Sie einer bestimmten Rolle Lese- und Schreibzugriff AWS](#)
- [Beispiel: Erteilen von schreibgeschütztem Zugriff](#)
- [Beispiel: Zugriff auf einen EFS-Zugriffspunkt gewähren](#)

Beispiel: Erteilen Sie einer bestimmten Rolle Lese- und Schreibzugriff AWS

Diese EFS-Dateisystemrichtlinie weist folgende Merkmale auf:

- Der Effekt ist Allow.
- Der Prinzipal ist auf die Testing_Role im AWS-Konto gesetzt.
- Die Aktion ist auf ClientMount (Lesen) und ClientWrite eingestellt.
- Die Bedingung für die Erteilung von Berechtigungen ist auf AccessedViaMountTarget gesetzt.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/Testing_Role"
      },
      "Action": [
        "elasticfilesystem:ClientWrite",
        "elasticfilesystem:ClientMount"
      ],
      "Resource": "arn:aws:elasticfilesystem:us-east-2:111122223333:file-system/fs-1234abcd",
      "Condition": {
        "Bool": {
          "elasticfilesystem:AccessedViaMountTarget": "true"
        }
      }
    }
  ]
}
```

Beispiel: Erteilen von schreibgeschütztem Zugriff

Die folgende Dateisystemrichtlinie gewährt ClientMount der EfsReadOnly IAM-Rolle nur oder nur Leseberechtigungen.

```
{
  "Id": "read-only-example-policy02",
  "Statement": [
    {
      "Sid": "efs-statement-example02",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/EfsReadOnly"
      },
      "Action": [
        "elasticfilesystem:ClientMount"
      ],
      "Resource": "arn:aws:elasticfilesystem:us-east-2:111122223333:file-system/fs-12345678"
    }
  ]
}
```

Weitere Informationen zum Festlegen zusätzlicher Dateisystemrichtlinien, einschließlich der Verweigerung des Root-Zugriffs für alle IAM-Prinzipale, mit Ausnahme einer bestimmten Management Workstation, finden Sie unter [Aktivieren Sie Root-Squashing mithilfe der IAM-Autorisierung für NFS-Clients](#).

Beispiel: Zugriff auf einen EFS-Zugriffspunkt gewähren

Sie verwenden eine EFS-Zugriffsrichtlinie, um einem NFS-Client die anwendungsspezifische Ansicht freigegebener dateibasierter Datensätze auf einem EFS-Dateisystem zu ermöglichen. Sie gewähren die Zugangspunktberechtigungen auf dem Dateisystem mithilfe einer Dateisystemrichtlinie.

In diesem Beispiel für eine Dateirichtlinie wird ein Bedingungelement verwendet, um einem bestimmten Zugangspunkt, der durch seinen ARN definiert ist, uneingeschränkten Zugriff auf das Dateisystem zu gewähren.

Weitere Hinweise zu EFS-Zugangspunkten finden Sie unter [Arbeiten mit Zugriffspunkten](#).

```
{
  "Id": "access-point-example03",
```

```

    "Statement": [
      {
        "Sid": "access-point-statement-example03",
        "Effect": "Allow",
        "Principal": {"AWS": "arn:aws:iam::555555555555:role/
EfsAccessPointFullAccess"},
        "Action": "elasticfilesystem:Client*",
        "Resource": "arn:aws:elasticfilesystem:us-east-2:111122223333:file-system/
fs-12345678",
        "Condition": {
          "StringEquals": {
            "elasticfilesystem:AccessPointArn": "arn:aws:elasticfilesystem:us-
east-2:555555555555:access-point/fsap-12345678" }
          }
        }
      ]
    ]
  }
}

```

AWS verwaltete Richtlinien für Amazon EFS

Eine AWS verwaltete Richtlinie ist eine eigenständige Richtlinie, die von erstellt und verwaltet AWS wird. AWS Verwaltete Richtlinien sind so konzipiert, dass sie Berechtigungen für viele gängige Anwendungsfälle bereitstellen, sodass Sie damit beginnen können, Benutzern, Gruppen und Rollen Berechtigungen zuzuweisen.

Beachten Sie, dass AWS verwaltete Richtlinien für Ihre speziellen Anwendungsfälle möglicherweise keine Berechtigungen mit den geringsten Rechten gewähren, da sie allen AWS Kunden zur Verfügung stehen. Wir empfehlen Ihnen, die Berechtigungen weiter zu reduzieren, indem Sie [vom Kunden verwaltete Richtlinien](#) definieren, die speziell auf Ihre Anwendungsfälle zugeschnitten sind.

Sie können die in AWS verwalteten Richtlinien definierten Berechtigungen nicht ändern. Wenn die in einer AWS verwalteten Richtlinie definierten Berechtigungen AWS aktualisiert werden, wirkt sich das Update auf alle Prinzidentitäten (Benutzer, Gruppen und Rollen) aus, denen die Richtlinie zugeordnet ist. AWS aktualisiert eine AWS verwaltete Richtlinie höchstwahrscheinlich, wenn eine neue Richtlinie eingeführt AWS-Service wird oder neue API-Operationen für bestehende Dienste verfügbar werden.

Weitere Informationen finden Sie unter [Von AWS verwaltete Richtlinien](#) im IAM-Benutzerhandbuch.

AWS verwaltete Richtlinie: AWSService RoleForAmazonElasticFileSystem

Amazon EFS verwendet die benannte dienstbezogene

RolleAWSServiceRoleForAmazonElasticFileSystem, damit Amazon EFS AWS

Ressourcen in Ihrem Namen verwalten kann. Diese Rolle vertraut darauf, dass der `elasticfilesystem.amazonaws.com` Service die Rolle übernimmt. Weitere Informationen finden Sie unter [Verwendung von Service-gebundenen Rollen für Amazon EFS](#).

AWS verwaltete Richtlinie: AmazonElasticFileSystemFullAccess

Sie können die `AmazonElasticFileSystemFullAccess`-Richtlinie an Ihre IAM-Identitäten anfügen.

Diese Richtlinie gewährt Administratorberechtigungen, die den vollen Zugriff auf Amazon EFS und den Zugriff auf zugehörige AWS Dienste über die ermöglichen AWS-Managementkonsole.

Details zu Berechtigungen

Diese Richtlinie umfasst die folgenden Berechtigungen.

- `elasticfilesystem` — Ermöglicht es Prinzipalen, alle Aktionen in der Amazon-EFS-Konsole durchzuführen. Außerdem können Prinzipale Backups erstellen (`elasticfilesystem:Backup`) und wiederherstellen (`elasticfilesystem:Restore`) mithilfe von AWS Backup.
- `cloudwatch` — Ermöglicht Prinzipalen, CloudWatch Amazon-Dateisystem-Metriken und Alarme für eine Metrik in der Amazon EFS-Konsole zu beschreiben.
- `ec2` — Ermöglicht Prinzipalen das Erstellen, Löschen und Beschreiben von Netzwerkschnittstellen, das Beschreiben und Ändern von Netzwerkschnittstellenattributen, das Beschreiben von Availability Zones, Sicherheitsgruppen, Subnetzen, virtuellen privaten Clouds (VPCs) und VPC-Attributen, die mit einem EFS-Dateisystem verknüpft sind, in der Amazon EFS-Konsole.
- `kms` — Ermöglicht Prinzipalen, Aliase für AWS Key Management Service (AWS KMS) -Schlüssel aufzulisten und KMS-Schlüssel in der Amazon EFS-Konsole zu beschreiben.
- `iam` — Erteilt die Berechtigung zum Erstellen einer serviceverknüpften Rolle, die es Amazon EFS ermöglicht, AWS Ressourcen im Namen des Benutzers zu verwalten.
- `iam:PassRole` — Erteilt die Erlaubnis, eine IAM-Rolle an Amazon EFS zu übergeben.

Die Berechtigungen für diese Richtlinie finden Sie [AmazonElasticFileSystemFullAccess](#) im Referenzhandbuch für AWS verwaltete Richtlinien.

AWS verwaltete Richtlinie: AmazonElasticFileSystemReadOnlyAccess

Sie können die `AmazonElasticFileSystemReadOnlyAccess`-Richtlinie an Ihre IAM-Identitäten anfügen.

Diese Richtlinie gewährt nur Lesezugriff auf Amazon EFS über die AWS-Managementkonsole.

Details zu Berechtigungen

Diese Richtlinie umfasst die folgenden Berechtigungen.

- `elasticfilesystem` – Ermöglicht es Prinzipalen, Attribute von Amazon-EFS-Dateisystemen zu beschreiben, einschließlich Kontoeinstellungen, Backup- und Dateisystemrichtlinien, Lebenszykluskonfiguration, Mounting-Ziele und deren Sicherheitsgruppen, Tags und Zugangspunkte in der Amazon-EFS-Konsole.
- `cloudwatch`— Ermöglicht Prinzipalen das Abrufen von CloudWatch Metriken und das Beschreiben von Alarmen für Metriken in der Amazon EFS-Konsole.
- `ec2`— Ermöglicht Prinzipalen, Availability Zones, Netzwerkschnittstellen und ihre Attribute, Sicherheitsgruppen, Subnetze VPCs und ihre Attribute in der Amazon EFS-Konsole anzuzeigen.
- `kms`— Ermöglicht Prinzipalen, Aliase für AWS KMS Schlüssel in der Amazon EFS-Konsole aufzulisten.

Informationen zu den Berechtigungen für diese Richtlinie finden Sie

[AmazonElasticFileSystemReadOnlyAccess](#) im Referenzhandbuch für AWS verwaltete Richtlinien.

AWS verwaltete Richtlinie: AmazonElasticFileSystemClientFullAccess

Sie können die AmazonElasticFileSystemClientFullAccess-Richtlinie an eine IAM-Entität anhängen.

Diese Richtlinie gewährt Lese- und Schreibclientzugriff auf EFS-Dateisysteme. Diese Richtlinie ermöglicht NFS-Clients das Mounten, Lesen und Schreiben in EFS-Dateisysteme.

Informationen zu den Berechtigungen für diese Richtlinie finden Sie

[AmazonElasticFileSystemClientFullAccess](#) im Referenzhandbuch für AWS verwaltete Richtlinien.

AWS verwaltete Richtlinie: AmazonElasticFileSystemClientReadWriteAccess

Sie können die AmazonElasticFileSystemClientReadWriteAccess-Richtlinie an eine IAM-Entität anhängen.

Diese Richtlinie gewährt Lese- und Schreibclientzugriff auf EFS-Dateisysteme. Diese Richtlinie ermöglicht NFS-Clients das Mounten, Lesen und Schreiben in EFS-Dateisysteme.

Informationen zu den Berechtigungen für diese Richtlinie finden Sie [AmazonElasticFileSystemClientReadWriteAccess](#) im Referenzhandbuch für AWS verwaltete Richtlinien.

Amazon EFS-Updates für AWS verwaltete Richtlinien

Sehen Sie sich Details zu Aktualisierungen der AWS verwalteten Richtlinien für Amazon EFS an, seit dieser Service begonnen hat, diese Änderungen zu verfolgen. Um automatische Warnungen über Änderungen an dieser Seite erhalten, abonnieren Sie den RSS-Feed auf der [Dokumentverlauf](#)-Seite.

Änderungen	Beschreibung	Date
Zur Aktualisierung einer bestehenden Richtlinie	Richtlinie: AmazonElasticFileSystemFullAccess Amazon EFS hat Folgendes hinzugefügt: • <code>ReplicationRead</code> und <code>ReplicationWrite</code> um die Erlaubnis zu erteilen, Dateisystemdaten für die Replikation zu lesen und zu schreiben. • <code>iam:PassRole</code> um Amazon EFS die Erlaubnis zu erteilen, Replikationskonfigurationen zu erstellen.	7. November 2024
Zur Aktualisierung einer bestehenden Richtlinie	Richtlinie: AmazonElasticFileSystemServiceRolePolicy Amazon EFS wurde hinzugefügt <code>ReplicationRead</code> <code>ReplicationWrite</code> , um die Erlaubnis zum Lesen und Schreiben von Dateisystemdaten für die Replikation zu erteilen.	7. November 2024
Zur Aktualisierung einer bestehenden Richtlinie	Richtlinie: AmazonElasticFileSystemReadOnlyAccess Amazon EFS hat die <code>ReplicationRead</code> Aktion hinzugefügt, um die Erlaubnis zum Lesen von Dateisystemdaten für die Replikation zu erteilen.	7. November 2024

Änderungen	Beschreibung	Date
Zur Aktualisierung einer bestehenden Richtlinie	Richtlinie: AmazonElasticFileSystemReadOnlyAccess Amazon EFS hat neue Berechtigungen hinzugefügt, die Quell- und Zielkonten Zugriff auf Dateisysteme für kontenübergreifende Replikationen gewähren.	7. August 2024
Zur Aktualisierung einer bestehenden Richtlinie	Richtlinie: AmazonElasticFileSystemFullAccess Amazon EFS hat eine neue Berechtigung hinzugefügt, die es Prinzipalen ermöglicht, den Schutz für ein Dateisystem zu deaktivieren und zu aktivieren. Die Berechtigungen sind erforderlich, damit Amazon EFS in ein vorhandenes Dateisystem replizieren kann.	27. November 2023
Zur Aktualisierung einer bestehenden Richtlinie	Richtlinie: AmazonElasticFileSystemServiceRolePolicy Amazon EFS hat neue Berechtigungen hinzugefügt, die es Prinzipalen ermöglichen, Amazon-EFS-Replikationen zu erstellen, zu beschreiben und zu löschen sowie Amazon-EFS-Dateisysteme zu erstellen. Die Berechtigungen sind erforderlich, damit Amazon EFS die Konfiguration der Dateisystemreplikation im Namen des Benutzers verwalten kann.	25 Januar 2022
Zur Aktualisierung einer bestehenden Richtlinie	Richtlinie: AmazonElasticFileSystemReadOnlyAccess Amazon EFS hat eine neue Berechtigung hinzugefügt, die es Prinzipalen erlaubt, Amazon-EFS-Replikationen zu beschreiben. Die Berechtigungen sind erforderlich, damit Benutzer die Replikationskonfigurationen des Dateisystems einsehen können.	25 Januar 2022

Änderungen	Beschreibung	Date
Zur Aktualisierung einer bestehenden Richtlinie	Richtlinie: AmazonElasticFileSystemFullAccess Amazon EFS hat neue Berechtigungen hinzugefügt, die es Prinzipalen ermöglichen, Amazon-EFS-Replikationen zu erstellen, zu beschreiben und zu löschen. Die Berechtigungen sind erforderlich, damit Benutzer die Replikationskonfigurationen des Dateisystems verwalten können.	25 Januar 2022
Änderungsverfolgung gestartet	Richtlinie: AmazonElasticFileSystemClientReadWriteAccess Gewährt NFS-Clients Lese- und Schreibrechte auf Amazon-EFS-Dateisystemen.	3. Januar 2022
Änderungsverfolgung gestartet	Richtlinie: AmazonElasticFileSystemServiceRolePolicy Serviceverknüpfte Rollenberechtigungen für Amazon EFS.	8. Oktober 2021
Zur Aktualisierung einer bestehenden Richtlinie	Richtlinie: AmazonElasticFileSystemFullAccess Amazon EFS hat neue Berechtigungen hinzugefügt, um es den Prinzipalen zu ermöglichen, die Amazon-EFS-Kontoeinstellungen zu ändern und zu beschreiben. Die Berechtigungen sind erforderlich, damit Benutzer die Kontoeinstellungen in der Amazon-EFS-Konsole anzeigen und festlegen können.	7. Mai 2021

Änderungen	Beschreibung	Date
Zur Aktualisierung einer bestehenden Richtlinie	Richtlinie: AmazonElasticFileSystemReadOnlyAccess Amazon EFS hat neue Berechtigungen hinzugefügt, die es Prinzipalen ermöglichen, Amazon-EFS-Kontoeinstellungen zu beschreiben. Die Berechtigungen sind erforderlich, damit Benutzer die Einstellungen für die Kontoeinstellungen in der Amazon-EFS-Konsole anzeigen können.	7. Mai 2021
Amazon EFS hat mit der Nachverfolgung von Änderungen begonnen	Amazon EFS hat damit begonnen, Änderungen für seine AWS verwalteten Richtlinien nachzuverfolgen.	7. Mai 2021

Verwendung von Tags mit Amazon EFS

Sie können Tags verwenden, um den Zugriff auf Amazon-EFS-Ressourcen zu steuern und die attributbasierte Zugriffskontrolle (ABAC) zu implementieren. Weitere Informationen finden Sie unter:

- [Taggen von EFS-Ressourcen](#)
- [Bestimmung des Zugriffs auf Ressourcen basierend auf Tags](#)
- [Wofür ist ABAC? AWS](#) im IAM-Benutzerhandbuch

Note

Die Amazon-EFS-Replikation unterstützt die Verwendung von Tags für die attributbasierte Zugriffskontrolle (ABAC) nicht.

Um während der Erstellung Tags auf Amazon-EFS-Ressourcen anzuwenden, müssen Benutzer über bestimmte AWS Identity and Access Management -(IAM-) Berechtigungen verfügen.

Erteilen von Berechtigungen zum Markieren von Ressourcen während der Erstellung

Mit den folgenden Tag-on create Amazon EFS API-Aktionen können Sie Tags angeben, wenn Sie die Ressource erstellen.

- `CreateAccessPoint`
- `CreateFileSystem`

Damit Benutzer Ressourcen bei der Erstellung markieren können, müssen sie über die Berechtigung zur Verwendung der Aktion verfügen, mit der die Ressourcen erstellt werden, wie z. B. `elasticfilesystem:CreateAccessPoint` oder `elasticfilesystem:CreateFileSystem`. Wenn bei der Aktion zur Erstellung von Ressourcen Tags angegeben wurden, AWS führt dieser Vorgang eine zusätzliche Autorisierung durch, um zu überprüfen, ob Benutzer berechtigt sind, Tags zu erstellen. `elasticfilesystem:TagResource` Daher benötigen die Benutzer außerdem die expliziten Berechtigungen zum Verwenden der `elasticfilesystem:TagResource`-Aktion.

Verwenden Sie in der IAM-Richtliniendefinition für die `elasticfilesystem:TagResource`-Aktion das Condition-Element mit dem `elasticfilesystem:CreateAction`-Bedingungsschlüssel, um der Aktion, die die Ressource erstellt, Markierungsberechtigungen zu erteilen.

Example Richtlinie: Das Hinzufügen von Tags zu Dateisystemen nur zum Zeitpunkt der Erstellung erlauben

Mit der folgenden Beispielrichtlinie können Benutzer nur während der Erstellung Dateisysteme erstellen und ihnen Tags zuordnen. Die Markierung von bestehenden Ressourcen durch die Benutzer ist nicht zulässig. (Sie können die `elasticfilesystem:TagResource`-Aktion nicht direkt aufrufen.)

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticfilesystem:CreateFileSystem"
      ],
      "Resource": "arn:aws:elasticfilesystem:region:account-id:file-system/*"
    },
    {
      "Effect": "Allow",
      "Action": [
```

```
    "elasticfilesystem:TagResource"
  ],
  "Resource": "arn:aws:elasticfilesystem:region:account-id:file-system/*",
  "Condition": {
    "StringEquals": {
      "elasticfilesystem:CreateAction": "CreateFileSystem"
    }
  }
}
```

Verwendung von Tags zur Steuerung des Zugriffs auf Ihre Amazon-EFS-Ressourcen

Um den Zugriff auf Amazon-EFS-Ressourcen und -Aktionen zu kontrollieren, können Sie IAM-Richtlinien verwenden, die auf Tags basieren. Sie können diese Kontrolle auf zwei Arten ausüben:

- Sie können den Zugriff auf Amazon-EFS-Ressourcen anhand der Tags auf diesen Ressourcen steuern.
- Sie können steuern, welche Tags in einer IAM-Anforderungsbedingung übergeben werden können.

Informationen zur Verwendung von Tags zur Steuerung des Zugriffs auf AWS Ressourcen finden Sie unter [Steuern des Zugriffs mithilfe von Tags](#) im IAM-Benutzerhandbuch.

Bestimmung des Zugriffs auf Ressourcen basierend auf Tags

Um zu steuern, welche Aktionen ein Benutzer oder eine Rolle mit einer Amazon-EFS-Ressource durchführen kann, können Sie Tags für die Ressource verwenden. So können Sie beispielsweise bestimmte API-Vorgänge für eine Dateisystemressource auf der Grundlage des Schlüssel-Wert-Paares des Tags der Ressource zulassen oder verbieten.

Example Richtlinie: Ein Dateisystem nur dann erstellen, wenn ein bestimmtes Tag verwendet wird

Die folgende Beispielrichtlinie erlaubt es dem Benutzer, ein Dateisystem nur dann zu erstellen, wenn er es mit einem bestimmten Tag-Schlüssel-Wert-Paar markiert, in diesem Beispiel key=Department, value=Finance.

```
{
  "Effect": "Allow",
  "Action": [
```

```

    "elasticfilesystem:CreateFileSystem",
    "elasticfilesystem:TagResource"
  ],
  "Resource": "arn:aws:elasticfilesystem:region:account-id:file-system/*",
  "Condition": {
    "StringEquals": {
      "aws:RequestTag/Department": "Finance"
    }
  }
}

```

Example Richtlinie: Dateisysteme mit bestimmten Tags löschen

Die folgende Beispielrichtlinie erlaubt es einem Benutzer, nur Dateisysteme zu löschen, die mit Department=Finance gekennzeichnet sind.

JSON

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticfilesystem:DeleteFileSystem"
      ],
      "Resource": "arn:aws:elasticfilesystem:us-east-1:111122223333:file-  
system/*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/Department": "Finance"
        }
      }
    }
  ]
}

```

Verwendung von Service-gebundenen Rollen für Amazon EFS

Amazon EFS verwendet eine AWS Identity and Access Management (IAM) [-Serviceverknüpfte Rolle](#). Die servicegebundene Amazon-EFS-Rolle ist eine einzigartige Art von IAM-Rolle, die direkt

mit Amazon EFS verknüpft ist. Die vordefinierte serviceverknüpfte Amazon EFS-Rolle umfasst Berechtigungen, die der Service benötigt, um andere in AWS-Services Ihrem Namen anzurufen.

Eine serviceverknüpfte Rolle erleichtert die Einrichtung von Amazon EFS, da Sie die erforderlichen Berechtigungen nicht manuell hinzufügen müssen. Amazon EFS definiert die Berechtigungen seiner servicegebundenen Rolle, und nur Amazon EFS kann seine Rolle übernehmen. Die definierten Berechtigungen umfassen die Vertrauens- und Berechtigungsrichtlinie. Diese Berechtigungsrichtlinie kann keinen anderen IAM-Entitäten zugewiesen werden.

Sie können die mit dem Amazon-EFS-Service verknüpfte Rolle erst löschen, nachdem Sie zuvor Ihre Amazon-EFS-Dateisysteme gelöscht haben. Dies schützt Ihre Amazon-EFS-Ressourcen, da Sie nicht versehentlich die Berechtigung zum Zugriff auf die Ressourcen entziehen können.

Die serviceverknüpfte Rolle ermöglicht es, dass alle API-Aufrufe sichtbar sind. AWS CloudTrail Dies hilft bei Überwachungs- und Prüfungsanforderungen, da Sie alle Aktionen, die Amazon EFS in Ihrem Namen durchführt, nachverfolgen können. Weitere Informationen finden Sie unter [Protokolleinträge für serviceverknüpfte EFS-Rollen](#).

Weitere Informationen finden Sie unter [serviceverknüpfte Rollenberechtigung](#) im IAM-Benutzerhandbuch.

Service-gebundene Rollenberechtigungen für Amazon EFS

Amazon EFS verwendet die benannte dienstbezogene Rolle `AWSServiceRoleForAmazonElasticFileSystem`, damit Amazon EFS AWS Ressourcen im Namen Ihrer EFS-Dateisysteme aufrufen und verwalten kann.

Die `AWSServiceRoleForAmazonElasticFileSystem` serviceverknüpfte Rolle vertraut darauf, dass `elasticfilesystem.amazonaws.com` sie die Rolle übernimmt.

Die Richtlinie für Rollenberechtigungen ermöglicht es Amazon EFS, die in der JSON-Definition der Richtlinie enthaltenen Aktionen auszuführen:

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
```

```

        "backup-storage:MountCapsule",
        "ec2:CreateNetworkInterface",
        "ec2:DeleteNetworkInterface",
        "ec2:DescribeSecurityGroups",
        "ec2:DescribeSubnets",
        "ec2:DescribeNetworkInterfaceAttribute",
        "ec2:ModifyNetworkInterfaceAttribute",
        "tag:GetResources"
    ],
    "Resource": "*"
},
{
    "Effect": "Allow",
    "Action": [
        "kms:DescribeKey"
    ],
    "Resource": "arn:aws:kms:*:*:key/*"
},
{
    "Effect": "Allow",
    "Action": [
        "backup:CreateBackupVault",
        "backup:PutBackupVaultAccessPolicy"
    ],
    "Resource": [
        "arn:aws:backup:*:*:backup-vault:aws/efs/automatic-backup-vault"
    ]
},
{
    "Effect": "Allow",
    "Action": [
        "backup:CreateBackupPlan",
        "backup:CreateBackupSelection"
    ],
    "Resource": [
        "arn:aws:backup:*:*:backup-plan:*"
    ]
},
{
    "Effect": "Allow",
    "Action": [
        "iam:CreateServiceLinkedRole"
    ],
    "Resource": "*",

```

```

        "Condition": {
            "StringEquals": {
                "iam:AWSServiceName": [
                    "backup.amazonaws.com"
                ]
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "iam:PassRole"
            ],
            "Resource": [
                "arn:aws:iam::*:role/aws-service-role/backup.amazonaws.com/AWSServiceRoleForBackup"
            ],
            "Condition": {
                "StringLike": {
                    "iam:PassedToService": "backup.amazonaws.com"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "elasticfilesystem:DescribeFileSystems",
                "elasticfilesystem:CreateReplicationConfiguration",
                "elasticfilesystem:DescribeReplicationConfigurations",
                "elasticfilesystem>DeleteReplicationConfiguration",
                "elasticfilesystem:ReplicationRead",
                "elasticfilesystem:ReplicationWrite"
            ],
            "Resource": "*"
        }
    ]
}

```

 Note

Sie müssen die IAM-Berechtigungen manuell konfigurieren, AWS KMS wenn Sie ein neues EFS-Dateisystem erstellen, das im Ruhezustand verschlüsselt ist. Weitere Informationen hierzu finden Sie unter [Verschlüsseln von Daten im Ruhezustand](#).

Sie müssen Berechtigungen konfigurieren, damit eine juristische Stelle von IAM (z. B. Benutzer, Gruppe oder Rolle) eine serviceverknüpfte Rolle erstellen, bearbeiten oder löschen kann. Weitere Informationen finden Sie unter [Dienstbezogene Rollenberechtigungen](#) im IAM-Benutzerhandbuch.

Erstellen einer serviceverknüpften Rolle für Amazon EFS

In den meisten Fällen müssen Sie eine dienstverknüpfte Rolle nicht manuell erstellen. Wenn Sie Mount-Ziele oder eine Replikationskonfiguration für Ihr EFS-Dateisystem in der AWS-Managementkonsole, der AWS CLI oder der AWS API erstellen, erstellt Amazon EFS die serviceverknüpfte Rolle für Sie.

Wenn Sie diese service-linked-role Datei manuell löschen und dann erneut erstellen müssen, können Sie die Rolle außerdem mit demselben Verfahren in Ihrem Konto neu erstellen. Wenn Sie Bereitstellungsziele oder eine Replikationskonfiguration für Ihr EFS-Dateisystem erstellen, erstellt Amazon EFS die serviceverknüpfte Rolle für Sie.

Wenn Amazon EFS die serviceverknüpfte Rolle jedoch nicht erstellt service-linked-role oder wenn Sie mit der Nutzung von Amazon EFS begonnen haben, bevor es serviceverknüpfte Rollen unterstützt hat, können Sie die serviceverknüpfte Rolle manuell erstellen. Anweisungen finden Sie im [IAM-Benutzerhandbuch unter Erstellen einer serviceverknüpften Rolle](#).

Bearbeiten einer serviceverknüpften Rolle für Amazon EFS

Amazon EFS erlaubt es Ihnen nicht, die mit dem `AWSServiceRoleForAmazonElasticFileSystem` Service verknüpfte Rolle zu bearbeiten. Da möglicherweise verschiedene Entitäten auf die Rolle verweisen, kann der Rollename nach dem Erstellen einer serviceverknüpften Rolle nicht mehr geändert werden. Sie können jedoch die Beschreibung der Rolle mit IAM bearbeiten. Weitere Informationen finden Sie im [IAM-Benutzerhandbuch unter Aktualisieren einer serviceverknüpften Rolle](#).

Löschen einer mit einem Service verknüpften Rolle für Amazon EFS

Wenn Sie ein Feature oder einen Service, die bzw. der eine serviceverknüpfte Rolle erfordert, nicht mehr benötigen, sollten Sie diese Rolle löschen. Auf diese Weise haben Sie keine ungenutzte Entität, die nicht aktiv überwacht oder verwaltet wird. Sie müssen jedoch die Ressourcen für Ihre serviceverknüpfte Rolle zunächst bereinigen, bevor Sie sie manuell löschen können. Weitere Informationen finden Sie unter [Säubern Sie Ressourcen und schützen Sie Ihr AWS Konto](#).

Note

Wenn der Amazon-EFS-Service die Rolle verwendet, wenn Sie versuchen, die Ressourcen zu löschen, kann der Löschvorgang fehlschlagen. Wenn dies passiert, warten Sie einige Minuten und versuchen Sie es erneut.

Verwenden Sie die IAM-Konsole, die oder die AWS API AWS CLI, um die dienstverknüpfte Rolle zu löschen. `AWSServiceRoleForAmazonElasticFileSystem` Weitere Informationen finden Sie unter [Löschen einer serviceverknüpften Rolle](#) im IAM-Benutzerhandbuch.

Unterstützte Regionen für serviceverknüpfte Amazon EFS-Rollen

Amazon EFS unterstützt die Verwendung von serviceverknüpften Rollen überall AWS-Regionen dort, wo der Service verfügbar ist. Weitere Informationen finden Sie unter [AWS Service-Endpunkte](#) im Allgemeine AWS-Referenz Benutzerhandbuch.

Fehlerbehebung für Amazon-Elastic-File-System-Identität und -Zugriff

Diagnostizieren und beheben Sie mithilfe der folgenden Informationen gängige Probleme, die bei der Verwendung von Amazon EFS und IAM auftreten können.

Themen

- [Ich bin nicht autorisiert, eine Aktion in Amazon EFS auszuführen](#)
- [Ich bin nicht berechtigt, iam auszuführen: PassRole](#)
- [Ich möchte Personen außerhalb von mir den Zugriff AWS-Konto auf meine Amazon EFS-Ressourcen ermöglichen](#)

Ich bin nicht autorisiert, eine Aktion in Amazon EFS auszuführen

Wenn Sie eine Fehlermeldung erhalten, dass Sie nicht zur Durchführung einer Aktion berechtigt sind, müssen Ihre Richtlinien aktualisiert werden, damit Sie die Aktion durchführen können.

Der folgende Beispielfehler tritt auf, wenn der IAM-Benutzer `mateojackson` versucht, über die Konsole Details zu einer fiktiven *my-example-widget*-Ressource anzuzeigen, jedoch nicht über `elasticfilesystem:GetWidget`-Berechtigungen verfügt.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
elasticfilesystem:GetWidget on resource: my-example-widget
```

In diesem Fall muss die Richtlinie für den Benutzer `mateojackson` aktualisiert werden, damit er mit der `elasticfilesystem:GetWidget`-Aktion auf die *my-example-widget*-Ressource zugreifen kann.

Wenn Sie Hilfe benötigen, wenden Sie sich an Ihren AWS Administrator. Ihr Administrator hat Ihnen Ihre Anmeldeinformationen zur Verfügung gestellt.

Ich bin nicht berechtigt, iam auszuführen: PassRole

Wenn Sie die Fehlermeldung erhalten, dass Sie nicht zur Ausführung der `iam:PassRole`-Aktion autorisiert sind, müssen Ihre Richtlinien aktualisiert werden, um eine Rolle an Amazon EFS übergeben zu können.

Einige AWS-Services ermöglichen es Ihnen, eine bestehende Rolle an diesen Dienst zu übergeben, anstatt eine neue Servicerolle oder eine dienstverknüpfte Rolle zu erstellen. Hierzu benötigen Sie Berechtigungen für die Übergabe der Rolle an den Dienst.

Der folgende Beispielfehler tritt auf, wenn ein IAM-Benutzer mit dem Namen `marymajor` versucht, die Konsole zu verwenden, um eine Aktion in Amazon EFS auszuführen. Die Aktion erfordert jedoch, dass der Service über Berechtigungen verfügt, die durch eine Servicerolle gewährt werden. Mary besitzt keine Berechtigungen für die Übergabe der Rolle an den Dienst.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
iam:PassRole
```

In diesem Fall müssen die Richtlinien von Mary aktualisiert werden, um die Aktion `iam:PassRole` ausführen zu können.

Wenn Sie Hilfe benötigen, wenden Sie sich an Ihren AWS Administrator. Ihr Administrator hat Ihnen Ihre Anmeldeinformationen zur Verfügung gestellt.

Ich möchte Personen außerhalb von mir den Zugriff AWS-Konto auf meine Amazon EFS-Ressourcen ermöglichen

Sie können eine Rolle erstellen, mit der Benutzer in anderen Konten oder Personen außerhalb Ihrer Organisation auf Ihre Ressourcen zugreifen können. Sie können festlegen, wem die Übernahme der Rolle anvertraut wird. Für Dienste, die ressourcenbasierte Richtlinien oder Zugriffskontrolllisten (ACLs) unterstützen, können Sie diese Richtlinien verwenden, um Personen Zugriff auf Ihre Ressourcen zu gewähren.

Weitere Informationen dazu finden Sie hier:

- Informationen dazu, ob Amazon EFS diese Features unterstützt, finden Sie unter [Funktionsweise von Amazon Elastic File System mit IAM](#).
- Informationen dazu, wie Sie Zugriff auf Ihre Ressourcen in AWS-Konten Ihrem Besitz gewähren können, finden Sie im IAM-Benutzerhandbuch unter [Gewähren des Zugriffs für einen IAM-Benutzer in einem anderen AWS-Konto, dem Sie gehören](#).
- Informationen dazu, wie Sie Dritten Zugriff auf Ihre Ressourcen gewähren können AWS-Konten, finden Sie [AWS-Konten im IAM-Benutzerhandbuch unter Gewähren des Zugriffs für Dritte](#).
- Informationen dazu, wie Sie über einen Identitätsverbund Zugriff gewähren, finden Sie unter [Gewähren von Zugriff für extern authentifizierte Benutzer \(Identitätsverbund\)](#) im IAM-Benutzerhandbuch.
- Informationen zum Unterschied zwischen der Verwendung von Rollen und ressourcenbasierten Richtlinien für den kontoübergreifenden Zugriff finden Sie unter [Kontoübergreifender Ressourcenzugriff in IAM](#) im IAM-Benutzerhandbuch.

Verwendung von IAM zur Steuerung des Zugriffs auf Dateisysteme

Sie können sowohl IAM-Identitätsrichtlinien als auch Ressourcenrichtlinien verwenden, um den Client-Zugriff auf Amazon-EFS-Ressourcen auf eine Weise zu steuern, die skalierbar und für Cloud-Umgebungen optimiert ist. Mit IAM können Sie Clients erlauben, bestimmte Aktionen auf einem Dateisystem durchzuführen, einschließlich Lese-, Schreib- und Root-Zugriff. Eine „allow“-Erlaubnis für eine Aktion in einer IAM-Identitätsrichtlinie oder einer Dateisystem-Ressourcenrichtlinie erlaubt den Zugriff auf diese Aktion. Die Genehmigung muss nicht sowohl in einer Identitäts- als auch in einer Ressourcenrichtlinie erteilt werden.

NFS-Klienten können sich mit einer IAM-Rolle identifizieren, wenn sie sich mit einem EFS-Dateisystem verbinden. Wenn ein Client eine Verbindung zu einem Dateisystem herstellt, wertet Amazon EFS die IAM-Ressourcenrichtlinie des Dateisystems, die als Dateisystemrichtlinie bezeichnet wird, zusammen mit allen identitätsbasierten IAM-Richtlinien aus, um die entsprechenden Zugriffsberechtigungen für das Dateisystem zu bestimmen.

Wenn Sie die IAM-Autorisierung für NFS-Klienten verwenden, werden Client-Verbindungen und IAM-Autorisierungsentscheidungen in AWS CloudTrail protokolliert. Weitere Informationen zum Protokollieren von Amazon EFS-API-Aufrufen mit CloudTrail finden Sie unter [Protokollieren von Amazon EFS-API-Aufrufen mit AWS CloudTrail](#).

Important

Sie müssen den EFS-Mount-Helper verwenden, um Ihre EFS-Dateisysteme zu mounten, um die IAM-Autorisierung zur Steuerung des Client-Zugriffs zu verwenden. Weitere Informationen finden Sie unter [Mounting mit IAM-Autorisierung](#).

Standard-EFS-Dateisystemrichtlinie

Die standardmäßige EFS-Dateisystemrichtlinie verwendet IAM nicht zur Authentifizierung und gewährt jedem anonymen Client, der über ein Mounting-Ziel eine Verbindung mit dem Dateisystem herstellen kann, Vollzugriff. Die Standardrichtlinie ist immer dann in Kraft, wenn eine vom Benutzer konfigurierte Dateisystemrichtlinie nicht in Kraft ist, auch bei der Erstellung des Dateisystems. Wenn die Standard-Dateisystemrichtlinie in Kraft ist, gibt eine [DescribeFileSystemPolicy](#)-API-Operation eine `PolicyNotFound`-Antwort zurück.

EFS-Aktionen für Clients

Sie können die folgenden Aktionen für Clients festlegen, die über eine Dateisystemrichtlinie auf ein Dateisystem zugreifen.

Action	Description
<code>elasticfilesystem:ClientMount</code>	Ermöglicht den Nur-Lese-Zugriff auf ein Dateisystem.
<code>elasticfilesystem:ClientWrite</code>	Bietet Schreibrechte für ein Dateisystem.

Action	Description
<code>elasticfilesystem:ClientRootAccess</code>	Ermöglicht die Verwendung des Root-Benutzers beim Zugriff auf ein Dateisystem.

EFS-Bedingungsschlüssel für Clients

Bedingungen werden mithilfe vordefinierter Bedingungsschlüssel formuliert. Amazon EFS verfügt über die folgenden vordefinierten Bedingungsschlüssel für NFS-Clients. Alle anderen Bedingungsschlüssel werden nicht erzwungen, wenn Sie IAM-Kontrollen verwenden, um den Zugriff auf EFS-Dateisysteme zu sichern.

EFS-Bedingungsschlüssel	Description	Operator
<code>aws:SecureTransport</code>	Verwenden Sie diesen Schlüssel, um Clients zu verpflichten, TLS zu verwenden, wenn sie sich mit einem EFS-Dateisystem verbinden.	Boolesch
<code>aws:SourceIp</code>	Verwenden Sie diesen Schlüssel, um die IP-Adresse des Anforderers mit der IP-Adresse zu vergleichen, die Sie in der Richtlinie angeben. Der <code>aws:SourceIp</code> -Bedingungsschlüssel kann nur für öffentliche IP-Adressbereiche verwendet werden.	Zeichenfolge
<code>elasticfilesystem:AccessPointArn</code>	ARN des EFS-Zugangspunkts, mit dem sich der Client verbindet.	Zeichenfolge

EFS-Bedingungsschlüssel	Description	Operator
<code>elasticfilesystem:AccessedViaMountTarget</code>	Verwenden Sie diesen Schlüssel, um den Zugriff auf ein EFS-Dateisystem durch Clients zu verhindern, die keine Dateisystem-Mountinghilfe verwenden.	Boolesch

Beispiele für Dateisystemrichtlinien

Beispiele für Amazon-EFS-Dateisystemrichtlinien finden Sie unter [Beispiele für ressourcenbasierte Richtlinien für Amazon EFS](#).

Konformitätsprüfung für Amazon EFS

Informationen darüber, ob AWS-Service ein [AWS-Services in den Geltungsbereich bestimmter Compliance-Programme fällt](#), finden Sie unter [Umfang nach Compliance-Programm AWS-Services unter](#) . Wählen Sie dort das Compliance-Programm aus, an dem Sie interessiert sind. Allgemeine Informationen finden Sie unter [AWS Compliance-Programme AWS](#) .

Sie können Prüfberichte von Drittanbietern unter [herunterladen AWS Artifact](#). Weitere Informationen finden Sie unter [Berichte herunterladen unter](#) .

Ihre Verantwortung für die Einhaltung der Vorschriften bei der Nutzung AWS-Services hängt von der Vertraulichkeit Ihrer Daten, den Compliance-Zielen Ihres Unternehmens und den geltenden Gesetzen und Vorschriften ab. Weitere Informationen zu Ihrer Verantwortung für die Einhaltung der Vorschriften bei der Nutzung AWS-Services finden Sie in der [AWS Sicherheitsdokumentation](#).

Resilienz in Amazon EFS

Die AWS globale Infrastruktur basiert auf AWS-Regionen Availability Zones (AZs). AWS-Regionen stellen mehrere physisch getrennte und isolierte Netzwerke bereit AZs, die über Netzwerke mit niedriger Latenz, hohem Durchsatz und hoher Redundanz miteinander verbunden sind. Mit können Sie Anwendungen und Datenbanken entwerfen und betreiben AZs, die automatisch und ohne Unterbrechung ein Failover zwischen Zonen durchführen. AZs sind hochverfügbar, fehlertoleranter und skalierbarer als herkömmliche Infrastrukturen mit einem oder mehreren Rechenzentren.

Amazon-EFS-Dateisysteme sind widerstandsfähig gegen einen oder mehrere Availability-Zone-Ausfälle innerhalb einer AWS-Region. Die Mountingziele selbst sind hochverfügbar. Denken Sie beim Entwurf für Hochverfügbarkeit und Failover auf andere Systeme daran AZs, dass die IP-Adressen und DNS für Ihre Mount-Ziele in jeder AZ zwar statisch sind, es sich aber um redundante Komponenten handelt, die von mehreren Ressourcen unterstützt werden. Weitere Informationen finden Sie unter [So funktioniert Amazon EFS mit Amazon EC2](#).

Weitere Informationen zu Availability Zones AWS-Regionen und Availability Zones finden Sie unter [AWS Globale Infrastruktur](#).

Steuern des Netzwerkzugriffs auf EFS-Dateisysteme für NFS-Clients

Sie können den Zugriff von NFS-Clients auf Amazon-EFS-Dateisysteme mithilfe von Netzwerksicherheit und EFS-Dateisystemrichtlinien kontrollieren. Sie können die in Amazon EC2 verfügbaren Sicherheitsmechanismen auf Netzwerkebene verwenden, z. B. VPC-Sicherheitsgruppenregeln und Netzwerk. ACLs Sie können AWS IAM auch verwenden, um den NFS-Zugriff mit einer EFS-Dateisystemrichtlinie und identitätsbasierten Richtlinien zu steuern.

Themen

- [Verwenden von VPC-Sicherheitsgruppen](#)
- [Arbeiten mit VPC-Endpunkten mit Schnittstellen in Amazon EFS](#)

Verwenden von VPC-Sicherheitsgruppen

Wenn Sie Amazon EFS verwenden, geben Sie VPC-Sicherheitsgruppen für Ihre EC2-Instances und Sicherheitsgruppen für die EFS-Mount-Ziele an, die dem Dateisystem zugeordnet sind. Eine Sicherheitsgruppe fungiert als Firewall, und die Regeln, die Sie hinzufügen, definieren den Datenfluss. In der [Übung Erste Schritte](#) haben Sie beim Start der EFS-Instance eine Sicherheitsgruppe erstellt. Dann haben Sie dem EFS-Mounting-Ziel eine weitere Sicherheitsgruppe (die Standardsicherheitsgruppe für Ihre Standard-VPC) zugeordnet. Dieser Ansatz eignet sich für die Übung Erste Schritte. Für ein Produktionssystem sollten Sie jedoch Sicherheitsgruppen mit minimalen Berechtigungen für die Verwendung mit Amazon EFS einrichten.

Sie können eingehenden und ausgehenden Datenverkehr für Ihr EFS-Dateisystem autorisieren. Dazu fügen Sie Regeln hinzu, die es EFS-Instances ermöglichen, über das Mount-Ziel mithilfe des NFS-Ports (Network File System) eine Verbindung zu Ihrem EFS-Dateisystem herzustellen.

- Jede EC2-Instance, die das Dateisystem mountet, muss über eine Sicherheitsgruppe mit einer Regel verfügen, die ausgehenden Zugriff auf das Mount-Ziel auf dem NFS-Port 2049 ermöglicht.
- Das EFS-Mount-Ziel muss über eine Sicherheitsgruppe mit einer Regel verfügen, die eingehenden Zugriff auf den NFS-Port 2049 von jeder EC2-Instance aus ermöglicht, auf der Sie das Dateisystem mounten möchten.

In der folgenden Tabelle sind die spezifischen Sicherheitsgruppenregeln aufgeführt, die erforderlich sind:

Sicherheitsgruppe	Regeltyp	Protocol (Protokol l)	Port	Quelle/Ziel
EC2-Instance	Ausgehend	TCP	2049	Zielsicherheitsgruppe einhängen
Mount-Ziel	Eingehend	TCP	2049	EC2-Instance-Siche rheitsgruppe

Quellports für die Arbeit mit Amazon EFS

Um eine breite Palette von NFS-Klienten zu unterstützen, erlaubt Amazon EFS Verbindungen von jedem Quellport. Wenn Sie möchten, dass nur privilegierte Benutzer auf Amazon EFS zugreifen können, empfehlen wir die Verwendung der folgenden Client-Firewall-Regel. Verbinden Sie sich über SSH mit Ihrem Dateisystem und führen Sie den folgenden Befehl aus:

```
iptables -I OUTPUT 1 -m owner --uid-owner 1-4294967294 -m tcp -p tcp --dport 2049 -j DROP
```

Mit diesem Befehl wird eine neue Regel am Beginn der OUTPUT-Kette (-I OUTPUT 1) eingefügt. Die Regel verhindert, dass ein unberechtigter Prozess (-m owner --uid-owner 1-4294967294), der keine Kernel-Rechte besitzt, eine Verbindung zum NFS-Port 2049 () öffnet. -m tcp -p tcp -dport 2049

Sicherheitsüberlegungen für den Netzwerkzugriff

Ein NFS-Client der Version 4.1 (NFSv4.1) kann ein Dateisystem nur mounten, wenn er eine Netzwerkverbindung zum NFS-Port (TCP-Port 2049) eines der Mount-Ziele des Dateisystems herstellen kann. Ebenso kann ein NFSv4.1-Client beim Zugriff auf ein Dateisystem nur dann eine Benutzer- und Gruppen-ID angeben, wenn er diese Netzwerkverbindung herstellen kann.

Die Fähigkeit, diese Netzwerkverbindung herzustellen, wird durch eine Kombination der folgenden Faktoren festgelegt:

- Von der VPC des Mounting-Ziels bereitgestellte Netzwerkisolierung – Den Mounting-Zielen von Dateisystemen dürfen keine öffentlichen IP-Adressen zugewiesen sein. Die einzigen Ziele, die Dateisysteme mounten können, sind die folgenden:
 - Amazon EC2-Instances in der lokalen Amazon VPC
 - EC2-Instances sind verbunden VPCs
 - Lokale Server, die über ein AWS Virtual Private Network (VPN) mit AWS Direct Connect einer Amazon VPC verbunden sind
- Netzwerkzugriffskontrolllisten (ACLs) für die VPC-Subnetze des Clients und der Mount-Ziele, für den Zugriff von außerhalb der Subnetze des Mount-Ziels — Um ein Dateisystem zu mounten, muss der Client in der Lage sein, eine TCP-Verbindung zum NFS-Port 2049 eines Mount-Ziels herzustellen und Rückverkehr zu empfangen.
- Regeln der VPC-Sicherheitsgruppen des Clients und der Mountinghilfe, für alle Zugriffe – Damit eine EC2-Instance ein Dateisystem einhängen kann, müssen die folgenden Sicherheitsgruppenregeln gelten:
 - Das Dateisystem muss über ein Mount-Ziel verfügen, dessen Netzwerkschnittstelle über eine Sicherheitsgruppe mit einer Regel verfügt, die eingehende Verbindungen über den NFS-Port 2049 von der Instance aus ermöglicht. entweder nach IP-Adresse (CIDR-Bereich) oder nach Sicherheitsgruppe. Die Quelle der Sicherheitsgruppenregeln für eingehende Verbindungen auf dem NFS-Port für Netzwerkschnittstellen von Mounting-Zielen stellt ein wesentliches Element der Steuerung des Zugriffs auf Dateisysteme dar. Andere Regeln für eingehende Nachrichten als die für den NFS-Port 2049 und alle ausgehenden Regeln werden von Netzwerkschnittstellen nicht für Dateisystem-Mount-Ziele verwendet.
 - Die Mount-Instance muss über eine Netzwerkschnittstelle mit einer Sicherheitsgruppenregel verfügen, die ausgehende Verbindungen zum NFS-Port 2049 auf einem der Mount-Ziele des Dateisystems ermöglicht. Sie können ausgehende Verbindungen anhand von IP-Adressen (CIDR-Bereich) oder Sicherheitsgruppen aktivieren.

Weitere Informationen finden Sie unter [Verwalten der Mountingziele](#).

Erstellen von Sicherheitsgruppen

So erstellen Sie Sicherheitsgruppen für EC2-Instances und EFS-Mount-Ziele

Im Folgenden finden Sie die allgemeinen Schritte, die Sie beim Erstellen der Sicherheitsgruppen für Amazon EFS ausführen werden. Anweisungen zum Erstellen der Sicherheitsgruppen finden Sie unter [Erstellen einer Sicherheitsgruppe](#) im Amazon VPC-Benutzerhandbuch.

1. Erstellen Sie für Ihre EC2-Instances eine Sicherheitsgruppe mit den folgenden Regeln:
 - Eine Regel für eingehenden Datenverkehr, die eingehenden Zugriff über Secure Shell (SSH) auf Port 22 von Ihrer IP-Adresse oder Ihrem Netzwerk aus ermöglicht. Beschränken Sie optional die Quelladresse aus Sicherheitsgründen.
 - Eine ausgehende Regel, die ausgehenden Zugriff auf die Mount-Zielsicherheitsgruppe über den NFS-Port 2049 ermöglicht. Identifizieren Sie die Mount-Ziel-Sicherheitsgruppe als Ziel.
2. Erstellen Sie für Ihr EFS-Mount-Ziel eine Sicherheitsgruppe mit den folgenden Regeln:
 - Eine Regel für eingehenden Datenverkehr, die den Zugriff auf den NFS-Port 2049 von der EC2-Sicherheitsgruppe aus ermöglicht. Identifizieren Sie die EC2-Sicherheitsgruppe als Quelle.

Note

Sie müssen keine ausgehende Regel hinzufügen, da die Standardregel für ausgehenden Datenverkehr den gesamten ausgehenden Datenverkehr zulässt.

Arbeiten mit VPC-Endpunkten mit Schnittstellen in Amazon EFS

Um eine private Verbindung zwischen Ihrer Virtual Private Cloud (VPC) und der Amazon-EFS-API herzustellen, können Sie einen Schnittstellen-VPC-Endpunkt erstellen. Der Endpunkt bietet eine sichere Verbindung zur Amazon-EFS-API, ohne dass ein Internet-Gateway, eine NAT-Instance oder ein Virtual Private Network (VPN) erforderlich ist. Weitere Informationen finden Sie unter [Zugreifen auf einen AWS Service über einen Schnittstellen-VPC-Endpunkt](#) im Amazon VPC-Benutzerhandbuch.

Schnittstelle, mit der VPC-Endpunkte betrieben werden AWS PrivateLink, eine Funktion, die private Kommunikation zwischen AWS Diensten unter Verwendung privater IP-Adressen ermöglicht. Erstellen Sie zur Verwendung AWS PrivateLink einen VPC-Schnittstellen-Endpunkt für Amazon EFS in Ihrer VPC mithilfe der Amazon VPC-Konsole, API oder CLI. Dadurch wird eine elastische Netzwerkschnittstelle in Ihrem Subnetz mit einer privaten IP-Adresse erstellt, die Amazon EFS API-Anfragen bedient. Sie können auch von lokalen Umgebungen aus auf einen VPC-Endpunkt zugreifen oder Site-to-Site VPN Direct Connect VPC-Peering VPCs verwenden. Weitere Informationen finden Sie unter [Connect Sie Ihre VPC mit Diensten mithilfe von Diensten AWS PrivateLink](#) im Amazon VPC-Benutzerhandbuch.

Einen Schnittstellenendpunkt für Amazon EFS erstellen

Um einen Schnittstellen-VPC-Endpunkt für Amazon EFS zu erstellen, verwenden Sie eine der folgenden Möglichkeiten:

- **com.amazonaws.*region*.elasticfilesystem** – Erzeugt einen Endpunkt für Amazon-EFS-API-Vorgänge.
- **com.amazonaws.*region*.elasticfilesystem-fips** – Erzeugt einen Endpunkt für die Amazon-EFS-API, der dem [Federal Information Processing Standard \(FIPS\) 140-2](#) entspricht.

Eine vollständige Liste der Amazon EFS-Endpunkte finden Sie unter [Amazon Elastic File System-Endpunkte und Kontingente](#) in der. Allgemeine Amazon Web Services-Referenz

Weitere Informationen zum Erstellen eines Schnittstellenendpunkts finden Sie unter [Zugreifen auf einen AWS Service mithilfe eines Schnittstellen-VPC-Endpunkts](#) im Amazon VPC-Benutzerhandbuch.

Erstellen einer VPC-Endpunkttrichtlinie für Amazon EFS

Um den Zugriff auf die Amazon EFS-API zu kontrollieren, können Sie Ihrem VPC-Endpunkt eine AWS Identity and Access Management (IAM-) Richtlinie hinzufügen. Die Richtlinie legt Folgendes fest:

- Prinzipal, der die Aktionen ausführen kann.
- Aktionen, die ausgeführt werden können
- Die Ressourcen, für die Aktionen ausgeführt werden können.

Weitere Informationen finden Sie unter [Steuern des Zugriffs auf VPC-Endpunkte mithilfe von Endpunkttrichtlinien](#) im Amazon-VPC-Benutzerhandbuch.

Das folgende Beispiel zeigt eine VPC-Endpunktrichtlinie, die jedem die Berechtigung zum Erstellen eines EFS-Dateisystems über den Endpunkt verweigert. Die Beispielrichtlinie gewährt auch jedem die Berechtigung, alle anderen Aktionen auszuführen.

```
{
  "Statement": [
    {
      "Action": "*",
      "Effect": "Allow",
      "Resource": "*",
      "Principal": "*"
    },
    {
      "Action": "elasticfilesystem:CreateFileSystem",
      "Effect": "Deny",
      "Resource": "*",
      "Principal": "*"
    }
  ]
}
```

Benutzer, Gruppen und Berechtigungen auf NFS-Ebene (Network File System)

Nach dem Erstellen eines Dateisystems verfügt standardmäßig nur der Root-Benutzer (UID 0) über Lese-, Schreib- und Ausführungsberechtigungen. Damit auch andere Benutzer das Dateisystem ändern können, muss Ihnen der Root-Benutzer ausdrücklich Zugriff gewähren. Sie können mithilfe von Zugangspunkten die Erstellung von Verzeichnissen automatisieren, von denen ein Nicht-Root-Benutzer schreiben kann. Weitere Informationen finden Sie unter [Arbeiten mit Zugriffspunkten](#).

EFS-Dateisystemobjekten ist ein Modus im UNIX-Stil zugeordnet. Dieser Moduswert definiert die Berechtigungen zum Ausführen von Aktionen für dieses Objekt. Benutzer, die mit Unix-Systemen vertraut sind, können leicht verstehen, wie sich diese Berechtigungen verhalten.

Darüber hinaus werden Benutzer und Gruppen auf Unix-Systemen numerischen Bezeichnern zugeordnet, die Amazon EFS zur Darstellung von Dateibesitz verwendet. Bei Amazon EFS werden Dateisystemobjekte (d.h. Dateien, Verzeichnisse usw.) von einem einzigen Eigentümer und einer einzigen Gruppe verwaltet. Amazon EFS verwendet die zugeordnete Zahl, IDs um Berechtigungen zu überprüfen, wenn ein Benutzer versucht, auf ein Dateisystemobjekt zuzugreifen.

 Note

Das NFS-Protokoll unterstützt maximal 16 Gruppen IDs (GIDs) pro Benutzer, und alle weiteren Gruppen GIDs werden bei NFS-Client-Anfragen gekürzt. Weitere Informationen finden Sie unter [Zugriff auf zulässige Dateien im NFS-Dateisystem verweigert](#).

Nachfolgend finden Sie Beispiele für Berechtigungen und eine Diskussion über Überlegungen zu NFS-Berechtigungen für Amazon EFS.

Themen

- [Datei- und Verzeichnisberechtigungen](#)
- [Beispiele für Anwendungsfälle und Berechtigungen für das EFS-Dateisystem](#)
- [Benutzer- und Gruppen-ID-Berechtigungen für Dateien und Verzeichnisse innerhalb eines Dateisystems](#)
- [Kein Root-Squashing](#)
- [Zwischenspeichern von Berechtigungen](#)
- [Ändern des Besitzes an Dateisystemobjekten](#)
- [EFS-Zugangspunkte](#)

Datei- und Verzeichnisberechtigungen

Dateien und Verzeichnisse in einem EFS-Dateisystem unterstützen standardmäßige Lese-, Schreib- und Ausführungsberechtigungen im UNIX-Stil, die auf der Benutzer- und Gruppen-ID basieren, die vom NFSv4 Mount-.1-Client zugewiesen wurden, sofern sie nicht durch einen EFS-Zugriffspunkt überschrieben werden. Weitere Informationen finden Sie unter [Benutzer, Gruppen und Berechtigungen auf NFS-Ebene \(Network File System\)](#).

 Note

Standardmäßig hängt diese Ebene der Zugriffskontrolle davon ab, dass dem NFSv4 .1-Client bei der Bestätigung der Benutzer- und Gruppen-ID vertraut wird. Sie können ressourcenbasierte AWS Identity and Access Management (IAM-) Richtlinien und Identitätsrichtlinien verwenden, um NFS-Clients zu autorisieren und nur Lese-, Schreib- und Root-Zugriffsberechtigungen zu gewähren. Sie können EFS-Zugangspunkte verwenden,

um die vom NFS-Client bereitgestellten Benutzer- und Gruppenidentitätsinformationen des Betriebssystems zu übergehen. Weitere Informationen erhalten Sie unter [Verwendung von IAM zur Steuerung des Zugriffs auf Dateisysteme](#) und [Erstellen von Zugriffspunkten](#).

In diesem Beispiel der Lese-, Schreib- und Ausführungsberechtigungen für Dateien und Verzeichnisse hat Alice die Berechtigung zum Lesen und Schreiben beliebiger Dateien in ihrem privaten Verzeichnis auf einem Dateisystem, `/alice`. Alice besitzt in diesem Beispiel jedoch keine Lese- oder Schreibberechtigungen für Dateien im persönlichen Dateisystem von Mark im gleichen Dateisystem: `/mark`. Sowohl Alice als auch Mark dürfen Dateien im freigegebenen Verzeichnis `/share` lesen, jedoch nicht schreiben.

Beispiele für Anwendungsfälle und Berechtigungen für das EFS-Dateisystem

Nachdem Sie ein EFS-Dateisystem erstellt und Ziele für das Dateisystem in Ihrer VPC bereitgestellt haben, können Sie das Remote-Dateisystem lokal auf Ihrer Amazon EC2 EC2-Instance mounten. Mit dem Befehl `mount` kann jedes Verzeichnis im Dateisystem gemountet werden. Beim ersten Erstellen des Dateisystems ist jedoch nur ein Stammverzeichnis unter `/` vorhanden. Der Root-Benutzer und die Root-Gruppe sind Besitzer des gemounteten Verzeichnisses.

Der folgende `mount`-Befehl hängt das Stammverzeichnis eines Amazon-EFS-Dateisystems, das durch den DNS-Namen des Dateisystems identifiziert wird, in das `/efs-mount-point` lokale Verzeichnis ein.

```
sudo mount -t nfs -o
nfsvers=4.1,rsize=1048576,wsiz=1048576,hard,timeo=600,retrans=2,noresvport file-
system-id.efs.aws-region.amazonaws.com:/ efs-mount-point
```

Im ursprünglichen Berechtigungsmodus gelten folgende Berechtigungen:

- Berechtigungen `read-write-execute` für den Root-Besitzer
- Berechtigungen `read-execute` für die Root-Gruppe
- Berechtigungen `read-execute` für andere Benutzer

Dieses Verzeichnis kann nur vom Root-Benutzer geändert werden. Der Root-Benutzer kann auch anderen Benutzern Schreibberechtigung für dieses Verzeichnis erteilen, beispielsweise:

- Erstellen beschreibbarer Unterverzeichnisse für die einzelnen Benutzer. step-by-stepAnweisungen finden Sie unter. [Tutorial: Schreibbare Unterverzeichnisse pro Benutzer erstellen](#)
- Erlaubt Benutzern, in den EFS-Dateisystemstamm zu schreiben. Ein Benutzer mit Root-Berechtigung kann anderen Benutzern Zugriff auf das Dateisystem erteilen.
- Verwenden Sie Folgendes, um den Eigentümer des EFS-Dateisystems auf einen Benutzer und eine Gruppe zu ändern, die kein Root-Benutzer sind:

```
$ sudo chown user:group /EFSroot
```

- Verwenden Sie den folgenden Befehl, um die Dateisystemberechtigungen auszuweiten:

```
$ sudo chmod 777 /EFSroot
```

Dieser Befehl gewährt allen Benutzern auf allen EC2-Instances, auf denen das Dateisystem gemountet ist, read-write-execute Rechte.

Benutzer- und Gruppen-ID-Berechtigungen für Dateien und Verzeichnisse innerhalb eines Dateisystems

Dateien und Verzeichnisse in einem EFS-Dateisystem unterstützen standardmäßige Lese-, Schreib- und Ausführungsberechtigungen im UNIX-Stil, die auf der Benutzer-ID und Gruppe basieren. IDs Wenn ein NFS-Client ein EFS-Dateisystem ohne Verwendung eines Zugangspunkts mountet, sind die vom Client bereitgestellte Benutzer-ID und die Gruppen-ID vertrauenswürdig. Sie können EFS-Zugriffspunkte verwenden, um die vom NFS-Client IDs verwendete Benutzer-ID und Gruppe zu überschreiben. Wenn Benutzer versuchen, auf Dateien und Verzeichnisse zuzugreifen, überprüft Amazon EFS ihren Benutzer IDs und ihre Gruppe, IDs um sicherzustellen, dass jeder Benutzer berechtigt ist, auf die Objekte zuzugreifen. Amazon EFS verwendet diese auch IDs , um den Eigentümer und Gruppenbesitzer für neue Dateien und Verzeichnisse anzugeben, die der Benutzer erstellt. Amazon EFS untersucht keine Benutzer- oder Gruppennamen, sondern verwendet nur die numerischen Bezeichner.

Note

Wenn Sie einen Benutzer in einer EC2-Instance erstellen, können Sie ihm eine beliebige numerische User-ID (UID) und Gruppen-ID (GID) zuweisen. Die numerischen Benutzer IDs sind in der `/etc/passwd` Datei auf Linux-Systemen festgelegt. Die numerische Gruppe IDs befindet sich in der `/etc/group` Datei. Diese Dateien definieren die Zuweisungen zwischen

Namen und IDs. Außerhalb der EC2-Instance führt Amazon EFS keine Authentifizierung dieser durch IDs, einschließlich der Root-ID 0.

Wenn ein Benutzer von zwei verschiedenen EC2-Instances aus auf ein EFS-Dateisystem zugreift, sehen Sie je nachdem, ob die UID für den Benutzer auf diesen Instances identisch oder unterschiedlich ist, ein unterschiedliches Verhalten wie folgt:

- Wenn die Benutzer-IDs auf beiden EC2-Instances gleich sind, geht Amazon EFS davon aus, dass es sich um denselben Benutzer handelt, unabhängig davon, welche EC2-Instances verwendet wird. Die Benutzererfahrung beim Zugriff auf das Dateisystem ist dann von beiden EC2-Instances aus gleich.
- Wenn die Benutzer-IDs auf beiden EC2-Instances nicht identisch sind, betrachtet Amazon EFS die Benutzer als unterschiedliche Benutzer. Die Benutzererfahrung ist nicht dieselbe, wenn von den beiden verschiedenen EC2-Instances aus auf das EFS-Dateisystem zugegriffen wird.
- Wenn zwei verschiedene Benutzer auf verschiedenen EC2-Instances eine ID teilen, betrachtet Amazon EFS sie als denselben Benutzer.

Sie könnten darüber nachdenken, Benutzer-ID-Zuweisung für alle EC2-Instances konsistent zu verwalten. Benutzer können ihre numerische ID mit dem Befehl `id` überprüfen.

```
$ id

uid=502(joe) gid=502(joe) groups=502(joe)
```

Ausschalten des ID-Mappers

Zu den NFS-Dienstprogrammen im Betriebssystem gehört ein Daemon namens ID Mapper, der die Zuordnung zwischen Benutzernamen und verwaltet. IDs In Amazon Linux heißt der Daemon `rpc.idmapd` und in Ubuntu `idmapd`. Er setzt Benutzer- und Gruppen-IDs in Namen um und umgekehrt. Amazon EFS arbeitet jedoch nur mit numerischen IDs. Wir empfehlen, dass Sie diesen Prozess auf Ihren EC2-Instances ausschalten. Auf Amazon Linux ist der ID-Mapper in der Regel deaktiviert. Falls nicht, aktivieren Sie ihn nicht. Um den ID-Mapper zu deaktivieren, verwenden Sie die im Folgenden dargestellten Befehle.

```
$ service rpcidmapd status
$ sudo service rpcidmapd stop
```


Kein Root-Squashing

Standardmäßig ist Root-Squashing auf EFS-Dateisystemen deaktiviert. Amazon EFS verhält sich wie ein Linux NFS-Server mit `no_root_squash`. Wenn eine Benutzer- oder Gruppen-ID 0 ist, behandelt Amazon EFS diesen Benutzer als `root`-Benutzer und umgeht die Berechtigungsprüfungen (und erlaubt den Zugriff und die Änderung aller Dateisystemobjekte). Root-Squashing kann auf einer Client-Verbindung aktiviert werden, wenn die AWS Identity and Access Management (AWS IAM-) Identitäts- oder Ressourcenrichtlinie keinen Zugriff auf die Aktion zulässt. `ClientRootAccess` Wenn Root-Squashing aktiviert ist, wird der Root-Benutzer auf dem NFS-Server in einen Benutzer mit beschränkten Berechtigungen konvertiert.

Weitere Informationen finden Sie unter [Verwendung von IAM zur Steuerung des Zugriffs auf Dateisysteme](#).

Aktivieren Sie Root-Squashing mithilfe der IAM-Autorisierung für NFS-Clients

Sie können Amazon EFS so konfigurieren, dass der Root-Zugriff auf Ihr EFS-Dateisystem für alle AWS Prinzipale außer für eine einzelne Management-Workstation verhindert wird. Dazu konfigurieren Sie die AWS Identity and Access Management -(IAM-)Autorisierung für Network File System (NFS)-Clients.

Dazu müssen Sie zwei IAM-Berechtigungsrichtlinien konfigurieren, und zwar wie folgt:

- Erstellen Sie eine EFS-Dateisystemrichtlinie, die explizit Lese- und Schreibzugriff auf das Dateisystem gewährt und den Root-Zugriff implizit verweigert.
- Weisen Sie der Amazon EC2-Management-Workstation, die Root-Zugriff auf das Dateisystem benötigt, mithilfe eines EC2-Instance-Profils eine IAM-Identität zu. Weitere Informationen zu Amazon EC2 EC2-Instance-Profilen finden Sie unter [Verwenden von Instance-Profilen](#) im AWS Identity and Access Management Benutzerhandbuch.
- Weisen Sie die `AmazonElasticFileSystemClientFullAccess` AWS verwaltete Richtlinie der IAM-Rolle der Management-Workstation zu. Weitere Informationen zu AWS verwalteten Richtlinien für Amazon EFS finden Sie unter [Identitäts- und Zugriffsmanagement für Amazon EFS](#).

Verwenden Sie die folgenden Verfahren, um Root-Squashing mit IAM-Autorisierung für NFS-Clients zu aktivieren.

So verhindern Sie den Root-Zugriff auf das Dateisystem

1. Öffnen Sie die Amazon Elastic File System-Konsole unter <https://console.aws.amazon.com/efs/>.

2. Wählen Sie Dateisysteme aus.
3. Wählen Sie das Dateisystem aus, auf dem Sie Root-Squashing aktivieren möchten.
4. Wählen Sie auf der Seite Dateisystemdetails die Option Dateisystemrichtlinie und dann Bearbeiten. Die Seite File system policy (Dateisystemrichtlinie) wird angezeigt.
5. Wählen Sie Root-Zugriff standardmäßig verhindern* unter Richtlinienoptionen. Das JSON-Objekt für die Richtlinie wird im Richtlinien-Editor angezeigt.
6. Wählen Sie Speichern, um die Dateisystemrichtlinie zu speichern.

Clients, die nicht anonym sind, können über eine identitätsbasierte Richtlinie Root-Zugriff auf das Dateisystem erhalten. Wenn Sie die `AmazonElasticFileSystemClientFullAccess`-verwaltete Richtlinie mit der Rolle der Arbeitsstation verknüpfen, gewährt IAM der Arbeitsstation Root-Zugriff auf der Grundlage ihrer Identitätsrichtlinie.

So aktivieren Sie den Root-Zugriff von der Management-Workstation aus:

1. Öffnen Sie unter <https://console.aws.amazon.com/iam/> die IAM-Konsole.
2. Erstellen Sie eine Rolle für Amazon EC2 mit dem Namen `EFS-client-root-access`. IAM erstellt ein Instance-Profil mit demselben Namen wie die von Ihnen erstellte EC2-Rolle.
3. Weisen Sie die AWS verwaltete Richtlinie der `AmazonElasticFileSystemClientFullAccess` von Ihnen erstellten EC2-Rolle zu. Der Inhalt dieser Richtlinie wird im Folgenden dargestellt.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticfilesystem:ClientMount",
        "elasticfilesystem:ClientRootAccess",
        "elasticfilesystem:ClientWrite",
        "elasticfilesystem:DescribeMountTargets"
      ],
      "Resource": "*"
    }
  ]
}
```

}

4. Fügen Sie das Instance-Profil an die EC2-Instance an, die Sie als Management-Workstation verwenden, wie im Folgenden beschrieben. Weitere Informationen finden Sie unter [Anhängen einer IAM-Rolle an eine Instance](#) im Amazon EC2-Benutzerhandbuch für Linux-Instances.
 - a. Öffnen Sie die Amazon-EC2-Konsole unter <https://console.aws.amazon.com/ec2/>.
 - b. Wählen Sie im Navigationsbereich Instances aus.
 - c. Wählen Sie die Instance aus. Wählen Sie unter Aktionen die Option Instance-Einstellungen und dann IAM-Rolle anfügen/ersetzen aus.
 - d. Wählen Sie die IAM-Rolle aus, die Sie im ersten Schritt, `EFS-client-root-access`, erstellt haben und wählen Sie Anwenden.
5. Installieren Sie die EFS-Mountinghilfe auf der Management-Workstation. Weitere Hinweise zum EFS-Mount-Helper-Mount-Helper und dem `amazon-efs-utils` Paket finden Sie unter [Den Amazon EFS-Client installieren](#).
6. Mounten Sie das EFS-Dateisystems auf der Management-Workstation mithilfe des folgenden Befehls mit der Mountingoption `iam`.

```
$ sudo mount -t efs -o tls,iam file-system-id:/ efs-mount-point
```

Sie können die Amazon EC2-Instance so konfigurieren, dass das Dateisystem automatisch mit IAM-Autorisierung eingehängt wird. Weitere Informationen zum Mounten eines EFS-Dateisystems mit IAM-Autorisierung finden Sie unter [Mounting mit IAM-Autorisierung](#).

Zwischenspeichern von Berechtigungen

Amazon EFS speichert die Dateiberechtigungen für einen kurzen Zeitraum. Infolgedessen kann es ein kurzes Zeitfenster geben, in dem ein Benutzer, dem vor kurzem der Zugriff entzogen wurde, immer noch auf dieses Objekt zugreifen kann.

Ändern des Besitzes an Dateisystemobjekten

Amazon EFS setzt das POSIX `chown_restricted`-Attribut durch. Somit kann nur der Root-Benutzer den Besitzer eines Dateisystemobjekts ändern. Der Root-Benutzer oder der Eigentümerbenutzer können die Besitzergruppe eines Dateisystemobjekts ändern. Sofern es sich jedoch nicht um den Root-Benutzer handelt, kann die Gruppe nur in eine Gruppe geändert werden, welcher der Eigentümerbenutzer angehört.

EFS-Zugangspunkte

Ein Zugangspunkt wendet Betriebssystembenutzer, -gruppe und -dateisystempfad auf alle Dateisystemanforderungen an, die mit dem Zugangspunkt durchgeführt werden. Der Betriebssystembenutzer und die Gruppe des Zugriffspunkts überschreiben alle vom NFS-Client bereitgestellten Identitätsinformationen. Der Dateisystempfad wird dem Client als Stammverzeichnis des Zugriffspunkts angezeigt. Durch diesen Ansatz wird sichergestellt, dass jede Anwendung beim Zugriff auf freigegebene dateibasierte Datasets immer die richtige Betriebssystemidentität und das richtige Verzeichnis verwendet. Anwendungen, die den Zugriffspunkt verwenden, können nur auf Daten in einem eigenen Verzeichnis und darunter zugreifen. Weitere Hinweise zu Zugangspunkten finden Sie unter [Arbeiten mit Zugriffspunkten](#).

Arbeiten mit Zugriffspunkten

EFS-Zugriffspunkte sind anwendungsspezifische Einstiegspunkte in ein EFS-Dateisystem, die die Verwaltung des Anwendungszugriffs auf gemeinsam genutzte Datensätze erleichtern. Zugangspunkte können eine Benutzeridentität, einschließlich der POSIX-Gruppen des Benutzers, für alle Dateisystemanforderungen erzwingen, die über den Zugangspunkt erfolgen. Zugriffspunkte können auch ein anderes Stammverzeichnis für das Dateisystem erzwingen, so dass Clients nur auf Daten im angegebenen Verzeichnis oder in seinen Unterverzeichnissen zugreifen können.

Sie können AWS Identity and Access Management (IAM) -Richtlinien verwenden, um zu erzwingen, dass bestimmte Anwendungen einen bestimmten Zugriffspunkt verwenden. Durch die Kombination von IAM-Richtlinien mit Zugriffspunkten können Sie ganz einfach einen sicheren Zugriff auf bestimmte Datasets für Ihre Anwendungen bereitstellen.

Sie können Zugriffspunkte für ein vorhandenes EFS-Dateisystem mithilfe der AWS-Managementkonsole, der AWS Command Line Interface (AWS CLI) und der Amazon EFS-API erstellen. step-by-stepVerfahren zum Erstellen eines Access Points finden Sie unter [Erstellen von Zugriffspunkten](#).

Access Points funktionieren mit Mount-Zielen

Sie müssen mindestens ein Mount-Ziel in Ihrer VPC erstellen, bevor Sie Access Points verwenden können. Mount-Ziele stellen die Netzwerkkonnektivität zu Ihrem EFS-Dateisystem bereit, während Access Points die Zugriffskontrolle und anwendungsspezifische Einstiegspunkte bereitstellen.

Access Points übernehmen die Platzierung des Mount-Ziels in der Availability Zone.

- Sicherheitsgruppen werden auf der Mount-Zielebene angewendet, nicht auf der Ebene des Access Points.
- Access Points sind in allen Availability Zones verfügbar, in denen Sie Mount-Ziele haben.
- Der IAM-Bedingungsschlüssel `elasticfilesystem:AccessedViaMountTarget` stellt sicher, dass der Dateisystemzugriff nur über Mount-Ziele erfolgt. Dies gilt sowohl für direkte Mounts als auch für Access Point-Mounts.

Sie verwenden den EFS-Mount-Helfer, wenn Sie ein Dateisystem mit einem Zugriffspunkt mounten. Im Mounting-Befehl müssen Sie die Dateisystem-ID, die Zugriffspunkt-ID und die im folgenden Beispiel gezeigte Mountingoption `tls` einschließen.

```
$ mount -t efs -o tls,iam,accesspoint=fsap-abcdef0123456789a fs-  
abc0123def456789a: /localmountpoint
```

Weitere Hinweise zum Mounting von Dateisystemen mithilfe eines Zugriffspunkts finden Sie unter [Mounting mit EFS-Zugangspunkten](#).

Themen

- [Erzwingen einer Benutzeridentität mithilfe eines Zugangspunkts](#)
- [Erzwingen eines Stammverzeichnisses mit einem Zugangspunkt](#)
- [Verwenden von Zugangspunkten in IAM-Richtlinien](#)

Erzwingen einer Benutzeridentität mithilfe eines Zugangspunkts

Sie können einen Zugriffspunkt verwenden, um Benutzer- und Gruppeninformationen für alle Dateisystemanforderungen durchzusetzen, die über den Zugriffspunkt erfolgen. Um diese Funktion zu aktivieren, müssen Sie die Betriebssystemidentität angeben, die beim Erstellen des Zugriffspunkts erzwungen werden soll.

Als Teil davon geben Sie Folgendes an:

- Benutzer-ID – Die numerische POSIX-Benutzer-ID für den Benutzer.
- Gruppen-ID – Die numerische POSIX-Gruppen-ID für den Benutzer.
- Sekundäre Gruppe IDs — Eine optionale Liste sekundärer Gruppen IDs.

Wenn die Benutzerdurchsetzung aktiviert ist, ersetzt Amazon EFS den Benutzer und die Gruppe des NFS-Clients IDs durch die Identität, die auf dem Access Point für alle Dateisystemoperationen konfiguriert ist. Die Benutzererzwingung zeigt zudem folgende Wirkung:

- Der Besitzer und die Gruppe für neue Dateien und Verzeichnisse werden auf die Benutzer-ID und die Gruppen-ID des Zugriffspunkts festgelegt.
- EFS berücksichtigt bei der Bewertung der Dateisystemberechtigungen die Benutzer-ID, Gruppen-ID und sekundäre Gruppe IDs des Access Points. EFS ignoriert die des NFS-Clients. IDs

Important

Das Erzwingen einer Benutzeridentität unterliegt der `ClientRootAccess-IAM`-Berechtigung.

In einigen Fällen können Sie beispielsweise die Benutzer-ID oder die Gruppen-ID des Zugriffspunkts oder beide als Root konfigurieren (d. h. die UID, die GID oder beide auf 0 setzen). In solchen Fällen müssen Sie dem NFS-Client die `ClientRootAccess-IAM`-Berechtigung erteilen.

Erzwingen eines Stammverzeichnisses mit einem Zugangspunkt

Sie können einen Zugriffspunkt verwenden, um das Stammverzeichnis für ein Dateisystem außer Kraft zu setzen. Wenn Sie ein Stammverzeichnis erzwingen, verwendet der NFS-Client, der den Zugriffspunkt verwendet, das auf dem Zugriffspunkt konfigurierte Stammverzeichnis anstelle des Stammverzeichnisses des Dateisystems.

Sie aktivieren diese Funktion, indem Sie beim Erstellen eines Zugriffspunkts das Attribut `Path` festlegen. Das `Path`-Attribut ist der vollständige Pfad des Stammverzeichnisses des Dateisystems für alle Dateisystemanforderungen, die über diesen Zugriffspunkt erfolgen. Der vollständige Pfad darf nicht mehr als 100 Zeichen lang sein. Es kann bis zu vier Unterverzeichnisse enthalten.

Wenn Sie ein Stammverzeichnis auf einem Zugriffspunkt angeben, wird es zum Stammverzeichnis des Dateisystems für den NFS-Client, der den Zugriffspunkt mountet. Angenommen, das Stammverzeichnis Ihres Zugriffspunkts ist `/data`. In diesem Fall zeigt das Mounten von `fs-12345678:/` mittels des Zugriffspunkts die gleiche Wirkung wie das Mounten von `fs-12345678:/data`, ohne den Zugriffspunkt zu verwenden.

Stellen Sie bei der Angabe eines Stammverzeichnisses im Zugriffspunkt sicher, dass die Verzeichnisberechtigungen so konfiguriert sind, dass der Benutzer des Zugriffspunkts das Dateisystem erfolgreich mounten kann. Stellen Sie insbesondere sicher, dass das Ausführungs-Bit für den Benutzer bzw. die Gruppe des Zugriffspunkts oder für alle festgelegt ist. Mit einem Verzeichnisberechtigenswert von 755 kann der Verzeichnisbenutzer beispielsweise Dateien auflisten, Dateien erstellen und mounten, und alle anderen Benutzer können Dateien auflisten und mounten.

Erstellen des Stammverzeichnisses für einen Zugangspunkt

Wenn auf dem Dateisystem kein Stammverzeichnispfad für einen Zugangspunkt vorhanden ist, erstellt Amazon EFS automatisch dieses Stammverzeichnis mit Besitzrechten und angegebenen Berechtigungen. Amazon EFS erstellt das Stammverzeichnis nicht, wenn Sie bei der Erstellung nicht den Verzeichnisbesitz und die Berechtigungen angeben. Dieser Ansatz ermöglicht es, einen Dateisystemzugriff für einen bestimmten Benutzer oder eine bestimmte Anwendung bereitzustellen, ohne Ihr Dateisystem von einem Linux-Host zu mounten. Zum Erstellen eines Stammverzeichnisses müssen Sie den Besitz und die Berechtigung des Stammverzeichnisses mithilfe der folgenden Attribute konfigurieren, wenn Sie einen Zugangspunkt erstellen:

- `OwnerId` – Die numerische POSIX-Benutzer-ID, die als Besitzer des Stammverzeichnisses verwendet werden soll.
- `OwnerGid` – Die numerische POSIX-Gruppen-ID, die als Besitzergruppe des Stammverzeichnisses verwendet werden soll.
- `Berechtigungen` – Der Unix-Modus des Verzeichnisses. Eine allgemeine Konfiguration ist 755. Stellen Sie sicher, dass das Ausführungs-Bit für den Benutzer des Zugriffspunkts festgelegt ist, damit er mounten kann. Diese Konfiguration erteilt dem Verzeichnisbesitzer die Berechtigung, neue Dateien in das Verzeichnis einzugeben und zu schreiben und in ihm aufzulisten. Sie erteilt allen anderen Benutzern die Berechtigung, Dateien einzugeben und aufzulisten. Weitere Informationen zum Arbeiten mit Unix-Datei- und Verzeichnismodi finden Sie unter [Benutzer, Gruppen und Berechtigungen auf NFS-Ebene \(Network File System\)](#).

Amazon EFS erstellt nur dann ein Access Point-Stammverzeichnis, wenn die `OwnerId`, `ownGID` und die Berechtigungen für das Verzeichnis angegeben sind. Wenn Sie diese Informationen nicht angeben, erstellt Amazon EFS das Stammverzeichnis nicht. Wenn das Stammverzeichnis nicht existiert, schlagen Mount-Versuche beim Zugangspunkt fehl.

Wenn Sie ein Dateisystem mit einem Access Point mounten, wird das Stammverzeichnis für den Access Point erstellt, sofern das Verzeichnis noch nicht existiert, vorausgesetzt, dass das Stammverzeichnis OwnerUid und die Berechtigungen bei der Erstellung des Access Points angegeben wurden. Wenn das auf dem Zugangspunkt konfigurierte Stammverzeichnis bereits vor dem Mounten vorhanden ist, werden die vorhandenen Berechtigungen vom Zugangspunkt nicht überschrieben. Wenn Sie das Stammverzeichnis löschen, erstellt EFS es neu, wenn das Dateisystem das nächste Mal über den Zugriffspunkt gemountet wird.

Note

Amazon EFS erstellt das Stammverzeichnis nicht, wenn Sie für das Zugangspunkt-Stammverzeichnis nicht den Besitz und die Berechtigungen angeben. Alle Versuche, den Zugangspunkt zu mounten, schlagen fehl.

Sicherheitsmodell für Zugangspunkt-Stammverzeichnisse

Wenn Stammverzeichnis überschrieben wird, verhält sich Amazon EFS wie ein Linux NFS-Server mit aktivierter `no_subtree_check`-Option.

Im NFS-Protokoll generieren Server Dateihandles, die von Clients als eindeutige Referenzen beim Zugriff auf Dateien verwendet werden. EFS generiert in sicherer Weise Dateihandles, die unvorhersehbar und spezifisch für ein EFS-Dateisystem sind. Wenn eine Stammverzeichnisüberschreibung aktiviert ist, gibt Amazon EFS keine Dateihandles für Dateien außerhalb des angegebenen Stammverzeichnisses bekannt. In einigen Fällen kann ein Benutzer jedoch mithilfe eines out-of-band Mechanismus ein Datei-Handle für eine Datei außerhalb seines Zugriffspunkts abrufen. Sie verfahren beispielsweise so, wenn sie Zugriff auf einen zweiten Zugriffspunkt haben. Wenn sie dies tun, können sie Lese- und Schreiboperationen für die Datei ausführen.

Dateibesitz und Zugriffsberechtigungen werden immer erzwungen, für den Zugriff auf Dateien innerhalb und außerhalb des Zugriffspunkt-Stammverzeichnisses eines Benutzers.

Verwenden von Zugangspunkten in IAM-Richtlinien

Mit einer IAM-Richtlinie können Sie erzwingen, dass ein bestimmter NFS-Client, der durch seine IAM-Rolle identifiziert wird, nur auf einen bestimmten Zugriffspunkt zugreifen kann. Dazu verwenden Sie den `elasticfilesystem:AccessPointArn`-IAM-Bedingungsschlüssel. Der `AccessPointArn` ist der Amazon-Ressourcenname (ARN) des Zugriffspunkts, mit dem das Dateisystem gemountet ist.

Es folgt ein Beispiel für eine Dateisystemrichtlinie, die es der IAM-Rolle app1 ermöglicht, über den Zugriffspunkt fsap-01234567 auf das Dateisystem zuzugreifen. Die Richtlinie ermöglicht app2 auch die Verwendung des Dateisystems über den Zugriffspunkt fsap-89abcdef.

JSON

```
{
  "Version": "2012-10-17",
  "Id": "MyFileSystemPolicy",
  "Statement": [
    {
      "Sid": "App1Access",
      "Effect": "Allow",
      "Principal": { "AWS": "arn:aws:iam::111122223333:role/app1" },
      "Action": [
        "elasticfilesystem:ClientMount",
        "elasticfilesystem:ClientWrite"
      ],
      "Resource": "arn:aws:elasticfilesystem:us-east-1:111122223333:file-
system/*",
      "Condition": {
        "StringEquals": {
          "elasticfilesystem:AccessPointArn" :
            "arn:aws:elasticfilesystem:us-east-1:222233334444:access-point/fsap-01234567"
        }
      }
    },
    {
      "Sid": "App2Access",
      "Effect": "Allow",
      "Principal": { "AWS": "arn:aws:iam::111122223333:role/app2" },
      "Action": [
        "elasticfilesystem:ClientMount",
        "elasticfilesystem:ClientWrite"
      ],
      "Resource": "arn:aws:elasticfilesystem:us-east-1:111122223333:file-
system/*",
      "Condition": {
        "StringEquals": {
          "elasticfilesystem:AccessPointArn" :
            "arn:aws:elasticfilesystem:us-east-1:222233334444:access-point/fsap-89abcdef"
        }
      }
    }
  ]
}
```

```
}  
  }  
    ]  
  }
```

Sperren des öffentlichen Zugriffs auf EFS-Dateisysteme

Die Amazon EFS-Funktion zum Blockieren des öffentlichen Zugriffs bietet Einstellungen, mit denen Sie den öffentlichen Zugriff auf EFS-Dateisysteme verwalten können. Standardmäßig erlauben neue EFS-Dateisysteme keinen öffentlichen Zugriff. Sie können jedoch die Dateisystemrichtlinien ändern, um den öffentlichen Zugriff zuzulassen.

Important

Die Aktivierung von Block Public Access trägt zum Schutz Ihrer Ressourcen bei, indem verhindert wird, dass öffentlicher Zugriff über die direkt mit dem Dateisystem verknüpften Ressourcenrichtlinien gewährt wird. Überprüfen Sie zusätzlich zur Aktivierung von Block Public Access sorgfältig die folgenden Richtlinien, um sicherzustellen, dass sie keinen öffentlichen Zugriff gewähren:

- Identitätsbasierte Richtlinien, die mit zugehörigen AWS Prinzipalen verknüpft sind (z. B. IAM-Rollen)
- Ressourcenbasierte Richtlinien, die mit zugehörigen AWS Ressourcen verknüpft sind (z. B. (KMS-) Schlüssel)AWS Key Management Service

Topics

- [Sperren des öffentlichen Zugriffs mit AWS Transfer Family](#)
- [Die Bedeutung von „öffentlich“](#)

Sperren des öffentlichen Zugriffs mit AWS Transfer Family

Wenn Sie Amazon EFS mit verwenden AWS Transfer Family, werden Dateisystemzugriffsanforderungen blockiert, die von einem Transfer Family Family-Server empfangen werden, der einem anderen Konto als dem Dateisystem gehört, wenn das Dateisystem öffentlichen Zugriff zulässt. Amazon EFS bewertet die IAM-Richtlinien des Dateisystems und wenn die Richtlinie

öffentlich ist, blockiert es die Anfrage. Um den AWS Transfer Family Zugriff auf Ihr Dateisystem zu ermöglichen, aktualisieren Sie Ihre Dateisystemrichtlinie, sodass sie nicht als öffentlich betrachtet wird.

Note

Die Verwendung von Transfer Family mit Amazon AWS-Konto EFS ist standardmäßig für Systeme deaktiviert, die über EFS-Dateisysteme mit Richtlinien verfügen, die öffentlichen Zugriff ermöglichen und die vor dem 6. Januar 2021 erstellt wurden. Wenden Sie sich an den AWS Support, um die Verwendung von Transfer Family für den Zugriff auf Ihr Dateisystem zu aktivieren.

Die Bedeutung von „öffentlich“

Bei der Bewertung, ob ein Dateisystem öffentlichen Zugriff zulässt, geht Amazon EFS davon aus, dass die Dateisystemrichtlinie öffentlich ist. Dann evaluiert es die Richtlinie des Dateisystems, um festzustellen, ob sie als nichtöffentlich eingestuft werden kann. Um als nichtöffentlich zu gelten, darf eine Dateisystemrichtlinie nur Zugriff für feste Werte (Werte, die keine Platzhalter aufweisen) gewähren, wie einen oder mehrere der folgenden Werte:

- Ein AWS Principal, ein Benutzer, eine Rolle oder ein Dienstprinzipal (zum Beispiel `aws:PrincipalOrgID`)
- `aws:SourceArn`
- `aws:SourceVpc`
- `aws:SourceVpce`
- `aws:SourceOwner`
- `aws:SourceAccount`
- `elasticfilesystem:AccessedViaMountTarget`
- `aws:userid`, outside the pattern `"AROLEID:*`

Nach diesen Regeln gilt die folgende Beispielrichtlinie als öffentlich.

JSON

```
{
```

```

"Version": "2012-10-17",
"Id": "efs-policy-wizard-15ad9567-2546-4bbb-8168-5541b6fc0e55",
"Statement": [
  {
    "Sid": "efs-statement-14a7191c-9401-40e7-a388-6af6cfb7dd9c",
    "Effect": "Allow",
    "Principal": {
      "AWS": "*"
    },
    "Action": [
      "elasticfilesystem:ClientMount",
      "elasticfilesystem:ClientWrite",
      "elasticfilesystem:ClientRootAccess"
    ],
    "Resource": "arn:aws:elasticfilesystem:us-east-1:111122223333:file-
system/*"
  }
]
}

```

Sie können diese Dateisystemrichtlinie nichtöffentlich machen, indem Sie den EFS-Bedingungsschlüssel `elasticfilesystem:AccessedViaMountTarget` verwenden, der auf „true“ gesetzt ist. Sie können `elasticfilesystem:AccessedViaMountTarget` verwenden, um Clients, die über ein Dateisystem-Mount-Ziel auf das EFS-Dateisystem zugreifen, die angegebenen EFS-Aktionen zuzulassen. Die folgende nichtöffentliche Richtlinie verwendet den `elasticfilesystem:AccessedViaMountTarget`-Bedingungsschlüssel, der auf „true“ gesetzt ist.

JSON

```

{
  "Version": "2012-10-17",
  "Id": "efs-policy-wizard-15ad9567-2546-4bbb-8168-5541b6fc0e55",
  "Statement": [
    {
      "Sid": "efs-statement-14a7191c-9401-40e7-a388-6af6cfb7dd9c",
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },

```

```
    "Action": [
      "elasticfilesystem:ClientMount",
      "elasticfilesystem:ClientWrite",
      "elasticfilesystem:ClientRootAccess"
    ],
    "Resource": "arn:aws:elasticfilesystem:us-east-1:111122223333:file-
system/*",
    "Condition": {
      "Bool": {
        "elasticfilesystem:AccessedViaMountTarget": "true"
      }
    }
  }
]
```

Weitere Informationen über Amazon-EFS-Bedingungsschlüssel finden Sie unter [EFS-Bedingungsschlüssel für Clients](#). Weitere Informationen zum Erstellen von Dateisystemrichtlinien finden Sie unter [Erstellen von Dateisystemrichtlinien](#).

Netzwerkisolierung für Amazon EFS

Als verwalteter Service ist Amazon Elastic File System durch AWS globale Netzwerksicherheit geschützt. Informationen zu AWS Sicherheitsdiensten und zum AWS Schutz der Infrastruktur finden Sie unter [AWS Cloud-Sicherheit](#). Informationen zum Entwerfen Ihrer AWS Umgebung unter Verwendung der bewährten Methoden für die Infrastruktursicherheit finden Sie unter [Infrastructure Protection](#) in Security Pillar AWS Well-Architected Framework.

Sie verwenden AWS veröffentlichte API-Aufrufe, um über das Netzwerk auf Amazon EFS zuzugreifen. Kunden müssen Folgendes unterstützen:

- Transport Layer Security (TLS). Wir benötigen TLS 1.2 und empfehlen TLS 1.3.
- Verschlüsselungs-Suiten mit Perfect Forward Secrecy (PFS) wie DHE (Ephemeral Diffie-Hellman) oder ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). Die meisten modernen Systeme wie Java 7 und höher unterstützen diese Modi.

Diese können von jedem Netzwerkstandort aus aufgerufen APIs werden, Amazon EFS unterstützt jedoch ressourcenbasierte Zugriffsrichtlinien, die Einschränkungen auf der Grundlage der Quell-IP-Adresse beinhalten können. Sie können auch Amazon EFS-Richtlinien verwenden, um den Zugriff von bestimmten Amazon Virtual Private Cloud (Amazon VPC) -Endpunkten oder bestimmten zu kontrollieren. VPCs Dadurch wird der Netzwerkzugriff auf eine bestimmte Amazon EFS-Ressource effektiv nur von der spezifischen VPC innerhalb des AWS Netzwerks isoliert.

Amazon EFS-Kontingente

Im Folgenden werden die Kontingente bei der Arbeit mit Amazon EFS beschrieben.

Themen

- [Amazon-EFS-Kontingente, die Sie erhöhen können](#)
- [Amazon-EFS-Ressourcenkontingente, die Sie nicht ändern können](#)
- [Kontingente für NFS-Clients](#)
- [Kontingente für Amazon-EFS-Dateisysteme](#)
- [Nicht unterstützte NFSv4 2.0- und 4.1-Funktionen](#)
- [Weitere Überlegungen](#)
- [Behebung von Fehlern bei Dateivorgängen im Zusammenhang mit Kontingenten](#)

Amazon-EFS-Kontingente, die Sie erhöhen können

Service Quotas ist ein AWS Service, mit dem Sie Ihre Kontingente oder Limits von einem Standort aus verwalten können. In der [Service Quotas Quotas-Konsole](#) können Sie Amazon EFS-Grenzwerte einsehen und eine Erhöhung des Kontingents für die Anzahl der EFS-Dateisysteme in einem AWS-Region und die gelesenen IOPS für häufig abgerufene Daten beantragen.

Sie können auch eine Erhöhung der folgenden Amazon-EFS-Kontingente beantragen, indem Sie sich an den AWS -Support wenden. Weitere Informationen hierzu finden Sie unter [Beantragen einer Kontingenterhöhung](#). Das Amazon-EFS-Serviceteam prüft jede Anfrage einzeln.

- Anzahl der Dateisysteme für jedes Kundenkonto.
- Anzahl der Zugriffspunkte für jedes Dateisystem.
- Maximale Lese-IOPS pro Dateisystem bei Verwendung von Elastic Throughput. Wenn die Lese-IOPS für Dateisysteme mit häufigem Zugriff erhöht werden, werden sowohl die Lese-IOPS für Dateisysteme, auf die selten zugegriffen wird, als auch die Schreib-IOPS erhöht.
- Elastischer Durchsatz pro regionalem Dateisystem für alle verbundenen Clients in einem AWS-Region
- Bereitgestellter Durchsatz pro regionalem Dateisystem für alle verbundenen Clients in einem AWS-Region.

In der folgenden Tabelle sind allgemeine Standardressourcenkontingente aufgeführt, deren Änderung Sie beantragen können.

Ressource	Standardkontingent
Anzahl der Dateisysteme für jedes Kundenkonto in einer AWS-Region	1.000
Anzahl der Zugriffspunkte für jedes Dateisystem	10.000
Maximale IOPS pro Dateisystem bei Verwendung von Elastic Throughput	Gelesene Daten, auf die selten zugegriffen wird: 90.000 Gelesene Daten, auf die häufig zugegriffen wird: 250.000 Schreiben Sie: 50.000
Sie können für Daten, auf die häufig zugegriffen wird, eine Erhöhung des Standardkontingents an Lese-IOPS um das bis zu 10-fache beantragen. Eine Erhöhung der Lese-IOPS für Daten, auf die häufig zugegriffen wird, führt auch zu einer Erhöhung der Lese-IOPS für Daten, auf die selten zugegriffen wird, und der Schreib-IOPS.	

In der folgenden Tabelle sind die Elastic Throughput-Kontingente pro Dateisystem für alle verbundenen Clients aufgeführt AWS-Region.

Regionale Dateisysteme — Gesamter elastischer Standarddurchsatz pro Dateisystem für alle verbundenen Clients AWS-Region

AWS-Region	Maximaler Lesedurchsatz	Maximaler Schreibdurchsatz (gemessener Durchsatz)
Region USA Ost (Ohio)	60 Gibibyte pro Sekunde () GiBps	5 GiBps
Region USA Ost (Nord-Virginia)		
US West (Oregon) Region		
Region Asien-Pazifik (Mumbai)		
Asia Pacific (Seoul) Region		
Region Asien-Pazifik (Singapur)		
Region Asien-Pazifik (Sydney)		
Asia Pacific (Tokyo) Region		
Region Europa (Frankfurt)		
Region Europa (Irland)		
Region Europa (London)		
Alle anderen AWS-Regionen	20 GiBps	1 GiBps

In der folgenden Tabelle sind die bereitgestellten Durchsatzquoten pro Dateisystem für alle verbundenen Clients in jedem AWS-Region Dateisystem aufgeführt.

Regionale Dateisysteme — Gesamter bereitgestellter Standarddurchsatz pro Dateisystem für alle verbundenen Clients AWS-Region

AWS-Region	Maximaler Lesedurchsatz	Maximaler Schreibdurchsatz (gemessener Durchsatz)
Region USA Ost (Ohio)	10 GiBps	3.33 GiBps
Region USA Ost (Nord-Virginia)		
Region USA West (Oregon)		
Region Europa (Irland)		
Alle anderen AWS-Regionen	3 GiBps	1 GiBps

Beantragen einer Kontingenterhöhung

Gehen Sie wie folgt vor AWS Support, um eine Erhöhung dieser Kontingente zu beantragen. Das Amazon-EFS-Team überprüft jede Anforderung zur Erhöhung des Kontingents.

Um eine Erhöhung des Kontingents zu beantragen über AWS Support

1. Öffnen Sie die Seite [AWS Support Center](#) und melden Sie sich bei Bedarf an. Wählen Sie dann Create Case (Fall erstellen) aus.
2. Wählen Sie unter Create case (Fall erstellen) die Option Service limit increase (Erhöhung des Service Limits) aus.
3. Wählen Sie für Limit Type (Limit-Typ) den Typ des Limits aus, das erhöht werden soll. Füllen Sie die erforderlichen Felder im Formular aus und wählen Sie dann Ihre bevorzugte Kontaktmethode aus.

Amazon-EFS-Ressourcenkontingente, die Sie nicht ändern können

Kontingente für mehrere Amazon-EFS-Ressourcen können nicht geändert werden, darunter:

- Kontingente für allgemeine Ressourcen, wie z. B. die Anzahl der Verbindungen für jedes Dateisystem.

- Elastische und bereitgestellte Durchsatzquoten pro One-Zone-Dateisystem für alle verbundenen Clients in einem AWS-Region.
- Erhöhung der Durchsatzquoten pro regionalem oder One-Zone-Dateisystem für alle verbundenen Clients in einem AWS-Region

In der folgenden Tabelle sind die allgemeinen Ressourcenkontingente aufgeführt, die nicht geändert werden können.

Allgemeine Ressourcenkontingente, die nicht geändert werden können

Ressource	Kontingent
Anzahl der Verbindungen für jedes Dateisystem	25,000
Anzahl der Mountingziele pro Dateisystem in einer Availability Zone	1
Anzahl der Mountingziele für jede Virtual Private Cloud (VPC)	1.400
Anzahl der Sicherheitsgruppen pro Mountingziel	5
Anzahl der Tags pro Dateisystem	50
Nummer von VPCs für jedes Dateisystem	1

Note

Clients können sich auch mit Mountingzielen verbinden, die sich in einem Konto oder einer VPC befinden, das bzw. die sich von dem des Dateisystems unterscheidet. Weitere Informationen finden Sie unter [Mounten von EFS-Dateisystemen von einer anderen AWS-Konto oder VPC](#).

In der folgenden Tabelle sind die gesamten standardmäßigen Durchsatzlimits für Elastic und Provisioned pro Dateisystem für alle verbundenen Clients aufgeführt. AWS-Region

Dateisysteme in einer Zone — Gesamter standardmäßiger Elastic- und Provisioned-Durchsatz pro Dateisystem für alle verbundenen Clients AWS-Region

AWS-Region	Maximaler Lesedurchsatz	Maximaler Schreibdurchsatz (gemessener Durchsatz)
Alle AWS-Regionen	3 GiBps	1 GiBps

In der folgenden Tabelle sind die Gesamtgrenzwerte für den Bursting-Durchsatz pro Dateisystem für alle verbundenen Clients aufgeführt. AWS-Region

Regionale Dateisysteme und Ein-Zonen-Dateisysteme — Gesamter Bursting-Durchsatz pro Dateisystem für alle verbundenen Clients AWS-Region

AWS-Region	Maximaler Lesedurchsatz	Maximaler Schreibdurchsatz
Region USA Ost (Ohio)	5 GiBps	3 GiBps
Region USA Ost (Nord-Virginia)		
Region USA West (Oregon)		
Region Asien-Pazifik (Sydney)		
Region Europa (Irland)		
Alle anderen AWS-Regionen	3 GiBps	1 GiBps

Kontingente für NFS-Clients

Die folgenden Kontingente für NFS-Clients gelten, vorausgesetzt, es handelt sich um einen Linux NFSv4 1.1-Client:

- Der maximale kombinierte Lese- und Schreibdurchsatz beträgt 1.500 Megabyte pro Sekunde (MiBps) für Dateisysteme, die Elastic Throughput verwenden und mit Version 2.0 oder höher des Amazon EFS-Clients (amazon-efs-utils Version) oder dem Amazon EFS CSI-Treiber (aws-efs-csi-driver) gemountet wurden. Der maximale Durchsatz für alle anderen Dateisysteme beträgt 500. MiBps Weitere Informationen zur Leistung finden Sie unter [Zusammenfassung der Leistung](#).

Der NFS-Clientdurchsatz wird als Gesamtanzahl der gesendeten und empfangenen Byte mit einer minimalen NFS-Anforderungsgröße von 4 KB (nach Anwenden einer 1/3-Messrate für Leseanforderungen) berechnet.

- Bis zu 65.536 aktive Benutzer pro Client können Dateien gleichzeitig öffnen.
- Auf der Instance werden bis zu 65.536 Dateien gleichzeitig geöffnet. Das Auflisten von Verzeichnisinhalten gilt nicht als Öffnen einer Datei.
- Jeder einzelne Mount auf dem Client kann insgesamt bis zu 65.536 Sperren pro Verbindung erwerben.
- Wenn Sie eine Verbindung mit Amazon EFS herstellen, können On-Premises-NFS-Clients oder NFS-Clients in einer anderen AWS-Region einen geringeren Durchsatz aufweisen, als wenn eine Verbindung zu EFS von der gleichen AWS-Region hergestellt wird. Dieser Effekt ist auf die erhöhte Netzwerklatenz zurückzuführen. Es ist eine Netzwerklatenz von höchstens 1 ms erforderlich, um den maximalen Durchsatz pro Client zu erreichen. Verwenden Sie den DataSync Datenmigrationsdienst, wenn Sie große Datensätze von lokalen NFS-Servern zu EFS migrieren.
- Das NFS-Protokoll unterstützt maximal 16 Gruppen IDs (GIDs) pro Benutzer, und alle weiteren Gruppen GIDs werden bei NFS-Client-Anfragen gekürzt. Weitere Informationen finden Sie unter [Zugriff auf zulässige Dateien im NFS-Dateisystem verweigert](#).
- Die Verwendung von Amazon EFS in Microsoft Windows wird nicht unterstützt.

Kontingente für Amazon-EFS-Dateisysteme

Die folgenden Kontingente sind für Amazon-EFS-Dateisysteme spezifisch.

Ressource	Kontingent
Länge des Dateinamens in Byte	255
Maximale Länge der symbolischen Verknüpfungen (Symlink) in Byte	4.080
Anzahl fester Verknüpfungen zu einer Datei	177
Größe einer Datei	52.673.613.135.872 Byte (47,9 TiB)
Anzahl der Ebenen für die Verzeichnistiefe	1.000

Ressource	Kontingent
Anzahl der Sperren für eine einzelne Datei für alle Instances und Benutzer	512
Zeichenlimit für jede Dateisystemrichtlinie	20 000
*Anzahl der Dateioperationen pro Sekunde im Modus „Allgemeine Zwecke“	250 000

*Weitere Informationen zur Anzahl der Dateioperationen pro Sekunde im Modus „Allgemeine Zwecke“ finden Sie unter [Zusammenfassung der Leistung](#).

Nicht unterstützte NFSv4 2.0- und 4.1-Funktionen

Amazon EFS unterstützt zwar nicht oder NFSv2 NFSv3, unterstützt aber sowohl NFSv4 .1 als auch NFSv4 .0, mit Ausnahme der folgenden Funktionen:

- pNFS
- Erweiterte Attribute
- Client-Delegation oder Callbacks jeglicher Art
 - Die Operation OPEN gibt immer OPEN_DELEGATE_NONE als Delegationstyp zurück.
 - Die Operation OPEN gibt NFSERR_NOTSUPP für die Anspruchstypen CLAIM_DELEGATE_CUR und CLAIM_DELEGATE_PREV zurück.
- Obligatorische Sperren

Alle Sperren in Amazon EFS sind empfohlene Sperren. Bei den Lese- und Schreiboperationen wird somit nicht geprüft, ob in Konflikt stehende Sperren vorhanden sind, bevor die Operation durchgeführt wird.

- Verweigerung der Freigabe

NFS unterstützt das Konzept einer Freigabeverweigerung. Eine Freigabeverweigerung wird hauptsächlich von Windows-Clients verwendet, damit Benutzer anderen den Zugriff auf eine bestimmte geöffnete Datei verweigern können. In Amazon EFS wird dies nicht unterstützt. Es wird der NFS-Fehler NFS4ERR_NOTSUPP zurückgegeben, wenn in OPEN-Befehlen ein anderer Wert

als `OPEN4_SHARE_DENY_NONE` für die Freigabeverweigerung angegeben wird. Linux-NFS-Clients verwenden keinen anderen Wert als `OPEN4_SHARE_DENY_NONE`.

- Zugriffskontrolllisten () ACLs
- In Amazon EFS wird das Attribut `time_access` beim Lesen von Dateien nicht aktualisiert. `time_access` wird von Amazon EFS in folgenden Fällen aktualisiert:
 - Beim Erstellen einer Datei (Erstellen eines Inode)
 - Wenn ein NFS-Client einen expliziten `setattr`-Aufruf vornimmt
 - Beim Schreiben auf den Inode, beispielsweise aufgrund von Dateigrößen- oder Dateimetadaten-Änderungen
 - Bei der Aktualisierung eines Inode-Attributs
- Namespaces
- Persistenter Antwort-Cache
- Kerberos-basierte Sicherheit
- NFSv4.1. Aufbewahrung von Daten
- SetUID auf Verzeichnissen
- Nicht unterstützte Dateitypen bei Verwendung des CREATE-Vorgangs: Blockgeräte (NF4BLK), Zeichengeräte (NF4CHR), Attributverzeichnis (NF4ATTRDIR) und benanntes Attribut (NAMEDATTR). NF4
- Nicht unterstützte Attribute: `FATTR4_ARCHIVE`, `FATTR4_FILES_AVAIL`, `FATTR4_FILES_FREE`, `FATTR4_FILES_TOTAL`, `FATTR4_FS_LOCATIONS`, `FATTR4_MIMETYPE`, `FATTR4_QUOTA_AVAIL_HARD`, `FATTR4_QUOTA_AVAIL_SOFT`, `FATTR4_QUOTA_USED`, `FATTR4_TIME_BACKUP` und `FATTR4_ACL`.

Beim Versuch, diese Attribute zu definieren, wird der Fehler `NFS4ERR_ATTRNOTSUPP` an den Client zurückgesendet.

Weitere Überlegungen

Beachten Sie außerdem Folgendes:

- Eine Liste der Orte AWS-Regionen , an denen Sie Amazon EFS-Dateisysteme erstellen können, finden Sie im [Allgemeine AWS-Referenz Benutzerhandbuch](#).
- Amazon EFS unterstützt die Mount-Option `nconnect` nicht.

- Sie können ein Amazon-EFS-Dateisystem von Servern in On-Premises-Rechenzentren mit Direct Connect und VPN erstellen. Weitere Informationen finden Sie unter [Tutorial: Mounten mit lokalen Clients](#).

Behebung von Fehlern bei Dateivorgängen im Zusammenhang mit Kontingenten

Wenn Sie auf EFS-Dateisysteme zugreifen, gelten bestimmte Beschränkungen für die Dateien im Dateisystem. Das Überschreiten dieser Einschränkungen führt zu Fehlern bei Dateivorgängen. Weitere Informationen zu dateibasierten Grenzwerten in Amazon EFS finden Sie unter [Amazon EFS-Kontingente](#)

Im Folgenden finden Sie einige gängige Fehler bei Dateivorgängen und die Einschränkungen, durch die diese hervorgerufen werden.

Themen

- [Suche nach offenen Dateien und Dateisperren](#)
- [Der Befehl schlägt mit dem Fehler „Disk quota exceeded“ fehl](#)
- [Befehl schlägt mit „E/A-Fehler“ fehl](#)
- [Befehl schlägt mit der Fehlermeldung „Dateiname ist zu lang“ fehl](#)
- [Befehl schlägt fehl mit dem Fehler „Datei nicht gefunden“](#)
- [Befehl schlägt mit der Fehlermeldung „Zu viele Links“ fehl](#)
- [Befehl schlägt mit der Fehlermeldung „Datei zu groß“ fehl.](#)

Suche nach offenen Dateien und Dateisperren

Um Fehler bei Dateioperationen zu beheben, können Sie überprüfen, ob Ihr Clientsystem die Grenzwerte erreicht hat, indem Sie offene Dateien und Dateisperren auf dem EFS-Bereitstellungspunkt untersuchen:

- Geöffnete Dateien überprüfen — Verwenden Sie die Tools Ihres Betriebssystems, um offene Dateien im EFS-Mountpfad aufzulisten. Auf diese Weise können Sie feststellen, ob Sie sich dem Limit für offene Dateien nähern. Unter Linux können Sie beispielsweise Folgendes verwenden:

```
ls -lsf <efs-mount-path>
```


- Dateisperren überprüfen — Verwenden Sie die Tools zur Sperrüberwachung Ihres Systems, um Dateien mit Sperren im EFS-Mountpfad aufzulisten. Auf diese Weise können Sie feststellen, ob die Grenzwerte für Sperren erreicht werden. Unter Linux können Sie beispielsweise Folgendes verwenden:

```
lslocks | grep <efs-mount-path>
```

Diese Befehle zeigen Ihnen die aktuelle Nutzung im Vergleich zu den EFS-Grenzwerten an, sodass Sie feststellen können, ob Dateibetriebsfehler mit dem Erreichen von System- oder Dienstlimits zusammenhängen.

Der Befehl schlägt mit dem Fehler „Disk quota exceeded“ fehl

Amazon EFS unterstützt derzeit keine Benutzer-Festplattenkontingente. Dieser Fehler kann auftreten, wenn einer der folgenden Grenzwerte überschritten wurde:

- Bis zu 65.536 aktive Benutzer können Dateien gleichzeitig öffnen. Ein Benutzerkonto, das mehrfach angemeldet ist, zählt als ein aktiver Benutzer.
- Für eine Instanz können bis zu 65.536 Dateien gleichzeitig geöffnet werden. Das Auflisten von Verzeichnisinhalten gilt nicht als Öffnen einer Datei.
- Jede einzelne Halterung auf dem Client kann insgesamt bis zu 65.536 Sperren pro Verbindung erwerben.

Maßnahme

Wenn dieses Problem auftritt, können Sie es beheben, indem Sie feststellen, welche dieser Einschränkungen Sie verletzen, und dann Änderungen vornehmen, um diese Einschränkung wieder einzuhalten. Weitere Informationen finden Sie unter [Kontingente für NFS-Clients](#). Informationen zur Überprüfung Ihrer aktuellen Nutzung finden Sie unter [Suche nach offenen Dateien und Dateisperren](#)

Befehl schlägt mit „E/A-Fehler“ fehl

Dieser Fehler tritt auf, wenn Sie mit einem der folgenden Probleme konfrontiert werden:

- Bei mehr als 65.536 aktiven Benutzerkonten pro Instanz sind Dateien gleichzeitig geöffnet.

Maßnahme

Wenn dieses Problem auftritt, können Sie es beheben, indem Sie das unterstützte Limit für offene Dateien auf Ihren Instances einhalten. Reduzieren Sie dazu die Anzahl der aktiven Benutzer, die gleichzeitig Dateien aus Ihrem Amazon-EFS-Dateisystem auf Ihren Instances geöffnet haben. Informationen zu Ihrer aktuellen Nutzung finden Sie unter. [Suche nach offenen Dateien und Dateisperren](#)

- Der AWS KMS Schlüssel, der Ihr Dateisystem verschlüsselt, wurde gelöscht.

Maßnahme

Wenn dieses Problem auftritt, können Sie die mit diesem Schlüssel einmal verschlüsselten Daten nicht mehr entschlüsseln. Das bedeutet, dass die Daten nicht wiederhergestellt werden können.

Befehl schlägt mit der Fehlermeldung „Dateiname ist zu lang“ fehl

Dieser Fehler tritt auf, wenn ein Dateiname oder seine symbolische Verknüpfung (symlink) zu lang ist. Für Dateinamen gelten die folgenden Beschränkungen:

- Ein Name kann bis zu 255 Byte lang sein.
- Ein symlink kann bis zu 4 080 Byte groß sein.

Maßnahme

Wenn dieses Problem auftritt, können Sie es beheben, indem Sie Ihren Dateinamen oder den symlink verkürzen, bis diese die unterstützten Grenzwerte einhalten.

Befehl schlägt fehl mit dem Fehler „Datei nicht gefunden“

Dieser Fehler tritt auf, weil einige ältere 32-Bit-Versionen der Oracle E-Business Suite I/O 32-Bit-Dateischnittstellen verwenden und EFS 64-Bit-Inode-Nummern verwendet. Systemaufrufe, die möglicherweise fehlschlagen, enthalten ``stat ()`` und ``readdir ()``.

Maßnahme

Wenn dieser Fehler auftritt, können Sie ihn mithilfe der `nfs.enable_ino64=0` kernel Boot-Option beheben. Diese Option komprimiert die 64-Bit-EFS-Inodenzahlen auf 32 Bit. Kernel-Boot-Optionen werden für verschiedene Linux-Distributionen unterschiedlich behandelt. Auf Amazon Linux aktivieren Sie diese Option, indem Sie `nfs.enable_ino64=0` kernel zur

GRUB_CMDLINE_LINUX_DEFAULT-Variablen in `/etc/default/grub`. Bitte konsultieren Sie Ihre Distribution für eine spezifische Dokumentation zum Aktivieren der Kernel-Boot-Optionen.

Befehl schlägt mit der Fehlermeldung „Zu viele Links“ fehl

Dieser Fehler tritt auf, wenn zu viele harte Links zu einer Datei bestehen. Sie können bis zu 177 harte Links in einer Datei haben.

Maßnahme

Wenn dieses Problem auftritt, können Sie es beheben, indem die Anzahl der harten Links zu einer Datei reduzieren, bis der Grenzwert eingehalten wird.

Befehl schlägt mit der Fehlermeldung „Datei zu groß“ fehl.

Dieser Fehler tritt auf, wenn eine Datei zu groß ist. Eine einzelne Datei kann bis zu 52.673.613.135872 Byte (47,9 TiB) groß sein.

Maßnahme

Wenn dieses Problem auftritt, können Sie es beheben, indem Sie die Größe einer Datei so reduzieren, dass sie den unterstützten Grenzwert einhält.

Amazon-EFS-API

Die Amazon-EFS-API ist ein Netzwerk-Protokoll basierend auf [HTTP \(RFC 2616\)](#). Für jeden API-Aufruf stellen Sie eine HTTP-Anfrage an den regionsspezifischen Amazon EFS-API-Endpunkt für den AWS-Region Ort, an dem Sie Dateisysteme verwalten möchten. Die API nutzt JSON-Dokumente (RFC 4627) für die HTTP-Anforderungs-/Antworttexte.

Die Amazon-EFS-API ist ein RPC-Modell. In diesem Modell gibt es einen festen Satz von Operationen, deren jeweilige Syntax den Clients ohne jede vorhergehende Interaktion bekannt ist. Im folgenden Abschnitt finden Sie eine Beschreibung für alle API-Operationen, die eine abstrakte RPC-Notation verwenden. Jeder verfügt über einen Operationsnamen, der nicht in den Wire-Daten zu sehen ist. Die jeweiligen Operationen werden den HTTP-Anforderungselementen zugeordnet.

Die genaue Amazon-EFS-Operation, der eine bestimmte Anforderung zugewiesen wird, hängt von der Kombination zweier Faktoren ab: der Anforderungsmethode (GET, PUT, POST oder DELETE) und dem Muster, das dem Anforderungs-URI entspricht. Bei einer PUT- oder POST-Operation extrahiert Amazon EFS die Aufrufargumente aus dem Anforderungs-URI-Pfadsegment, Abfrageparametern und dem JSON-Objekt im Anforderungstext.

Note

Operationsnamen, wie z. B. `CreateFileSystem`, erscheinen zwar nicht auf der Leitung, sind aber in AWS Identity and Access Management (IAM-) Richtlinien von Bedeutung. Weitere Informationen finden Sie unter [Identitäts- und Zugriffsmanagement für Amazon EFS](#). Der Operationsname wird auch zur Benennung von Befehlen in Befehlszeilentools und Elementen des SDK verwendet. AWS APIs Beispielsweise gibt es einen AWS CLI Befehl mit dem Namen `create-file-system`, der der `CreateFileSystem` Operation zugeordnet ist. Der Name des Vorgangs erscheint auch in den AWS CloudTrail Protokollen für Amazon EFS-API-Aufrufe.

API-Endpunkte

Ein Endpunkt ist eine URL, die als Einstiegspunkt für einen AWS Webservice dient. Amazon EFS unterstützt die folgenden Endpunkttypen:

- IPv4 Endpunkte

- Dual-Stack-Endpunkte (IPv4 sowohl IPv6 als auch)
- [FIPS-Endpunkte](#)

Wenn Sie eine Anfrage stellen, können Sie den zu verwendenden Endpunkt angeben. Wenn Sie keinen Endpunkt angeben, wird der IPv4 Endpunkt standardmäßig verwendet. Um einen anderen Endpunkttyp zu verwenden, müssen Sie ihn in Ihrer Anforderung angeben. Beispiele für diese Vorgehensweise finden Sie unter [Angaben von Endpunkten](#). Eine Tabelle der verfügbaren Endpunkte finden Sie unter [Amazon EFS-Endpunkte und](#) Kontingente.

IPv4 Endpunkte

IPv4 Endpunkte unterstützen nur IPv4 Datenverkehr. IPv4 Endpunkte sind für alle Regionen verfügbar. Das Format des IPv4 Endpunkts hängt von der Region ab.

- Für IPv4 Endgeräte in den AWS kommerziellen Regionen lautet `elasticfilesystem.region.amazonaws.com` das Format.

Wenn Sie beispielsweise `elasticfilesystem.us-west-2.amazonaws.com` als Endpunkt angeben, leiten wir Ihre Anfrage an den Endpunkt der Region USA West (Oregon) (`us-west-2`) weiter.

- Für IPv4 Endgeräte in den Regionen China lautet `elasticfilesystem.region.amazonaws.com.cn` das Format.

Der Amazon EFS-API-Endpunkt für die Region China (Peking) ist beispielsweise `elasticfilesystem.cn-north-1.amazonaws.com.cn`.

Dual-Stack-Endpunkte (IPv4 und IPv6)

Dual-Stack-Endpunkte unterstützen sowohl den als auch den Datenverkehr. IPv4 IPv6 Wenn Sie eine Anfrage an einen Dual-Stack-Endpunkt stellen, wird die Endpunkt-URL je nach dem von Ihrem Netzwerk und Client verwendeten Protokoll in eine IPv6 oder eine IPv4 Adresse aufgelöst. Das Format des Dual-Stack-Endpunkts hängt von der Region ab.

- Für Dual-Stack-Endpunkte in den AWS kommerziellen Regionen lautet das Format. `efs.aws-region.api.aws`

Der Dual-Stack-Endpoint für die Region USA West (Oregon) (us-west-2) lautet beispielsweise.
`efs.eu-west-2.api.aws`

- Für Dual-Stack-Endpoints in den Regionen China lautet das Format.
`efs.region.api.amazonaws.com.cn`

Der Dual-Stack-Endpoint für die Region China (Peking) ist `efs.cn-north-1.api.amazonaws.com.cn` beispielsweise.

Angeben von Endpunkten

Die folgenden Beispiele zeigen, wie Sie mithilfe von für Amazon EFS einen Endpoint für die Region USA West (Nordkalifornien) (us-west-1) angeben. AWS CLI

IPv4

```
aws efs get-rest-apis --region us-west-1 --endpoint-url https://elasticfilesystem.us-west-1.amazonaws.com
```

Dual-Stack

```
aws efs get-rest-apis --region us-west-1 --endpoint-url https://efs.us-west-1.api.aws
```

API-Version

Die für einen Aufruf verwendete API-Version wird vom ersten Pfadsegment des Anforderungs-URLs bestimmt und weist ein Datumsformat nach ISO 8601 auf. Ein Beispiel finden Sie unter [CreateFileSystem](#).

Die Beschreibung in der Dokumentation bezieht sich auf die API-Version 2015-02-01.

Verwandte Themen

In den folgenden Abschnitten erhalten Sie Beschreibungen der API-Operationen. Sie erfahren, wie Sie Signaturen zur Authentifizierung von Anforderungen erstellen und wie Sie mithilfe der IAM-Richtlinien Berechtigungen für die API-Operationen erteilen.

- [Identitäts- und Zugriffsmanagement für Amazon EFS](#)

- [Aktionen](#)
- [Datentypen](#)

Arbeiten mit der Abfrage-API-Anforderungsrate für Amazon EFS

Amazon EFS-API-Anfragen werden für jede AWS-Konto Anfrage pro Region gedrosselt, um die Serviceleistung zu verbessern. Alle Amazon EFS-API-Aufrufe zusammen, unabhängig davon, ob sie von einer Anwendung AWS CLI, der oder der Amazon EFS-Konsole stammen, dürfen die maximal zulässige API-Anforderungsrate nicht überschreiten. Die maximale API-Anforderungsrate kann von Land zu Land variieren AWS-Regionen. Die gestellten API-Anfragen werden dem zugrunde liegenden Objekt zugeordnet AWS-Konto.

Wenn eine API-Anforderung die API-Anforderungsrate für ihre Kategorie überschreitet, gibt die Anforderung den Fehlercode `ThrottlingException` zurück. Um diesen Fehler zu vermeiden, stellen Sie sicher, dass Ihre Anwendung API-Anfragen nicht in schneller Folge erneut versucht. Sie können dies tun, indem Sie beim Abrufen vorsichtig sind und Wiederholungen mit exponentiellem Backoff verwenden.

Abrufen

Möglicherweise muss Ihre Anwendung wiederholt eine API-Operation aufrufen, um auf ein Aktualisierung des Status zu prüfen. Bevor Sie mit dem Abrufen beginnen, geben Sie die Anforderungszeit für den potenziellen Abschluss ein. Wenn Sie mit dem Abrufen beginnen, verwenden Sie ein geeignetes Energiesparintervall zwischen aufeinanderfolgenden Anforderungen. Um die besten Ergebnisse zu erzielen, verwenden Sie ein zunehmendes Energiesparintervall.

Wiederholversuche oder Stapelverarbeitung

Möglicherweise muss Ihre Anwendung nach dem Auftreten eines Fehlers eine API-Anforderung wiederholen oder mehrere Ressourcen verarbeiten (z. B. all Ihre Amazon-EFS-Dateisysteme). Um die Rate von API-Anforderungen zu senken, verwenden Sie ein geeignetes Energiesparintervall zwischen aufeinanderfolgenden Anforderungen. Um die besten Ergebnisse zu erzielen, verwenden Sie ein zunehmendes oder variables Energiesparintervall.

Berechnen des Energiesparintervalls

Wenn Sie eine API-Anforderung abrufen oder wiederholen müssen, empfehlen wir die Verwendung eines exponentiellen Backoff-Algorithmus zum Berechnen des Energiesparintervalls zwischen API-

Aufrufen. Die Idee hinter dem exponentiellen Backoff ist, bei aufeinander folgenden Fehlermeldungen progressiv längere Wartezeiten zwischen den Wiederholversuchen zu verwenden. Weitere Informationen und Implementierungsbeispiele für diesen Algorithmus finden Sie unter [Verhalten bei Wiederholungsversuchen](#) in der Allgemeine Amazon Web Services-Referenz.

Aktionen

Folgende Aktionen werden unterstützt:

- [CreateAccessPoint](#)
- [CreateFileSystem](#)
- [CreateMountTarget](#)
- [CreateReplicationConfiguration](#)
- [CreateTags](#)
- [DeleteAccessPoint](#)
- [DeleteFileSystem](#)
- [DeleteFileSystemPolicy](#)
- [DeleteMountTarget](#)
- [DeleteReplicationConfiguration](#)
- [DeleteTags](#)
- [DescribeAccessPoints](#)
- [DescribeAccountPreferences](#)
- [DescribeBackupPolicy](#)
- [DescribeFileSystemPolicy](#)
- [DescribeFileSystems](#)
- [DescribeLifecycleConfiguration](#)
- [DescribeMountTargets](#)
- [DescribeMountTargetSecurityGroups](#)
- [DescribeReplicationConfigurations](#)
- [DescribeTags](#)
- [ListTagsForResource](#)
- [ModifyMountTargetSecurityGroups](#)

- [PutAccountPreferences](#)
- [PutBackupPolicy](#)
- [PutFileSystemPolicy](#)
- [PutLifecycleConfiguration](#)
- [TagResource](#)
- [UntagResource](#)
- [UpdateFileSystem](#)
- [UpdateFileSystemProtection](#)

CreateAccessPoint

Erstellt einen EFS-Zugangspunkt. Ein Zugangspunkt ist eine anwendungsspezifische Ansicht in ein EFS-Dateisystem, die einen Betriebssystembenutzer und eine Gruppe sowie einen Dateisystempfad auf jede über den Zugangspunkt erfolgte Dateisystemanforderung anwendet. Der Betriebssystembenutzer und die Gruppe überschreiben alle vom NFS-Client bereitgestellten Identitätsinformationen. Der Dateisystempfad wird als Stammverzeichnis des Zugangspunkts verfügbar gemacht. Anwendungen, die den Zugangspunkt verwenden, können nur auf Daten in einem eigenen Verzeichnis und die darunter liegende Ebene zugreifen. Ein Dateisystem kann maximal 10.000 Zugriffspunkte haben, sofern Sie keine Erhöhung beantragen. Weitere Informationen finden Sie unter [Mounten eines Dateisystems mithilfe von EFS-Zugangspunkten](#).

Note

Wenn mehrere Anfragen zur Erstellung von Zugriffspunkten auf demselben Dateisystem schnell hintereinander gesendet werden und das Dateisystem fast die maximale Anzahl an Zugriffspunkten erreicht, kann es bei diesen Anfragen zu einer Drosselung der Antwort kommen. Dadurch wird sichergestellt, dass das Dateisystem den angegebenen Grenzwert nicht überschreitet.

Diese Operation erfordert Berechtigungen für die Aktion `elasticfilesystem:CreateAccessPoint`.

Zugangspunkte können bei der Erstellung mit einem Tag versehen werden. Wenn Tags in der Aktion angegeben werden, mit der die Zugangspunkte erstellt werden, führt IAM eine zusätzliche Autorisierung für die Aktion `elasticfilesystem:TagResource` aus, um die Berechtigungen der Benutzer zum Erstellen von Tags zu überprüfen. Daher müssen Sie explizite Berechtigungen für die Verwendung der Aktion `elasticfilesystem:TagResource` gewähren. Weitere Informationen finden Sie unter [Erteilen der Berechtigung zum Taggen von Ressourcen während der Erstellung](#).

Anforderungssyntax

```
POST /2015-02-01/access-points HTTP/1.1
Content-type: application/json
```

```
{
  "ClientToken": "string",
  "FileSystemId": "string",
```

```
"PosixUser": {
  "Gid": number,
  "SecondaryGids": [ number ],
  "Uid": number
},
"RootDirectory": {
  "CreationInfo": {
    "OwnerGid": number,
    "OwnerUid": number,
    "Permissions": "string"
  },
  "Path": "string"
},
"Tags": [
  {
    "Key": "string",
    "Value": "string"
  }
]
}
```

URI-Anfrageparameter

Die Anforderung verwendet keine URI-Parameter.

Anforderungstext

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

ClientToken

Eine Zeichenfolge mit bis zu 64 ASCII-Zeichen, die Amazon EFS verwendet, um eine idempotente Erstellung sicherzustellen.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 64 Zeichen.

Pattern: . +

Erforderlich: Ja

FileSystemId

Die ID des EFS-Dateisystems, für das der Zugangspunkt Zugang gewährt.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Erforderlich: Ja

PosixUser

Der Betriebssystembenutzer und die Gruppe, die auf alle über den Zugangspunkt getätigten Dateisystemanfragen angewendet wurden.

Typ: [PosixUser](#) Objekt

Erforderlich: Nein

RootDirectory

Gibt das Verzeichnis auf dem EFS-Dateisystem an, das der Zugangspunkt als Stammverzeichnis des Dateisystems für NFS-Clients verfügbar macht, die den Zugangspunkt verwenden. Die Clients, die den Zugangspunkt verwenden, können nur auf das Stammverzeichnis und die darunter liegende Ebene zugreifen. Wenn das angegebene Verzeichnis `RootDirectory > Path` nicht vorhanden ist, wird es von EFS unter Anwendung der `CreationInfo`-Einstellungen erstellt, wenn ein Client eine Verbindung zu einem Zugangspunkt herstellt. Wenn Sie ein `RootDirectory` angeben, müssen Sie den Path und die `CreationInfo` bereitstellen.

Amazon EFS erstellt nur dann ein Stammverzeichnis, wenn Sie `CreationInfo: OwnUid`, `ownGID` und Berechtigungen für das Verzeichnis angegeben haben. Wenn Sie diese Informationen nicht angeben, erstellt Amazon EFS das Stammverzeichnis nicht. Wenn das Stammverzeichnis nicht existiert, schlagen Mount-Versuche beim Zugangspunkt fehl.

Typ: [RootDirectory](#) Objekt

Erforderlich: Nein

Tags

Erzeugt Tags, die dem Zugangspunkt zugeordnet sind. Jedes Tag ist ein Schlüssel-Wert-Paar, wobei jeder Schlüssel eineindeutig sein muss. Weitere Informationen finden Sie im [AWS Allgemeinen Referenzhandbuch unter AWS Ressourcen](#) kennzeichnen.

Typ: Array von [Tag](#)-Objekten

Erforderlich: Nein

Antwortsyntax

```
HTTP/1.1 200
Content-type: application/json

{
  "AccessPointArn": "string",
  "AccessPointId": "string",
  "ClientToken": "string",
  "FileSystemId": "string",
  "LifecycleState": "string",
  "Name": "string",
  "OwnerId": "string",
  "PosixUser": {
    "Gid": number,
    "SecondaryGids": [ number ],
    "Uid": number
  },
  "RootDirectory": {
    "CreationInfo": {
      "OwnerGid": number,
      "OwnerUid": number,
      "Permissions": "string"
    },
    "Path": "string"
  },
  "Tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ]
}
```

Antwortelemente

Wenn die Aktion erfolgreich ist, sendet der Service eine HTTP 200-Antwort zurück.

Die folgenden Daten werden vom Service im JSON-Format zurückgegeben.

AccessPointArn

Der eindeutige Amazon-Ressourcenname (ARN), der dem Zugangspunkt zugeordnet ist.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:access-point/fsap-[0-9a-f]{8,40}$`

AccessPointId

Die von Amazon EFS zugewiesene ID des Zugangspunkts.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:access-point/fsap-[0-9a-f]{8,40}|fsap-[0-9a-f]{8,40})$`

ClientToken

Die Opaque-Zeichenfolge, die in der Anforderung angegeben wird, um eine idempotente Erstellung zu gewährleisten.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 64 Zeichen.

Pattern: `.+`

FileSystemId

Die ID des EFS-Dateisystems, auf das der Zugangspunkt angewendet wird.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

LifeCycleState

Identifiziert die Lebenszyklusphase des Zugangspunkts.

Typ: Zeichenfolge

Zulässige Werte: `creating` | `available` | `updating` | `deleting` | `deleted` | `error`

Name

Der Name dieses Zugangspunkts. Dies ist der Wert des Name-Tags.

Typ: Zeichenfolge

OwnerId

Identifiziert den AWS-Konto , dem die Access Point-Ressource gehört.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 14 Zeichen.

Pattern: `^(\d{12})|(\d{4}-\d{4}-\d{4})$`

PosixUser

Die vollständige POSIX-Identität, einschließlich der Benutzer-ID, der Gruppen-ID und der sekundären Gruppe IDs auf dem Access Point, die für alle Dateioperationen von NFS-Clients verwendet wird, die den Access Point verwenden.

Typ: [PosixUser](#) Objekt

RootDirectory

Das Verzeichnis im EFS-Dateisystem, das der Zugangspunkt als Stammverzeichnis für NFS-Clients verfügbar macht, die den Zugangspunkts verwenden.

Typ: [RootDirectory](#) Objekt

Tags

Die mit dem Zugangspunkt verknüpften Tags, dargestellt als ein Array von Tag-Objekten.

Typ: Array von [Tag](#)-Objekten

Fehler

AccessPointAlreadyExists

Wird zurückgegeben, wenn der Zugangspunkt, den Sie erstellen möchten, bereits existiert, und zwar mit dem Erstellungstoken, das Sie in der Anfrage angegeben haben.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 409

AccessPointLimitExceeded

Wird zurückgegeben, wenn die AWS-Konto bereits die maximal zulässige Anzahl von Zugriffspunkten pro Dateisystem erstellt hat. Weitere Informationen finden Sie unter <https://docs.aws.amazon.com/efs/latest/ug/limits.html#limits-efs-resources-per-account-per-region>.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 403

BadRequest

Wird zurückgegeben, wenn die Anfrage fehlerhaft ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene FileSystemId Wert im Wert des Anforderers nicht vorhanden ist AWS-Konto.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

IncorrectFileSystemLifecycleState

Wird zurückgegeben, wenn der Lebenszyklusstatus des Dateisystems nicht „verfügbar“ ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 409

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 500

ThrottlingException

Wird zurückgegeben, wenn die API-Aktion `CreateAccessPoint` zu schnell aufgerufen wird und sich die Anzahl der Zugangspunkte im Dateisystem dem [Grenzwert von 120](#) nähert.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 429

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen finden Sie im AWS SDKs Folgenden:

- [AWS Befehlszeilenschnittstelle V2](#)
- [AWS SDK for .NET V4](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS SDK für Kotlin](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK for Ruby V3](#)

CreateFileSystem

Erstellt ein leeres Dateisystem. Die Operation erfordert ein Erstellungs-Token in der Anforderung, die Amazon EFS verwendet, um eine idempotente Erstellung zu gewährleisten (der Aufruf der Operation mit demselben Erstellungs-Token hat keine Wirkung). Wenn derzeit kein Dateisystem existiert, das dem Aufrufer AWS-Konto mit dem angegebenen Erstellungstoken gehört, führt diese Operation Folgendes aus:

- Erstellt ein leeres Dateisystem. Das Dateisystem hat eine von Amazon EFS zugewiesene ID und den anfänglichen Lebenszyklusstatus `creating`.
- Wird mit der Beschreibung des erstellten Dateisystems zurückgegeben.

Andernfalls gibt diese Operation einen `FileSystemAlreadyExists`-Fehler mit der ID des vorhandenen Dateisystems zurück.

Note

Bei Basis-Anwendungsfällen können Sie eine zufällig generierte UUID für das Erstellungs-Token verwenden.

Mit der idempotenten Operation können Sie den `CreateFileSystem`-Aufruf wiederholen, ohne das Risiko einzugehen, ein zusätzliches Dateisystem zu erstellen. Dies kann passieren, wenn ein erster Aufruf in einer Weise fehlschlägt, bei der ungewiss ist, ob tatsächlich ein Dateisystem erstellt wurde. Ein Beispiel könnte sein, dass ein Timeout für die Transportschicht aufgetreten ist oder Ihre Verbindung zurückgesetzt wurde. Solange Sie dasselbe Erstellungs-Token verwenden, kann der Client bei einer erfolgreichen Erstellung eines Dateisystems über den Fehler `FileSystemAlreadyExists` auf dessen Vorhandensein schließen.

Weitere Informationen finden Sie unter [Erstellen eines Dateisystems](#) im Amazon Elastic File System-Benutzerhandbuch.

Note

Der `CreateFileSystem`-Aufruf wird zurückgegeben, während der Lebenszyklusstatus des Dateisystems noch `creating` ist. Sie können den Erstellungstatus des Dateisystems

überprüfen, indem Sie die Operation [DescribeFileSystems](#) aufrufen. Diese gibt unter anderem den Status des Dateisystems zurück.

Diese Operation nimmt einen optionalen Parameter `PerformanceMode` entgegen, den Sie für das Dateisystem wählen. Wir empfehlen `generalPurpose` `PerformanceMode` für alle Dateisysteme. Der `maxIO` Modus ist ein Leistungstyp der vorherigen Generation, der für stark parallelisierte Workloads konzipiert wurde, die höhere Latenzen als der Modus tolerieren können. `generalPurpose` `MaxIO` Der Modus wird für One-Zone-Dateisysteme oder Dateisysteme, die Elastic Throughput verwenden, nicht unterstützt.

Der `PerformanceMode` kann nicht geändert werden, nachdem das Dateisystem erstellt wurde. Weitere Informationen finden Sie unter [Amazon EFS: Leistungsmodi](#).

Sie können den Durchsatzmodus für das Dateisystem mit dem Parameter `ThroughputMode` festlegen.

Nachdem das Dateisystem vollständig erstellt wurde, setzt Amazon EFS seinen Lebenszyklusstatus auf `available`, woraufhin Sie in Ihrer VPC ein oder mehrere Mount-Ziele für das Dateisystem erstellen können. Weitere Informationen finden Sie unter [CreateMountTarget](#). Sie mounten Ihr Amazon EFS-Dateisystem mithilfe des Mount-Ziels auf einer EC2 Instance in Ihrer VPC. Weitere Informationen finden Sie unter [Funktionsweise von Amazon EFS](#).

Diese Operation erfordert Berechtigungen für die Aktion `elasticfilesystem:CreateFileSystem`.

Dateisysteme können bei der Erstellung mit einem Tag versehen werden. Wenn Tags in der Aktion angegeben werden, mit der die Zugangspunkte erstellt werden, führt IAM eine zusätzliche Autorisierung für die Aktion `elasticfilesystem:TagResource` aus, um die Berechtigungen der Benutzer zum Erstellen von Tags zu überprüfen. Daher müssen Sie explizite Berechtigungen für die Verwendung der Aktion `elasticfilesystem:TagResource` gewähren. Weitere Informationen finden Sie unter [Erteilen der Berechtigung zum Taggen von Ressourcen während der Erstellung](#).

Anforderungssyntax

```
POST /2015-02-01/file-systems HTTP/1.1
Content-type: application/json

{
  "AvailabilityZoneName": "string",
```

```
{
  "Backup": boolean,
  "CreationToken": "string",
  "Encrypted": boolean,
  "KmsKeyId": "string",
  "PerformanceMode": "string",
  "ProvisionedThroughputInMibps": number,
  "Tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ],
  "ThroughputMode": "string"
}
```

URI-Anfrageparameter

Die Anforderung verwendet keine URI-Parameter.

Anforderungstext

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

AvailabilityZoneName

Geben Sie für One Zone-Dateisysteme die AWS Availability Zone an, in der das Dateisystem erstellt werden soll. Verwenden Sie das Format `us-east-1a`, um die Availability Zone anzugeben. Weitere Informationen zu One Zone-Dateisystemen finden Sie unter [EFS-Dateisystemtypen](#) im Amazon EFS-Benutzerhandbuch.

Note

Dateisysteme mit einer Zone sind nicht in allen Availability Zones verfügbar AWS-Regionen , in denen Amazon EFS verfügbar ist.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 64 Zeichen.

Pattern: `.+`

Erforderlich: Nein

Backup

Gibt an, ob für das Dateisystem, das Sie erstellen, automatische Backups aktiviert sind. Stellen Sie den Wert auf `true` ein, um automatische Backups zu aktivieren. Wenn Sie ein One-Zone-Dateisystem erstellen, sind automatische Backups standardmäßig aktiviert. Weitere Informationen finden Sie unter [Automatisierte Backups](#) im Amazon-EFS-Benutzerhandbuch.

Der Standardwert ist `false`. Wenn Sie jedoch einen `AvailabilityZoneName` angeben, lautet die Standardeinstellung `true`.

Note

AWS Backup ist nicht überall verfügbar AWS-Regionen , wo Amazon EFS verfügbar ist.

Typ: Boolesch

Erforderlich: Nein

CreationToken

Eine Zeichenfolge mit maximal 64 ASCII-Zeichen. Amazon EFS verwendet diese, um eine idempotente Erstellung zu gewährleisten.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 64 Zeichen.

Pattern: `.+`

Erforderlich: Ja

Encrypted

Ein boolescher Wert, der, wenn „true“, ein verschlüsseltes Dateisystem erstellt. Wenn Sie ein verschlüsseltes Dateisystem erstellen, haben Sie die Möglichkeit, einen vorhandenen AWS Key Management Service Schlüssel (KMS-Schlüssel) anzugeben. Wenn Sie keinen KMS-Schlüssel angeben, wird der standardmäßige KMS-Schlüssel für Amazon EFS, `/aws/elasticfilesystem`, verwendet, um das verschlüsselte Dateisystem zu schützen.

Typ: Boolesch

Erforderlich: Nein

KmsKeyId

Die ID des KMS-Schlüssels zum Schutz des verschlüsselten Dateisystems. Dieser Parameter ist nur erforderlich, wenn Sie einen nicht standardmäßigen KMS-Schlüssel verwenden möchten. Wenn dieser Parameter nicht angegeben ist, wird der standardmäßige KMS-Schlüssel für Amazon EFS verwendet. Sie können die ID des KMS-Schlüssels in den folgenden Formaten angeben:

- Schlüssel-ID – Eine eindeutige Kennzeichnung des Schlüssels, z. B. 1234abcd-12ab-34cd-56ef-1234567890ab.
- ARN – Ein Amazon-Ressourcenname (ARN) für den Schlüssel, z. B. `arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab`.
- Schlüsselalias: Ein zuvor erstellter Anzeigename für einen Schlüssel, z. B. `alias/projectKey1`.
- Schlüsselalias-ARN – Ein ARN für einen Schlüsselalias, z. B. `arn:aws:kms:us-west-2:444455556666:alias/projectKey1`.

Wenn Sie dies verwenden `KmsKeyId`, müssen Sie den [CreateFileSystemParameter:Encrypted auf true](#) setzen.



Important

EFS akzeptiert nur symmetrische KMS-Schlüssel. Sie können für Amazon-EFS-Dateisysteme keine asymmetrischen KMS-Schlüssel verwenden.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 2 048 Zeichen.

Pattern: `^([0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}|mrk-[0-9a-f]{32}|alias/[a-zA-Z0-9/_-]+|(arn:aws[-a-z]*:kms:[a-z0-9-]+:\d{12}:((key/[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12})|(key/mrk-[0-9a-f]{32})|(alias/[a-zA-Z0-9/_-]+))))$`

Erforderlich: Nein

PerformanceMode

Der Leistungsmodus des Dateisystems. Wir empfehlen für alle Dateisysteme den `generalPurpose`-Leistungsmodus. Dateisysteme, die den `maxIO`-Leistungsmodus verwenden,

können auf einen höheren Gesamtdurchsatz und mehr Operationen pro Sekunde skaliert werden, wobei bei den meisten Dateioperationen etwas höhere Latenzen auftreten. Der Leistungsmodus kann nach dem Anlegen des Dateisystems nicht mehr geändert werden. Der Modus `maxIO` wird in Dateisystemen, die One-Zone-Speicherklassen verwenden, nicht unterstützt.

 Important

Aufgrund der höheren Latenzen pro Vorgang beim Modus „Max. E/A“ empfehlen wir, für alle Dateisysteme den Allzweckleistungsmodus zu verwenden.

Der Standardwert ist `generalPurpose`.

Typ: Zeichenfolge

Zulässige Werte: `generalPurpose` | `maxIO`

Erforderlich: Nein

ProvisionedThroughputInMibps

Der Durchsatz, gemessen in Megabyte pro Sekunde (MiBps), den Sie für ein Dateisystem bereitstellen möchten, das Sie gerade erstellen. Erforderlich, wenn `ThroughputMode` auf `provisioned` festgelegt wird. Gültige Werte sind 1—3414 MiBps, wobei die Obergrenze von der Region abhängt. Um diesen Grenzwert zu erhöhen, wenden Sie sich an. Support Weitere Informationen finden Sie unter [Amazon-EFS-Kontingente, die Sie erhöhen können](#) im Amazon-EFS-Benutzerhandbuch.

Type: Double

Gültiger Bereich: Mindestwert 1.0.

Erforderlich: Nein

Tags

Wird verwendet, um ein oder mehrere Tags zu erstellen, die dem Dateisystem zugeordnet sind. Jeder Tag ist ein benutzerdefiniertes Schlüssel-Wert-Paar. Name Ihres Dateisystems bei der Erstellung durch Einschließen eines `"Key": "Name", "Value": "{value}"`-Schlüssel-Wert-Paars. Jeder Schlüssel muss eindeutig sein. Weitere Informationen finden Sie im AWS Allgemeinen Referenzhandbuch unter [AWS Ressourcen taggen](#).

Typ: Array von [Tag](#)-Objekten

Erforderlich: Nein

[ThroughputMode](#)

Gibt den Durchsatzmodus für das Dateisystem an. Der Modus kann `bursting`, `provisioned` oder `elastic` sein. Wenn `ThroughputMode` auf `provisioned` festgelegt ist, müssen Sie zudem einen Wert für `ProvisionedThroughputInMibps` angeben. Nachdem Sie das Dateisystem erstellt haben, können Sie den bereitgestellten Durchsatz des Dateisystems verringern oder mit bestimmten Zeitbeschränkungen zwischen den Durchsatzmodi wechseln. Weitere Informationen finden Sie unter [Angeben des Durchsatzes im Modus „Bereitgestellt“](#) im Amazon-EFS-Benutzerhandbuch.

Der Standardwert ist `bursting`.

Typ: Zeichenfolge

Zulässige Werte: `bursting` | `provisioned` | `elastic`

Erforderlich: Nein

Antwortsyntax

```
HTTP/1.1 201
Content-type: application/json

{
  "AvailabilityZoneId": "string",
  "AvailabilityZoneName": "string",
  "CreationTime": number,
  "CreationToken": "string",
  "Encrypted": boolean,
  "FileSystemArn": "string",
  "FileSystemId": "string",
  "FileSystemProtection": {
    "ReplicationOverwriteProtection": "string"
  },
  "KmsKeyId": "string",
  "LifeCycleState": "string",
  "Name": "string",
  "NumberOfMountTargets": number,
  "OwnerId": "string",
```

```

"PerformanceMode": "string",
"ProvisionedThroughputInMibps": number,
"SizeInBytes": {
  "Timestamp": number,
  "Value": number,
  "ValueInArchive": number,
  "ValueInIA": number,
  "ValueInStandard": number
},
"Tags": [
  {
    "Key": "string",
    "Value": "string"
  }
],
"ThroughputMode": "string"
}

```

Antwortelemente

Wenn die Aktion erfolgreich ist, sendet der Service eine HTTP-201-Antwort zurück.

Die folgenden Daten werden vom Service im JSON-Format zurückgegeben.

AvailabilityZoneId

Die eindeutige und konsistente Kennung der Availability Zone, in der sich das Dateisystem befindet. Sie ist nur für One-Zone-Dateisysteme gültig. use1-az1 ist beispielsweise eine Availability Zone ID für die US-East-1 AWS-Region, und sie hat in jedem Fall den gleichen Standort. AWS-Konto

Typ: Zeichenfolge

AvailabilityZoneName

Beschreibt die AWS Availability Zone, in der sich das Dateisystem befindet, und ist nur für One Zone-Dateisysteme gültig. Weitere Informationen finden Sie unter [Verwenden von EFS-Speicherklassen](#) im Amazon-EFS-Benutzerhandbuch.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 64 Zeichen.

Pattern: . +

[CreationTime](#)

Die Zeit, zu der das Dateisystem erstellt wurde, in Sekunden (seit 1970-01-01T00:00:00Z).

Typ: Zeitstempel

[CreationToken](#)

Die Opaque-Zeichenfolge, die in der Anforderung angegeben wird.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 64 Zeichen.

Pattern: .+

[Encrypted](#)

Ein boolescher Wert, der mit True anzeigt, dass das Dateisystem verschlüsselt ist.

Typ: Boolescher Wert

[FileSystemArn](#)

Der Amazon-Ressourcenname (ARN) für das EFS-Dateisystem, im Format `arn:aws:elasticfilesystem:region:account-id:file-system/file-system-id`. Beispiel mit Beispieldaten: `arn:aws:elasticfilesystem:us-west-2:1111333322228888:file-system/fs-01234567`

Typ: Zeichenfolge

[FileSystemId](#)

Die von Amazon EFS zugewiesene ID des Dateisystems.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

[FileSystemProtection](#)

Gibt den Schutz des Dateisystems an.

Typ: [FileSystemProtectionDescription](#) Objekt

KmsKeyId

Die ID eines, das zum Schutz des verschlüsselten Dateisystems AWS KMS key verwendet wird.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 2 048 Zeichen.

Pattern: `^([0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}|mrk-[0-9a-f]{32}|alias/[a-zA-Z0-9/_-]+|(arn:aws[-a-z]*:kms:[a-z0-9-]+:\d{12}:((key/[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12})|(key/mrk-[0-9a-f]{32})|(alias/[a-zA-Z0-9/_-]+))))$`

LifeCycleState

Die Lebenszyklusphase des Dateisystems.

Typ: Zeichenfolge

Zulässige Werte: `creating | available | updating | deleting | deleted | error`

Name

Sie können einem Dateisystem Tags hinzufügen, einschließlich eines Name-Tags. Weitere Informationen finden Sie unter [CreateFileSystem](#). Wenn das Dateisystem über ein Name-Tag verfügt, gibt Amazon EFS den Wert in diesem Feld zurück.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 256 Zeichen.

Pattern: `^([\p{L}\p{Z}\p{N}_.: /+=\ -@]*)$`

NumberOfMountTargets

Die aktuelle Anzahl von Mounting-Zielen, die das Dateisystem aufweist. Weitere Informationen finden Sie unter [CreateMountTarget](#).

Typ: Ganzzahl

Gültiger Bereich: Mindestwert 0.

OwnerId

AWS-Konto Derjenige, der das Dateisystem erstellt hat.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 14 Zeichen.

Pattern: `^(\\d{12})|(\\d{4}-\\d{4}-\\d{4})$`

[PerformanceMode](#)

Der Leistungsmodus des Dateisystems.

Typ: Zeichenfolge

Zulässige Werte: `generalPurpose` | `maxIO`

[ProvisionedThroughputInMibps](#)

Die Menge des bereitgestellten Durchsatzes, gemessen in MiBps, für das Dateisystem. Gültig für Dateisysteme, bei denen `ThroughputMode` auf `provisioned` eingestellt ist.

Type: Double

Gültiger Bereich: Mindestwert 1.0.

[SizeInBytes](#)

Die letzte bekannte gemessene Größe (in Bytes) der im Dateisystem gespeicherten Daten im Feld `Value` und die Zeit, zu der diese Größe ermittelt wurde, im Feld `Timestamp`. Der Wert `Timestamp` ist die ganzzahlige Anzahl der Sekunden seit 1970-01-01T00:00:00Z. Der Wert `SizeInBytes` steht nicht für die Größe eines konsistenten Snapshots des Dateisystems, ist aber letztlich konsistent, wenn keine Schreibvorgänge im Dateisystem vorgenommen werden. Das heißt, `SizeInBytes` steht nur dann für die tatsächliche Größe, wenn das Dateisystem länger als einige Stunden nicht verändert wurde. Andernfalls entspricht der Wert nicht exakt der Größe, die das Dateisystem zu einem beliebigen Zeitpunkt hatte.

Typ: [FileSystemSize](#) Objekt

[Tags](#)

Die Tags, die dem Dateisystem zugeordnet sind, dargestellt als ein Array von Tag-Objekten.

Typ: Array von [Tag](#)-Objekten

[ThroughputMode](#)

Zeigt den Durchsatzmodus des Dateisystems an. Weitere Informationen finden Sie unter [Durchsatzmodi](#) im Amazon-EFS-Benutzerhandbuch.

Typ: Zeichenfolge

Zulässige Werte: `bursting` | `provisioned` | `elastic`

Fehler

BadRequest

Wird zurückgegeben, wenn die Anfrage fehlerhaft ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

FileSystemAlreadyExists

Wird zurückgegeben, wenn das Dateisystem, das Sie erstellen möchten, bereits existiert, und zwar mit dem Erstellungstoken, das Sie angegeben haben.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können.

Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 409

FileSystemLimitExceeded

Wird zurückgegeben, wenn die AWS-Konto bereits die maximal zulässige Anzahl von Dateisystemen pro Konto erstellt hat.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 403

InsufficientThroughputCapacity

Wird zurückgegeben, wenn die Kapazität nicht ausreicht, um zusätzlichen Durchsatz bereitzustellen. Dieser Wert kann zurückgegeben werden, wenn Sie versuchen, ein Dateisystem im Modus „Bereitgestellter Durchsatz“ zu erstellen, wenn Sie versuchen, den bereitgestellten Durchsatz eines vorhandenen Dateisystems zu erhöhen oder wenn Sie versuchen, ein vorhandenes Dateisystem vom Modus „Bursting Throughput“ auf „Bereitgestellter Durchsatz“ umzustellen. Bitte versuchen Sie es später erneut.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden.

Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 503

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 500

ThroughputLimitExceeded

Wird zurückgegeben, wenn der Durchsatzmodus oder die Menge des bereitgestellten Durchsatzes nicht geändert werden können, weil die Durchsatzgrenze von 1024 erreicht MiB/s wurde.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können.

Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

UnsupportedAvailabilityZone

Wird zurückgegeben, wenn die angeforderte Amazon-EFS-Funktion in der angegebenen Availability Zone nicht verfügbar ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

Beispiele

Erstellen eines verschlüsselten Dateisystems

Im folgenden Beispiel wird eine POST-Anforderung gesendet, um ein Dateisystem in der Region us-west-2 mit aktivierten automatischen Backups zu erstellen. Die Anforderung gibt myFileSystem1 als Erstellungstoken für Idempotenz an.

Beispielanforderung

```
POST /2015-02-01/file-systems HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20140620T215117Z
Authorization: <...>
Content-Type: application/json
Content-Length: 42

{
```

```
"CreationToken" : "myFileSystem1",
"PerformanceMode" : "generalPurpose",
"Backup": true,
"Encrypted": true,
"Tags":[
  {
    "Key": "Name",
    "Value": "Test Group1"
  }
]
```

Beispielantwort

```
HTTP/1.1 201 Created
x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
Content-Type: application/json
Content-Length: 319
```

```
{
  "ownerId":"251839141158",
  "CreationToken":"myFileSystem1",
  "Encrypted": true,
  "PerformanceMode" : "generalPurpose",
  "fileSystemId":"fs-01234567",
  "CreationTime":"1403301078",
  "LifeCycleState":"creating",
  "numberOfMountTargets":0,
  "SizeInBytes":{
    "Timestamp": 1403301078,
    "Value": 29313618372,
    "ValueInArchive": 201156,
    "ValueInIA": 675432,
    "ValueInStandard": 29312741784
  },
  "Tags":[
    {
      "Key": "Name",
      "Value": "Test Group1"
    }
  ],
  "ThroughputMode": "elastic"
}
```

Erstellen eines verschlüsselten EFS-Dateisystems mit One-Zone-Verfügbarkeit

Im folgenden Beispiel wird eine POST-Anforderung gesendet, um ein Dateisystem in der Region us-west-2 mit aktivierten automatischen Backups zu erstellen. Das Dateisystem wird über einen One-Zone-Speicher in der Availability Zone us-west-2b verfügen.

Beispielanforderung

```
POST /2015-02-01/file-systems HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20140620T215117Z
Authorization: <...>
Content-Type: application/json
Content-Length: 42

{
  "CreationToken" : "myFileSystem2",
  "PerformanceMode" : "generalPurpose",
  "Backup": true,
  "AvailabilityZoneName": "us-west-2b",
  "Encrypted": true,
  "ThroughputMode": "elastic",
  "Tags":[
    {
      "Key": "Name",
      "Value": "Test Group1"
    }
  ]
}
```

Beispielantwort

```
HTTP/1.1 201 Created
x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
Content-Type: application/json
Content-Length: 319

{
  "ownerId":"251839141158",
  "CreationToken":"myFileSystem1",
  "Encrypted": true,
  "AvailabilityZoneId": "usew2-az2",
  "AvailabilityZoneName": "us-west-2b",
```

```
"PerformanceMode" : "generalPurpose",
"fileSystemId":"fs-01234567",
"CreationTime":"1403301078",
"LifeCycleState":"creating",
"numberOfMountTargets":0,
"SizeInBytes":{
  "Timestamp": 1403301078,
  "Value": 29313618372,
  "ValueInArchive": 201156,
  "ValueInIA": 675432,
  "ValueInStandard": 29312741784
},
"Tags":[
  {
    "Key": "Name",
    "Value": "Test Group1"
  }
],
"ThroughputMode": "elastic"
}
```

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen finden Sie im AWS SDKs Folgenden:

- [AWS Befehlszeilenschnittstelle V2](#)
- [AWS SDK for .NET V4](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS SDK für Kotlin](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK for Ruby V3](#)

CreateMountTarget

Erstellt ein Mountingziel für ein Dateisystem. Anschließend können Sie das Dateisystem mithilfe des Mount-Ziels auf EC2 Instanzen mounten. Weitere Informationen zu Mount-Zielen finden Sie unter [Mount-Ziele verwalten](#).

Um ein Mount-Ziel für ein EFS-Dateisystem zu erstellen, muss der Lebenszyklusstatus des Dateisystems `seinaavailable` sein. Weitere Informationen finden Sie unter [DescribeFileSystems](#).

Nach der Erstellung des Mount-Ziels gibt Amazon EFS eine Antwort zurück, die eine `MountTargetId` und eine IP-Adresse (`IpAddress` oder `Ipv6Address`) enthält. Sie verwenden diese IP-Adresse, wenn Sie das Dateisystem in einer EC2 Instance mounten. Sie können außerdem den DNS-Namen des Mount-Ziels beim Mounten des Dateisystems verwenden. Die EC2 Instanz, auf der Sie das Dateisystem mithilfe des Mount-Ziels mounten, kann den DNS-Namen des Mount-Ziels in seine IP-Adresse auflösen. Weitere Informationen finden Sie unter [So funktioniert Amazon EFS](#).

Beachten Sie, dass Sie Mount-Ziele für ein Dateisystem nur in einer VPC erstellen können. Es kann nur ein Mount-Ziel pro Availability Zone geben. Weitere Informationen finden Sie unter [Bereitstellungsziele erstellen](#).

Wenn die Anfrage die Anforderungen erfüllt, geht Amazon EFS wie folgt vor:

- Erstellt ein neues Mount-Ziel im angegebenen Subnetz.
- Erstellt eine neue Netzwerkschnittstelle im Subnetz mit folgenden Eigenschaften:
 - Die Beschreibung `Mount target fsmt-id for file system fs-id`, bei der *fsmt-id* es sich um die Mount-Ziel-ID *fs-id* handelt, und um die `FileSystemId`
 - Die `requesterManaged` Eigenschaft der Netzwerkschnittstelle ist auf `eingestellttrue`, und der `requesterId` Wert, der auf `gesetzt istEFS`.

Jedes Mount-Ziel hat eine entsprechende, vom Anforderer verwaltete EC2 Netzwerkschnittstelle. Nachdem die Netzwerkschnittstelle erstellt wurde, setzt Amazon EFS das `NetworkInterfaceId` Feld in der Beschreibung des Mount-Ziels auf die Netzwerkschnittstellen-ID und die IP-Adresse auf deren Adresse. Wenn die Erstellung der Netzwerkschnittstelle fehlschlägt, schlägt die gesamte `CreateMountTarget`-Operation fehl.

 Note

Der Aufruf von `CreateMountTarget` gibt erst nach der Erstellung der Netzwerkschnittstelle einen Wert zurück. Da aber der Status des Mountingziel weiterhin `creating` lautet, können Sie ihn durch Aufruf der Operation [DescribeMountTargets](#) überprüfen, die unter anderem den Status des Mountingziels zurückgibt.

Diese Operation erfordert Berechtigungen für die folgende Dateisystemaktion:

- `elasticfilesystem:CreateMountTarget`

Für diesen Vorgang sind auch Berechtigungen für die folgenden EC2 Amazon-Aktionen erforderlich:

- `ec2:DescribeSubnets`
- `ec2:DescribeNetworkInterfaces`
- `ec2:CreateNetworkInterface`

Anforderungssyntax

```
POST /2015-02-01/mount-targets HTTP/1.1
Content-type: application/json
```

```
{
  "FileSystemId": "string",
  "IpAddress": "string",
  "IpAddressType": "string",
  "Ipv6Address": "string",
  "SecurityGroups": [ "string" ],
  "SubnetId": "string"
}
```

URI-Anfrageparameter

Die Anforderung verwendet keine URI-Parameter.

Anforderungstext

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

FileSystemId

Die ID des Dateisystems, für das das Bereitstellungsziel erstellt werden soll.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Erforderlich: Ja

IpAddress

Wenn das `IpAddressType` Mount-Ziel IPv4 (`IPV4_ONLY` oder `DUAL_STACK`) lautet, geben Sie die zu IPv4 verwendende Adresse an. Wenn Sie keine angeben `IpAddress`, wählt Amazon EFS eine ungenutzte IP-Adresse aus dem Subnetz aus, für `SubnetId` das angegeben ist.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 7 Zeichen. Maximale Länge beträgt 15 Zeichen.

Pattern: `^[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}$`

Erforderlich: Nein

IpAddressType

Der IP-Adresstyp für das Mount-Ziel. Die möglichen Werte sind `IPV4_ONLY` (nur IPv4 Adressen), `IPV6_ONLY` (nur IPv6 Adressen) und `DUAL_STACK` (Dual-Stack, beide IPv6 Adressen IPv4 und Adressen). Wenn Sie keinen angeben `IpAddressType`, `IPV4_ONLY` wird verwendet.

Note

Das `IpAddressType` muss dem IP-Typ des Subnetzes entsprechen. Darüber hinaus überschreibt der `IpAddressType` Parameter den Wert, der als Standard-IP-Adresse für das Subnetz in der VPC festgelegt ist. Wenn beispielsweise `IpAddressType` is `IPV4_ONLY` und `true`, dann `AssignIpv6AddressOnCreation` IPv4 wird es für das Mount-Ziel verwendet. Weitere Informationen finden Sie unter [Ändern des öffentlichen IP-Adressierungsattributs Ihres Subnetzes](#).

Typ: Zeichenfolge

Zulässige Werte: IPV4_ONLY | IPV6_ONLY | DUAL_STACK

Erforderlich: Nein

Ipv6Address

Wenn das IPAddressType für das Einhängeziel IPv6 (IPV6_ONLY oder DUAL_STACK) lautet, geben Sie die zu verwendende IPv6 Adresse an. Wenn Sie keine angeben Ipv6Address, wählt Amazon EFS eine ungenutzte IP-Adresse aus dem Subnetz aus, für SubnetId das angegeben ist.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 3. Die maximale Länge beträgt 39.

Erforderlich: Nein

SecurityGroups

VPC-Sicherheitsgruppe IDs in der Form sg-xxxxxxx. Diese müssen für dieselbe VPC wie das angegebene Subnetz gelten. Die maximale Anzahl von Sicherheitsgruppen hängt vom Kontingent ab. Weitere Informationen finden Sie unter [Amazon VPC-Kontingente](#) im Amazon VPC-Benutzerhandbuch (siehe Tabelle mit Sicherheitsgruppen). Wenn Sie keine Sicherheitsgruppe angeben, verwendet Amazon EFS die Standardsicherheitsgruppe für die VPC des Subnetzes.

Typ: Zeichenfolgen-Array

Array-Mitglieder: Maximale Anzahl von 100 Elementen.

Längenbeschränkungen: Mindestlänge von 11. Maximale Länge von 43.

Pattern: ^sg-[0-9a-f]{8,40}

Erforderlich: Nein

SubnetId

Die ID des Subnetzes, in dem das Bereitstellungsziel hinzugefügt werden soll. Verwenden Sie für One-Zone-Dateisysteme, das Subnetz, das der Availability Zone des Dateisystems zugeordnet ist. Der Subnetztyp muss derselbe Typ sein wie der. IPAddressType

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 15 Zeichen. Maximale Länge beträgt 47 Zeichen.

Pattern: `^subnet-[0-9a-f]{8,40}$`

Erforderlich: Ja

Antwortsyntax

```
HTTP/1.1 200
Content-type: application/json

{
  "AvailabilityZoneId": "string",
  "AvailabilityZoneName": "string",
  "FileSystemId": "string",
  "IpAddress": "string",
  "Ipv6Address": "string",
  "LifeCycleState": "string",
  "MountTargetId": "string",
  "NetworkInterfaceId": "string",
  "OwnerId": "string",
  "SubnetId": "string",
  "VpcId": "string"
}
```

Antwortelemente

Wenn die Aktion erfolgreich ist, sendet der Service eine HTTP 200-Antwort zurück.

Die folgenden Daten werden vom Service im JSON-Format zurückgegeben.

AvailabilityZoneId

Die eindeutige und konsistente Kennung der Availability Zone, in der sich das Mountingziel befindet. `use1-az1` ist beispielsweise eine AZ-ID für die Region `us-east-1` und sie hat in jeder Region den gleichen Standort. AWS-Konto

Typ: Zeichenfolge

AvailabilityZoneName

Der Name der Availability Zone, in der sich das Mountingziel befindet. Availability Zones werden den jeweiligen Namen unabhängig voneinander zugeordnet. AWS-Konto Beispielsweise ist die Availability Zone us-east-1a für Sie AWS-Konto möglicherweise nicht derselbe Standort wie us-east-1a für eine andere AWS-Konto.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 64 Zeichen.

Pattern: .+

FileSystemId

Die ID des Dateisystems, für das das Mountingziel erstellt werden soll.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

IpAddress

Die IPv4 Adresse für das Mount-Ziel.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 7 Zeichen. Maximale Länge beträgt 15 Zeichen.

Pattern: `^[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}$`

Ipv6Address

Die IPv6 Adresse für das Mount-Ziel.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 3. Die maximale Länge beträgt 39.

LifeCycleState

Der Lebenszyklusstatus des Mounting-Ziels.

Typ: Zeichenfolge

Zulässige Werte: `creating` | `available` | `updating` | `deleting` | `deleted` | `error`

MountTargetId

Vom System zugewiesene ID für das Mountingziel.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 13 Zeichen. Maximale Länge beträgt 45 Zeichen.

Pattern: `^fsmt-[0-9a-f]{8,40}$`

NetworkInterfaceId

Die ID der Netzwerkschnittstelle, die Amazon EFS bei der Erstellung des Mountingziels erstellt hat.

Typ: Zeichenfolge

OwnerId

AWS-Konto ID, der die Ressource gehört.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 14 Zeichen.

Pattern: `^(\\d{12})|(\\d{4}-\\d{4}-\\d{4})$`

SubnetId

Die ID des Subnetzes des Mountingziels.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 15 Zeichen. Maximale Länge beträgt 47 Zeichen.

Pattern: `^subnet-[0-9a-f]{8,40}$`

VpcId

Die ID der Virtual Private Cloud (VPC), in der das Mountingziel konfiguriert ist.

Typ: Zeichenfolge

Fehler

AvailabilityZonesMismatch

Wird zurückgegeben, wenn sich die Availability Zone, die für das Mount-Ziel angegeben wurde, von der Availability Zone des Dateisystems unterscheidet. Weitere Informationen finden Sie unter [Redundanz von regionalem und One-Zone-Speicher](#).

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

BadRequest

Wird zurückgegeben, wenn die Anforderung falsch formatiert ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden.

Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

`FileSystemNotFound`

Wird zurückgegeben, wenn der angegebene `FileSystemId` Wert im Wert des Anforderers nicht vorhanden ist AWS-Konto.

`ErrorCode`

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

`Message`

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

`IncorrectFileSystemLifecycleState`

Wird zurückgegeben, wenn der Lebenszyklusstatus des Dateisystems nicht „verfügbar“ ist.

`ErrorCode`

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

`Message`

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können.

Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 409

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 500

IpAddressInUse

Wird zurückgegeben, wenn in der Anfrage eine IP-Adresse (IpAddressoderIpv6Address) angegeben wurde, die bereits im Subnetz verwendet wird.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 409

MountTargetConflict

Wird zurückgegeben, wenn das Mountingziel eine der auf der Grundlage der vorhandenen Mountingziele des Dateisystems angegebenen Einschränkungen verletzen würde.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 409

NetworkInterfaceLimitExceeded

Das aufrufende Konto hat den Grenzwert für elastische Netzwerkschnittstellen für die spezifische AWS-Region erreicht. Löschen Sie entweder einige Netzwerkschnittstellen oder fordern Sie eine Erhöhung des Kontingents an. Weitere Informationen finden Sie unter [Amazon VPC-Kontingente](#) im Amazon VPC-Benutzerhandbuch (siehe den Eintrag Netzwerkschnittstellen pro Region in der Tabelle Netzwerkschnittstellen).

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 409

NoFreeAddressesInSubnet

Hat „Nein“ zurückgegeben `IpAddress` oder `Ipv6Address` wurde in der Anfrage angegeben und es gibt keine freien IP-Adressen im angegebenen Subnetz.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 409

SecurityGroupLimitExceeded

Wird zurückgegeben, wenn die in der Anfrage `SecurityGroups` angegebene Anzahl von Daten den Grenzwert überschreitet, der auf dem Kontingent basiert. Löschen Sie entweder einige Sicherheitsgruppen oder fordern Sie eine Erhöhung des Kontokontingents an. Weitere Informationen finden Sie unter [Amazon VPC-Kontingente](#) im Amazon VPC-Benutzerhandbuch (siehe Tabelle mit Sicherheitsgruppen).

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

SecurityGroupNotFound

Wird zurückgegeben, wenn eine der angegebenen Sicherheitsgruppen nicht in der Virtual Private Cloud (VPC) des Subnetzes vorhanden ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

SubnetNotFound

Wird zurückgegeben, wenn in der Anforderung kein Subnetz mit der ID SubnetId angegeben wurde.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

UnsupportedAvailabilityZone

Wird zurückgegeben, wenn die angeforderte Amazon-EFS-Funktion in der angegebenen Availability Zone nicht verfügbar ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

Beispiele

Erstellen Sie ein Mount-Ziel an einer verfügbaren IPv4 Adresse in einem Subnetz

Die folgende Anfrage spezifiziert nur das Dateisystem und das Subnetz für das Mount-Ziel. Das Ziel wird an einer verfügbaren IPv4 Adresse im angegebenen Subnetz erstellt, wobei die Standardsicherheitsgruppe der VPC zugeordnet ist.

Beispielanforderung

```
POST /2015-02-01/mount-targets HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20140620T221118Z
Authorization: <...>
Content-Type: application/json
Content-Length: 160

{"SubnetId": "subnet-748c5d03", "FileSystemId": "fs-01234567"}
```

Beispielantwort

```
HTTP/1.1 200 OK
```

```
x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
Content-Type: application/json
Content-Length: 252

{
  "OwnerId": "231243201240",
  "MountTargetId": "fsmt-55a4413c",
  "FileSystemId": "fs-01234567",
  "SubnetId": "subnet-01234567",
  "LifecycleState": "available",
  "IpAddress": "172.31.22.183"
  "NetworkInterfaceId": "eni-1bcb7772"
  "AvailabilityZoneId": "eus1-az2",
  "AvailabilityZoneName": "eu-south-1b",
  "VpcId": "vpc-08d45b31fa009a15e"
}
```

Erstellen Sie ein Mount-Ziel an einer bestimmten Adresse IPv4

Die folgende Anfrage spezifiziert das Dateisystem, das Subnetz, die Sicherheitsgruppe und die IPv4 Adresse, die für das Mount-Ziel verwendet werden sollen.

Beispielanforderung

```
POST /2015-02-01/mount-targets HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20140620T221118Z
Authorization: <...>
Content-Type: application/json
Content-Length: 160

{
  "FileSystemId": "fs-01234567",
  "SubnetId": "subnet-01234567",
  "IpAddress": "10.0.2.42",
  "SecurityGroups": [
    "sg-01234567"
  ]
}
```

Beispielantwort

```
HTTP/1.1 200 OK
```

```
x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
Content-Type: application/json
Content-Length: 252

{
  "OwnerId": "251839141158",
  "MountTargetId": "fsmt-9a13661e",
  "FileSystemId": "fs-01234567",
  "SubnetId": "subnet-fd04ff94",
  "LifeCycleState": "available",
  "IpAddress": "10.0.2.42",
  "NetworkInterfaceId": "eni-1bcb7772",
  "AvailabilityZoneId": "eus1-az2",
  "AvailabilityZoneName": "eu-south-1b",
  "VpcId": "vpc-08d45b31fa009a15e"
}
```

Erstellen Sie ein Mount-Ziel an einer bestimmten Adresse IPv6

Die folgende Anfrage spezifiziert das Dateisystem, das Subnetz, die Sicherheitsgruppe und die IPv6 Adresse, die für das Mount-Ziel verwendet werden sollen.

Beispielanforderung

```
POST /2015-02-01/mount-targets HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20140620T221118Z
Authorization: <...>
Content-Type: application/json
Content-Length: 160

{
  "FileSystemId": "fs-01234567",
  "SubnetId": "subnet-01234567",
  "Ipv6Address": "2001:0db8:85a3:0000:0000:8a2e:0370:7334",
  "IpAddressType": "IPV6_ONLY",
  "SecurityGroups": [
    "sg-01234567"
  ]
}
```

Beispielantwort

```
HTTP/1.1 200 OK
x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
Content-Type: application/json
Content-Length: 252

{
  "OwnerId": "251839141158",
  "MountTargetId": "fsmt-9a13661e",
  "FileSystemId": "fs-01234567",
  "SubnetId": "subnet-fd04ff94",
  "LifeCycleState": "available",
  "Ipv6Address": "2001:0db8:85a3:0000:0000:8a2e:0370:7334",
  "NetworkInterfaceId": "eni-1bcb7772",
  "AvailabilityZoneId": "eus1-az2",
  "AvailabilityZoneName": "eu-south-1b",
  "VpcId": "vpc-08d45b31fa009a15e"
}
```

Erstellen Sie ein Mount-Ziel an einer verfügbaren IPv4 Adresse im Dual-Stack-Subnetz

Die folgende Anfrage spezifiziert das Dateisystem, das Subnetz und den Adresstyp für das Mount-Ziel. Das Ziel wird an einer verfügbaren IPv4 Adresse im angegebenen Dual-Stack-Subnetz erstellt, wobei die Standardsicherheitsgruppe der VPC zugeordnet ist.

Beispielanforderung

```
POST /2015-02-01/mount-targets HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20140620T221118Z
Authorization: <...>
Content-Type: application/json
Content-Length: 160

{"SubnetId": "subnet-748c5d03", "FileSystemId": "fs-01234567"}
```

Beispielantwort

```
HTTP/1.1 200 OK
x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
```

```
Content-Type: application/json
```

```
Content-Length: 252
```

```
{
  "OwnerId": "251839141158",
  "MountTargetId": "fsmt-55a4413c",
  "FileSystemId": "fs-01234567",
  "SubnetId": "subnet-fd04ff94",
  "LifecycleState": "available",
  "IpAddress": "172.31.22.183",
  "Ipv6Address": "2a05:d01a:419:8611:3e57:75ab:5719:b238",
  "NetworkInterfaceId": "eni-01234567",
  "AvailabilityZoneId": "eus1-az2",
  "AvailabilityZoneName": "eu-south-1b",
  "VpcId": "vpc-08d45b31fa009a15e"
}
```

Erstellen Sie ein Mount-Ziel für ein bestimmtes IPv4 und IPv6 für Adressen im Dual-Stack-Subnetz

Die folgende Anfrage spezifiziert das Dateisystem, das Subnetz, die Sicherheitsgruppe, die Adresse und die IPv4 IPv6 Adresse für das Mount-Ziel. Das Ziel wird an den angegebenen IPv4 IPv6 Adressen in einem Dual-Stack-Subnetz erstellt.

Beispielanforderung

```
POST /2015-02-01/mount-targets HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20140620T221118Z
Authorization: <...>
Content-Type: application/json
Content-Length: 160

{
  "FileSystemId": "fs-01234567",
  "SubnetId": "subnet-01234567",
  "IpAddressType": "DUAL_STACK",
  "IpAddress": "10.0.1.25",
  "Ipv6Address": "2001:0db8:85a3:0000:0000:8a2e:0370:7334",
  "SecurityGroups": [
    "sg-01234567"
  ]
}
```

Beispielantwort

```
HTTP/1.1 200 OK
x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
Content-Type: application/json
Content-Length: 252

{
  "OwnerId": "231243201240",
  "MountTargetId": "fsmt-55a4413c",
  "FileSystemId": "fs-01234567",
  "SubnetId": "subnet-01234567",
  "LifecycleState": "available",
  "IpAddress": "10.0.1.25",
  "Ipv6Address": "2001:0db8:85a3:0000:0000:8a2e:0370:7334",
  "NetworkInterfaceId": "eni-01234567",
  "AvailabilityZoneId": "eus1-az2",
  "AvailabilityZoneName": "eu-south-1b",
  "VpcId": "vpc-08d45b31fa009a15e"
}
```

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen finden Sie im Folgenden AWS SDKs:

- [AWS Befehlszeilenschnittstelle V2](#)
- [AWS SDK for .NET V4](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS SDK für Kotlin](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK for Ruby V3](#)

CreateReplicationConfiguration

Erstellt eine Replikationskonfiguration für ein neues oder ein vorhandenes EFS-Dateisystem. Weitere Informationen finden Sie unter [Amazon-EFS-Replikation](#) im Amazon-EFS-Benutzerhandbuch. In der Replikationskonfiguration ist Folgendes festgelegt:

- Quelldateisystem — Das EFS-Dateisystem, das Sie replizieren möchten.
- Zieldateisystem — Das Zieldateisystem, in das das Quelldateisystem repliziert wird. In einer Replikationskonfiguration kann es nur ein Zieldateisystem geben.

Note

Ein Dateisystem kann nur Teil einer Replikationskonfiguration sein.

Die Zielparameter für die Replikationskonfiguration hängen davon ab, ob Sie in ein neues Dateisystem oder in ein vorhandenes Dateisystem replizieren und ob Sie über ein anderes Dateisystem replizieren. AWS-Konten Weitere Informationen finden Sie unter [DestinationToCreate](#).

Diese Operation erfordert Berechtigungen für die Aktion `elasticfilesystem:CreateReplicationConfiguration`. Darüber hinaus sind je nachdem, wie Sie Dateisysteme replizieren, weitere Berechtigungen erforderlich. Weitere Informationen finden Sie unter [Erforderliche Berechtigungen für die Replikation](#) im Amazon EFS-Benutzerhandbuch.

Anforderungssyntax

```
POST /2015-02-01/file-systems/SourceFileSystemId/replication-configuration HTTP/1.1
Content-type: application/json
```

```
{
  "Destinations": [
    {
      "AvailabilityZoneName": "string",
      "FileSystemId": "string",
      "KmsKeyId": "string",
      "Region": "string",
      "RoleArn": "string"
    }
  ]
}
```

```
}
```

URI-Anfrageparameter

Die Anforderung verwendet die folgenden URI-Parameter.

[SourceFileSystemId](#)

Gibt das Amazon-EFS-Dateisystem an, das Sie replizieren möchten. Dieses Dateisystem kann in einer anderen Replikationskonfiguration kein Quell- oder Zieldateisystem sein.

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Erforderlich: Ja

Anforderungstext

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

[Destinations](#)

Ein Array von Objekten, die eine Zielkonfiguration beschreiben. Es wird nur ein Zielkonfigurationsobjekt unterstützt.

Typ: Array von [DestinationToCreate](#)-Objekten

Erforderlich: Ja

Antwortsyntax

```
HTTP/1.1 200
Content-type: application/json

{
  "CreationTime": number,
  "Destinations": [
    {
      "FileSystemId": "string",
```

```
    "LastReplicatedTimestamp": number,
    "OwnerId": "string",
    "Region": "string",
    "RoleArn": "string",
    "Status": "string",
    "StatusMessage": "string"
  }
],
"OriginalSourceFileSystemArn": "string",
"SourceFileSystemArn": "string",
"SourceFileSystemId": "string",
"SourceFileSystemOwnerId": "string",
"SourceFileSystemRegion": "string"
}
```

Antwortelemente

Wenn die Aktion erfolgreich ist, sendet der Service eine HTTP 200-Antwort zurück.

Die folgenden Daten werden vom Service im JSON-Format zurückgegeben.

CreationTime

Der Zeitpunkt der Erstellung der Replikationskonfiguration.

Typ: Zeitstempel

Destinations

Ein Array von Zielobjekten. Es wird nur ein Zielobjekt unterstützt.

Typ: Array von Destination-Objekten

OriginalSourceFileSystemArn

Der Amazon-Ressourcenname (ARN) des ursprünglichen EFS-Dateisystems in der Replikationskonfiguration.

Typ: Zeichenfolge

SourceFileSystemArn

Der Amazon-Ressourcenname (ARN) des aktuellen EFS-Dateisystems in der Replikationskonfiguration.

Typ: Zeichenfolge

SourceFileSystemId

Die ID des Amazon-EFS-Quelldateisystems, das repliziert wird.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

SourceFileSystemOwnerId

ID des Dateisystems, AWS-Konto in dem sich das Quelldateisystem befindet.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 14 Zeichen.

Pattern: `^(\\d{12})|(\\d{4}-\\d{4}-\\d{4})$`

SourceFileSystemRegion

Das, AWS-Region in dem sich das EFS-Quelldateisystem befindet.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 64 Zeichen.

Pattern: `^[a-z]{2}-((iso[a-z]{0,1}-)|(gov-)){0,1}[a-z]+-{0,1}[0-9]{0,1}$`

Fehler

BadRequest

Wird zurückgegeben, wenn die Anfrage fehlerhaft ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

ConflictException

Wird zurückgegeben, wenn das Quelldateisystem in einer Replikation verschlüsselt, das Zieldateisystem jedoch unverschlüsselt ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 409

FileSystemLimitExceeded

Wird zurückgegeben, wenn die AWS-Konto bereits die maximal zulässige Anzahl von Dateisystemen pro Konto erstellt hat.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 403

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene `FileSystemId` Wert im Wert des Anforderers nicht vorhanden ist AWS-Konto.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

IncorrectFileSystemLifecycleState

Wird zurückgegeben, wenn der Lebenszyklusstatus des Dateisystems nicht „verfügbar“ ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden.

Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können.

Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 409

InsufficientThroughputCapacity

Wird zurückgegeben, wenn die Kapazität nicht ausreicht, um zusätzlichen Durchsatz bereitzustellen. Dieser Wert kann zurückgegeben werden, wenn Sie versuchen, ein Dateisystem im Modus „Bereitgestellter Durchsatz“ zu erstellen, wenn Sie versuchen, den bereitgestellten Durchsatz eines vorhandenen Dateisystems zu erhöhen oder wenn Sie versuchen, ein vorhandenes Dateisystem vom Modus „Bursting Throughput“ auf „Bereitgestellter Durchsatz“ umzustellen. Bitte versuchen Sie es später erneut.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden.

Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können.

Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 503

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden.

Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 500

ReplicationNotFound

Wird zurückgegeben, wenn das angegebene Dateisystem keine Replikationskonfiguration aufweist.

ErrorCode

ReplicationNotFound

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

ThroughputLimitExceeded

Wird zurückgegeben, wenn der Durchsatzmodus oder die Menge des bereitgestellten Durchsatzes nicht geändert werden können, da die Durchsatzgrenze von 1024 erreicht MiB/s wurde.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können.

Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

UnsupportedAvailabilityZone

Wird zurückgegeben, wenn die angeforderte Amazon-EFS-Funktion in der angegebenen Availability Zone nicht verfügbar ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

ValidationException

Wird zurückgegeben, wenn der AWS Backup Dienst in dem Land, AWS-Region in dem die Anfrage gestellt wurde, nicht verfügbar ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen AWS SDKs finden Sie im Folgenden:

- [AWS Befehlszeilenschnittstelle V2](#)
- [AWS SDK for .NET V4](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS SDK für Kotlin](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK for Ruby V3](#)

CreateTags

Note

VERALTET – CreateTags ist veraltet und wird nicht mehr unterstützt. Sie können Tags für EFS-Ressourcen mit der API-Aktion [TagResource](#) erstellen.

Erstellt oder überschreibt einem Dateisystem zugeordnete Tags Jeder Tag ist ein Schlüssel/Wert-Paar. Wenn ein in der Anforderung angegebener Tag-Schlüssel bereits im Dateisystem vorhanden ist, wird dessen Wert durch diesen Vorgang mit dem in der Anforderung angegebenen Wert überschrieben. Wenn Sie das Tag Name zum Dateisystem hinzufügen, gibt Amazon EFS es als Antwort auf den die Operation [DescribeFileSystems](#) zurück.

Diese Operation setzt eine Berechtigung für die `elasticfilesystem:CreateTags`-Aktion voraus.

Anforderungssyntax

```
POST /2015-02-01/create-tags/FileSystemId HTTP/1.1
Content-type: application/json
```

```
{
  "Tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ]
}
```

URI-Anfrageparameter

Die Anforderung verwendet die folgenden URI-Parameter.

[FileSystemId](#)

Die ID des Dateisystems, dessen Tags Sie ändern möchten (Zeichenfolge). Durch diese Operation werden nur die Tags geändert, nicht das Dateisystem.

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Erforderlich: Ja

Anforderungstext

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

Tags

Ein Array von Tag-Objekten, die hinzugefügt werden sollen. Jedes Tag-Objekt ist ein Schlüssel-Wert-Paar.

Typ: Array von [Tag](#)-Objekten

Erforderlich: Ja

Antwortsyntax

```
HTTP/1.1 204
```

Antwortelemente

Wenn die Aktion erfolgreich ist, gibt der Dienst eine HTTP-204-Antwort mit leerem HTTP-Textinhalt zurück.

Fehler

BadRequest

Wird zurückgegeben, wenn die Anfrage fehlerhaft ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene `FileSystemId` Wert im Wert des Anforderers nicht vorhanden ist AWS-Konto.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden.

Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 500

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen finden Sie im AWS SDKs Folgenden:

- [AWS Befehlszeilenschnittstelle V2](#)
- [AWS SDK for .NET V4](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS SDK für Kotlin](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK for Ruby V3](#)

DeleteAccessPoint

Löscht den angegebenen Zugangspunkt. Nach Abschluss des Löschvorgangs können sich neue Clients nicht mehr mit den Zugangspunkten verbinden. Clients, die zum Zeitpunkt des Löschvorgangs mit dem Zugangspunkte verbunden waren, funktionieren bis zur Beendigung der Verbindung weiter.

Diese Operation erfordert Berechtigungen für die Aktion `elasticfilesystem:DeleteAccessPoint`.

Anforderungssyntax

```
DELETE /2015-02-01/access-points/AccessPointId HTTP/1.1
```

URI-Anfrageparameter

Die Anforderung verwendet die folgenden URI-Parameter.

AccessPointId

Die ID des Zugangspunkts, den Sie löschen möchten.

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-]+:access-point/fsap-[0-9a-f]{8,40}|fsap-[0-9a-f]{8,40})$`

Erforderlich: Ja

Anforderungstext

Der Anforderung besitzt keinen Anforderungstext.

Antwortsyntax

```
HTTP/1.1 204
```

Antwortelemente

Wenn die Aktion erfolgreich ist, gibt der Dienst eine HTTP-204-Antwort mit leerem HTTP-Textinhalt zurück.

Fehler

AccessPointNotFound

Wird zurückgegeben, wenn der angegebene `AccessPointId` Wert im Wert des Anforderers nicht vorhanden ist AWS-Konto.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

BadRequest

Wird zurückgegeben, wenn die Anfrage fehlerhaft ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 500

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen finden Sie im AWS SDKs Folgenden:

- [AWS Befehlszeilenschnittstelle V2](#)
- [AWS SDK for .NET V4](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS SDK für Kotlin](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK for Ruby V3](#)

DeleteFileSystem

Löscht ein Dateisystem und verhindert endgültig den Zugriff auf seinen Inhalt. Nach der Zurückgabe ist das Dateisystem nicht mehr vorhanden und Sie können nicht auf Inhalte des gelöschten Dateisystems zugreifen.

Sie müssen Mountingziele, die an ein Dateisystem angehängt sind, manuell löschen, bevor Sie ein EFS-Dateisystem löschen können. Dieser Schritt wird für Sie ausgeführt, wenn Sie die AWS Konsole zum Löschen eines Dateisystems verwenden.

Note

Sie können kein Dateisystem löschen, das Teil einer EFS-Replikationskonfiguration ist. Sie müssen zuerst die Replikationskonfiguration löschen.

Verwendete Dateisysteme können nicht gelöscht werden. Das bedeutet, dass Sie gegebenenfalls zuerst die Mountingziele des Dateisystems löschen müssen. Weitere Informationen erhalten Sie unter [DescribeMountTargets](#) und [DeleteMountTarget](#).

Note

Der DeleteFileSystem-Aufruf wird zurückgegeben, während der Systemstatus des Dateisystems noch `deleting` lautet. Sie können den Löschststatus des Dateisystems überprüfen, indem Sie die Operation [DescribeFileSystems](#) aufrufen, die eine Liste der Dateisysteme in Ihrem Konto zurückgibt. Wenn Sie die Dateisystem-ID oder das Erstellungstoken für das gelöschte Dateisystem übergeben, gibt die [DescribeFileSystems](#) den Fehler `404 FileSystemNotFound` zurück.

Diese Operation erfordert Berechtigungen für die Aktion `elasticfilesystem:DeleteFileSystem`.

Anforderungssyntax

```
DELETE /2015-02-01/file-systems/FileSystemId HTTP/1.1
```

URI-Anfrageparameter

Die Anforderung verwendet die folgenden URI-Parameter.

FileSystemId

Die ID des Dateisystems, das Sie löschen möchten.

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Erforderlich: Ja

Anforderungstext

Der Anforderung besitzt keinen Anforderungstext.

Antwortsyntax

```
HTTP/1.1 204
```

Antwortelemente

Wenn die Aktion erfolgreich ist, gibt der Dienst eine HTTP-204-Antwort mit leerem HTTP-Textinhalt zurück.

Fehler

BadRequest

Wird zurückgegeben, wenn die Anfrage fehlerhaft ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

FileSystemInUse

Wird zurückgegeben, wenn ein Dateisystem Mountingziele hat.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 409

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene FileSystemId Wert im Wert des Anforderers nicht vorhanden ist AWS-Konto.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden.

Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 500

Beispiele

Löschen eines Dateisystems

Das folgende Beispiel sendet eine DELETE-Anforderung an den Endpunkt `file-systems` (`elasticfilesystem.us-west-2.amazonaws.com/2015-02-01/file-systems/fs-01234567`), um ein Dateisystem zu löschen, dessen ID `fs-01234567` lautet.

Beispielanforderung

```
DELETE /2015-02-01/file-systems/fs-01234567 HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20140622T233021Z
Authorization: <...>
```

Beispielantwort

```
HTTP/1.1 204 No Content
x-amzn-RequestId: a2d125b3-7ebd-4d6a-ab3d-5548630bff33
Content-Length: 0
```

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen finden Sie im AWS SDKs Folgenden:

- [AWS Befehlszeilenschnittstelle V2](#)
- [AWS SDK for .NET V4](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS SDK für Kotlin](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK for Ruby V3](#)

DeleteFileSystemPolicy

Löscht die `FileSystemPolicy` für das angegebene Dateisystem. Die Standardeinstellung für `FileSystemPolicy` wird wirksam, sobald die vorhandene Richtlinie gelöscht wurde. Weitere Informationen zur standardmäßigen Dateisystemrichtlinie finden Sie unter [Verwenden von ressourcenbasierten Richtlinien mit EFS](#).

Diese Operation erfordert Berechtigungen für die Aktion `elasticfilesystem:DeleteFileSystemPolicy`.

Anforderungssyntax

```
DELETE /2015-02-01/file-systems/FileSystemId/policy HTTP/1.1
```

URI-Anfrageparameter

Die Anforderung verwendet die folgenden URI-Parameter.

FileSystemId

Gibt das EFS-Dateisystem an, für das die `FileSystemPolicy` gelöscht werden soll.

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Erforderlich: Ja

Anforderungstext

Der Anforderung besitzt keinen Anforderungstext.

Antwortsyntax

```
HTTP/1.1 200
```

Antwortelemente

Wenn die Aktion erfolgreich ist, gibt der Dienst eine HTTP 200-Antwort mit leerem HTTP-Textinhalt zurück.

Fehler

BadRequest

Wird zurückgegeben, wenn die Anfrage fehlerhaft ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene FileSystemId Wert im Wert des Anforderers nicht vorhanden ist AWS-Konto.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

IncorrectFileSystemLifecycleState

Wird zurückgegeben, wenn der Lebenszyklusstatus des Dateisystems nicht „verfügbar“ ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 409

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 500

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen finden Sie im AWS SDKs Folgenden:

- [AWS Befehlszeilenschnittstelle V2](#)
- [AWS SDK for .NET V4](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS SDK für Kotlin](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK for Ruby V3](#)

DeleteMountTarget

Löscht das angegebene Mountingziel

Wenn Sie das gelöschte Mountingziel verwenden, werden bei dieser Operation zwangsweise alle Dateisystem-Mounts aufgehoben. Dies könnte zu einer Störung der Instances oder Anwendungen führen, die diese Mounts verwenden. Um zu verhindern, dass Anwendungen abrupt getrennt werden, sollten Sie erwägen, alle Mounts des Mountingziels aufzuheben, sofern dies möglich ist. Bei dieser Operation wird auch die zugehörige Netzwerkschnittstelle gelöscht. Nicht festgeschriebene Schreibvorgänge können verloren gehen, jedoch bleibt das Dateisystem selbst intakt, wenn ein Mountingziel durch diese Operation aufgehoben wird. Das von Ihnen erstellte Dateisystem bleibt erhalten. Sie können eine EC2 Instance in Ihrer VPC mounten, indem Sie ein anderes Mount-Ziel verwenden.

Diese Operation erfordert Berechtigungen für die folgende Dateisystemaktion:

- `elasticfilesystem:DeleteMountTarget`

Note

Der Aufruf gibt `DeleteMountTarget` zurück, solange der Status des Mountingziels `deleting` lautet. Sie können überprüfen, ob das Mountingziel gelöscht wurde, indem Sie die Operation [DescribeMountTargets](#) aufrufen, die eine Liste von Beschreibungen der Mountingziele für das angegebene Dateisystem zurückgibt.

Für den Vorgang sind außerdem Berechtigungen für die folgende EC2 Amazon-Aktion auf der Netzwerkschnittstelle des Mount-Ziels erforderlich:

- `ec2:DeleteNetworkInterface`

Anforderungssyntax

```
DELETE /2015-02-01/mount-targets/MountTargetId HTTP/1.1
```

URI-Anfrageparameter

Die Anforderung verwendet die folgenden URI-Parameter.

MountTargetId

Die ID des Mountingziels, das gelöscht werden soll (Zeichenfolge).

Längenbeschränkungen: Mindestlänge von 13. Maximale Länge beträgt 45 Zeichen.

Pattern: `^fsmt-[0-9a-f]{8,40}$`

Erforderlich: Ja

Anforderungstext

Der Anforderung besitzt keinen Anforderungstext.

Antwortsyntax

```
HTTP/1.1 204
```

Antwortelemente

Wenn die Aktion erfolgreich ist, gibt der Dienst eine HTTP-204-Antwort mit leerem HTTP-Textinhalt zurück.

Fehler

BadRequest

Wird zurückgegeben, wenn die Anfrage fehlerhaft ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für Menschen vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können.

Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

DependencyTimeout

Bei dem Service ist beim Versuch, der Anforderung nachzukommen, eine Zeitüberschreitung aufgetreten, und der Client sollte den Aufruf wiederholen.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 504

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 500

MountTargetNotFound

Wird zurückgegeben, wenn kein Mountingziel mit der angegebenen ID im AWS-Konto des Aufrufers gefunden wurde.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

Beispiele

Entfernen des Mountingziels eines Dateisystems

Im folgenden Beispiel wird eine DELETE-Anfrage gesendet, um ein bestimmtes Mountingziel zu löschen.

Beispielanforderung

```
DELETE /2015-02-01/mount-targets/fsmt-9a13661e HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20140622T232908Z
Authorization: <...>
```

Beispielantwort

```
HTTP/1.1 204 No Content
x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
```

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen AWS SDKs finden Sie im Folgenden:

- [AWS Befehlszeilenschnittstelle V2](#)
- [AWS SDK for .NET V4](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS SDK für Kotlin](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK for Ruby V3](#)

DeleteReplicationConfiguration

Löscht eine Replikationskonfiguration. Durch das Löschen einer Replikationskonfiguration wird der Replikationsvorgang beendet. Nach dem Löschen einer Replikationskonfiguration wird das Zieldateisystem wieder `Writeable` und der Schutz vor Überschreibungen der Replikation wird wieder aktiviert. Weitere Informationen finden Sie unter [Löschen einer Replikationskonfiguration](#).

Diese Operation erfordert Berechtigungen für die Aktion `elasticfilesystem:DeleteReplicationConfiguration`.

Anforderungssyntax

```
DELETE /2015-02-01/file-systems/SourceFileSystemId/replication-configuration?
deletionMode=DeletionMode HTTP/1.1
```

URI-Anfrageparameter

Die Anforderung verwendet die folgenden URI-Parameter.

[DeletionMode](#)

Bei der Replikation zwischen AWS-Konten oder zwischen diesen AWS-Regionen löscht Amazon EFS die Replikationskonfiguration standardmäßig sowohl aus dem Quell- als auch aus dem Zielkonto oder der Region (`ALL_CONFIGURATIONS`). Wenn es ein Konfigurations- oder Berechtigungsproblem gibt, das Amazon EFS daran hindert, die Replikationskonfiguration von beiden Seiten zu löschen, können Sie den `LOCAL_CONFIGURATION_ONLY` Modus verwenden, um die Replikationskonfiguration nur von der lokalen Seite zu löschen (dem Konto oder der Region, von der aus der Löschvorgang durchgeführt wird).

Note

Verwenden Sie den `LOCAL_CONFIGURATION_ONLY` Modus nur für den Fall, dass Amazon EFS die Replikationskonfiguration nicht sowohl im Quell- als auch im Zielkonto oder in der Region löschen kann. Wenn Sie die lokale Konfiguration löschen, kann die Konfiguration im anderen Konto oder in der anderen Region nicht wiederhergestellt werden.

Verwenden Sie diesen Modus außerdem nicht für die Replikation mit demselben Konto und derselben Region, da dies zu einem Ausnahmefehler führt. `BadRequest`

Zulässige Werte: ALL_CONFIGURATIONS | LOCAL_CONFIGURATION_ONLY

SourceFileSystemId

Die ID des Quelldateisystems in der Replikationskonfiguration.

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Erforderlich: Ja

Anforderungstext

Der Anforderung besitzt keinen Anforderungstext.

Antwortsyntax

```
HTTP/1.1 204
```

Antwortelemente

Wenn die Aktion erfolgreich ist, gibt der Dienst eine HTTP-204-Antwort mit leerem HTTP-Textinhalt zurück.

Fehler

BadRequest

Wird zurückgegeben, wenn die Anfrage fehlerhaft ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden.

Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine

Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können.

Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene FileSystemId Wert im Wert des Anforderers nicht vorhanden ist. AWS-Konto

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden.

Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können.

Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden.

Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können.

Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 500

ReplicationNotFound

Wird zurückgegeben, wenn das angegebene Dateisystem keine Replikationskonfiguration aufweist.

ErrorCode

ReplicationNotFound

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen finden Sie im AWS SDKs Folgenden:

- [AWS Befehlszeilenschnittstelle V2](#)
- [AWS SDK for .NET V4](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS SDK für Kotlin](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK for Ruby V3](#)

DeleteTags

Note

VERALTET – DeleteTags ist veraltet und wird nicht mehr unterstützt. Sie können Tags mit der API-Aktion [UntagResource](#) aus einer EFS-Ressource entfernen.

Löscht die angegebenen Tags aus einem Dateisystem Wenn die Anforderung DeleteTags einen nicht vorhandenen Tag-Schlüssel enthält, wird sie von Amazon EFS ignoriert, ohne dass ein Fehler ausgegeben wird. Weitere Informationen zu Tags und damit verbundenen Einschränkungen finden Sie unter [Tag-Einschränkungen](#) im AWS Fakturierung und Kostenmanagement Benutzerhandbuch.

Diese Operation erfordert Berechtigungen für die Aktion `elasticfilesystem:DeleteTags`.

Anforderungssyntax

```
POST /2015-02-01/delete-tags/FileSystemId HTTP/1.1
Content-type: application/json

{
  "TagKeys": [ "string" ]
}
```

URI-Anfrageparameter

Die Anforderung verwendet die folgenden URI-Parameter.

[FileSystemId](#)

Die ID des Dateisystems, dessen Tags Sie löschen möchten (Zeichenfolge).

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Erforderlich: Ja

Anforderungstext

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

TagKeys

Eine Liste der Tag-Schlüssel, die gelöscht werden sollen.

Typ: Zeichenfolgen-Array

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Die maximale Anzahl beträgt 50 Elemente.

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 128 Zeichen.

Pattern: `^(?![aA]{1}[wW]{1}[sS]{1}:)([\p{L}\p{Z}\p{N}_.:/+=\ -@]+)$`

Erforderlich: Ja

Antwortsyntax

```
HTTP/1.1 204
```

Antwortelemente

Wenn die Aktion erfolgreich ist, gibt der Dienst eine HTTP-204-Antwort mit leerem HTTP-Textinhalt zurück.

Fehler

BadRequest

Wird zurückgegeben, wenn die Anfrage fehlerhaft ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden.

Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene `FileSystemId` Wert im Wert des Anforderers nicht vorhanden ist AWS-Konto.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 500

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen finden Sie im AWS SDKs Folgenden:

- [AWS Befehlszeilenschnittstelle V2](#)
- [AWS SDK for .NET V4](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS SDK für Kotlin](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK for Ruby V3](#)

DescribeAccessPoints

Gibt die Beschreibung eines bestimmten Amazon-EFS-Zugangspunkts zurück, sofern die `AccessPointId` angegeben ist. Wenn Sie eine `FileSystemId` für EFS angeben, werden Beschreibungen aller Zugangspunkte für dieses Dateisystem zurückgegeben. Sie können in der Anfrage entweder eine `AccessPointId` oder eine `FileSystemId` angeben, nicht jedoch beide.

Diese Operation erfordert Berechtigungen für die Aktion `elasticfilesystem:DescribeAccessPoints`.

Anforderungssyntax

```
GET /2015-02-01/access-points?  
AccessPointId=AccessPointId&FileSystemId=FileSystemId&MaxResults=MaxResults&NextToken=NextToken  
HTTP/1.1
```

URI-Anfrageparameter

Die Anforderung verwendet die folgenden URI-Parameter.

AccessPointId

(Optional) Gibt einen EFS-Zugangspunkt an, der in der Antwort beschrieben werden soll; ist eine sich mit `FileSystemId` ausschließende Option.

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:access-point/fsap-[0-9a-f]{8,40}|fsap-[0-9a-f]{8,40})$`

FileSystemId

(Optional) Wenn Sie eine `FileSystemId` angeben, gibt EFS alle Zugangspunkte für dieses Dateisystem zurück; ist eine sich mit `AccessPointId` ausschließende Option.

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

MaxResults

(Optional) Wenn Sie alle Zugangspunkte eines Dateisystems abrufen, können Sie optional den Parameter `MaxItems` angeben, um die Anzahl der in einer Antwort zurückgegebenen Objekte zu begrenzen. Der Standardwert lautet 100.

Gültiger Bereich: Mindestwert 1.

NextToken

`NextToken` ist vorhanden, wenn die Antwort paginiert ist. Sie können `NextMarker` in der nachfolgenden Anforderung verwenden, um die nächste Seite mit Beschreibungen von Zugangspunkten abzurufen.

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 128 Zeichen.

Pattern: `.+`

Anforderungstext

Der Anforderung besitzt keinen Anforderungstext.

Antwortsyntax

```
HTTP/1.1 200
Content-type: application/json

{
  "AccessPoints": [
    {
      "AccessPointArn": "string",
      "AccessPointId": "string",
      "ClientToken": "string",
      "FileSystemId": "string",
      "LifeCycleState": "string",
      "Name": "string",
      "OwnerId": "string",
      "PosixUser": {
        "Gid": number,
        "SecondaryGids": [ number ],
        "Uid": number
```

```
    },
    "RootDirectory": {
      "CreationInfo": {
        "OwnerGid": number,
        "OwnerUid": number,
        "Permissions": "string"
      },
      "Path": "string"
    },
    "Tags": [
      {
        "Key": "string",
        "Value": "string"
      }
    ]
  },
  "NextToken": "string"
}
```

Antwortelemente

Wenn die Aktion erfolgreich ist, sendet der Service eine HTTP 200-Antwort zurück.

Die folgenden Daten werden vom Service im JSON-Format zurückgegeben.

[AccessPoints](#)

Ein Array mit Beschreibungen von Zugangspunkten.

Typ: Array von [AccessPointDescription](#)-Objekten

[NextToken](#)

Vorhanden, wenn es mehr Zugangspunkte gibt, als in der Antwort zurückgegeben wurden. Sie können das NextMarker in der nachfolgenden Anfrage verwenden, um die zusätzlichen Beschreibungen abzurufen.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 128 Zeichen.

Pattern: . +

Fehler

AccessPointNotFound

Wird zurückgegeben, wenn der angegebene `AccessPointId` Wert in dem Wert des Anforderers nicht vorhanden ist. AWS-Konto

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

BadRequest

Wird zurückgegeben, wenn die Anfrage fehlerhaft ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene `FileSystemId` Wert im Wert des Anforderers nicht vorhanden ist. AWS-Konto

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 500

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen finden Sie im AWS SDKs Folgenden:

- [AWS Befehlszeilenschnittstelle V2](#)
- [AWS SDK for .NET V4](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS SDK für Kotlin](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK for Ruby V3](#)

DescribeAccountPreferences

Gibt die Kontoeinstellungen für das Konto zurück, das dem Benutzer AWS-Konto zugeordnet ist, der die Anfrage gestellt hat, in der aktuellen Version AWS-Region.

Anforderungssyntax

```
GET /2015-02-01/account-preferences HTTP/1.1
Content-type: application/json
```

```
{
  "MaxResults": number,
  "NextToken": "string"
}
```

URI-Anfrageparameter

Die Anforderung verwendet keine URI-Parameter.

Anforderungstext

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

MaxResults

(Optional) Wenn Sie alle Kontoeinstellungen abrufen, können Sie optional den Parameter `MaxItems` angeben, um die Anzahl der in einer Antwort zurückgegebenen Objekte zu begrenzen. Der Standardwert lautet 100.

Typ: Ganzzahl

Gültiger Bereich: Mindestwert 1.

Erforderlich: Nein

NextToken

(Optional) Sie können `NextToken` in einer nachfolgenden Anfrage verwenden, um die nächste Seite mit Einstellungen zum AWS-Konto abzurufen, wenn die Antwortnutzlast paginiert wurde.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 128 Zeichen.

Pattern: .+

Erforderlich: Nein

Antwortsyntax

```
HTTP/1.1 200
Content-type: application/json

{
  "NextToken": "string",
  "ResourceIdPreference": {
    "ResourceIdType": "string",
    "Resources": [ "string" ]
  }
}
```

Antwortelemente

Wenn die Aktion erfolgreich ist, sendet der Service eine HTTP 200-Antwort zurück.

Die folgenden Daten werden vom Service im JSON-Format zurückgegeben.

[NextToken](#)

Vorhanden, wenn es mehr Datensätze gibt, als in der Antwort zurückgegeben wurden. Sie können NextToken in einer nachfolgenden Anforderung verwenden, um die zusätzlichen Beschreibungen abzurufen.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 128 Zeichen.

Pattern: .+

[ResourceIdPreference](#)

Beschreibt die Einstellung für die Ressourcen-ID, die dem Benutzer AWS-Konto zugeordnet ist, der die Anfrage gestellt hat, in der aktuellen Version AWS-Region.

Typ: [ResourcePreference](#) Objekt

Fehler

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 500

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen AWS SDKs finden Sie im Folgenden:

- [AWS Befehlszeilenschnittstelle V2](#)
- [AWS SDK for .NET V4](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS SDK für Kotlin](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK für Python](#)

- [AWS SDK for Ruby V3](#)

DescribeBackupPolicy

Gibt die Backup-Richtlinie für das angegebene EFS-Dateisystem zurück.

Anforderungssyntax

```
GET /2015-02-01/file-systems/FileSystemId/backup-policy HTTP/1.1
```

URI-Anfrageparameter

Die Anforderung verwendet die folgenden URI-Parameter.

FileSystemId

Gibt an, für welches EFS-Dateisystem die BackupPolicy abgerufen werden soll.

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Erforderlich: Ja

Anforderungstext

Der Anforderung besitzt keinen Anforderungstext.

Antwortsyntax

```
HTTP/1.1 200
Content-type: application/json

{
  "BackupPolicy": {
    "Status": "string"
  }
}
```

Antwortelemente

Wenn die Aktion erfolgreich ist, sendet der Service eine HTTP 200-Antwort zurück.

Die folgenden Daten werden vom Service im JSON-Format zurückgegeben.

[BackupPolicy](#)

Beschreibt die Backup-Richtlinie des Dateisystems und gibt an, ob automatische Backups aktiviert oder deaktiviert sind.

Typ: [BackupPolicy](#) Objekt

Fehler

BadRequest

Wird zurückgegeben, wenn die Anfrage fehlerhaft ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene `FileSystemId` Wert im Wert des Anforderers nicht vorhanden ist AWS-Konto.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 500

PolicyNotFound

Wird zurückgegeben, wenn die angegebene IAM-Richtlinie oder Ressourcenrichtlinie nicht gefunden oder darauf zugegriffen werden kann. Je nach ausgeführter Aktion kann der Fehler auch zurückgegeben werden, wenn die Standard-Dateisystemrichtlinie für das EFS-Dateisystem gültig ist oder wenn keine Backup-Richtlinie angegeben ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

ValidationException

Wird zurückgegeben, wenn der AWS Backup Dienst in dem Land, AWS-Region in dem die Anfrage gestellt wurde, nicht verfügbar ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen AWS SDKs finden Sie im Folgenden:

- [AWS Befehlszeilenschnittstelle V2](#)
- [AWS SDK for .NET V4](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)

- [AWS SDK for Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS SDK für Kotlin](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK for Ruby V3](#)

DescribeFileSystemPolicy

Gibt die `FileSystemPolicy` für das angegebene EFS-Dateisystem zurück.

Diese Operation erfordert Berechtigungen für die Aktion `elasticfilesystem:DescribeFileSystemPolicy`.

Anforderungssyntax

```
GET /2015-02-01/file-systems/FileSystemId/policy HTTP/1.1
```

URI-Anfrageparameter

Die Anforderung verwendet die folgenden URI-Parameter.

FileSystemId

Gibt an, für welches EFS-Dateisystem die `FileSystemPolicy` abgerufen werden soll.

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Erforderlich: Ja

Anforderungstext

Der Anforderung besitzt keinen Anforderungstext.

Antwortsyntax

```
HTTP/1.1 200
Content-type: application/json

{
  "FileSystemId": "string",
  "Policy": "string"
}
```

Antwortelemente

Wenn die Aktion erfolgreich ist, sendet der Service eine HTTP 200-Antwort zurück.

Die folgenden Daten werden vom Service im JSON-Format zurückgegeben.

FileSystemId

Gibt das EFS-Dateisystem an, für das die FileSystemPolicy gilt.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Policy

Die FileSystemPolicy im JSON-Format für das EFS-Dateisystem.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge von 20 000.

Pattern: `[\s\S]+`

Fehler

BadRequest

Wird zurückgegeben, wenn die Anfrage fehlerhaft ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine

Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können.

Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene FileSystemId Wert im Wert des Anforderers nicht vorhanden ist AWS-Konto.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden.

Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können.

Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden.

Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können.

Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 500

PolicyNotFound

Wird zurückgegeben, wenn die angegebene IAM-Richtlinie oder Ressourcenrichtlinie nicht gefunden oder darauf zugegriffen werden kann. Je nach ausgeführter Aktion kann der Fehler auch zurückgegeben werden, wenn die Standard-Dateisystemrichtlinie für das EFS-Dateisystem gültig ist oder wenn keine Backup-Richtlinie angegeben ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

Beispiele

Beispiel

Dieses Beispiel veranschaulicht eine Verwendung von `DescribeFileSystemPolicy`.

Beispielanforderung

```
GET /2015-02-01/file-systems/fs-01234567/policy HTTP/1.1
```

Beispielantwort

```
{
  "FileSystemId": "fs-01234567",
  "Policy": "{
    \"Version\": \"2012-10-17\",
    \"Id\": \"efs-policy-wizard-cdef0123-aaaa-6666-5555-444455556666\",
```

```

    "Statement": [
      {
        "Sid": "efs-statement-abcdef01-1111-bbbb-2222-111122224444",
        "Effect" : "Deny",
        "Principal": {
          "AWS": "*"
        },
        "Action": "*",
        "Resource": "arn:aws:elasticfilesystem:us-east-2:111122223333:file-
system/fs-01234567",
        "Condition": {
          "Bool": {
            "aws:SecureTransport": "false"
          }
        }
      },
      {
        "Sid": "efs-statement-01234567-aaaa-3333-4444-111122223333",
        "Effect": "Allow",
        "Principal": {
          "AWS": "*"
        },
        "Action": [
          "elasticfilesystem:ClientMount",
          "elasticfilesystem:ClientWrite"
        ],
        "Resource" : "arn:aws:elasticfilesystem:us-east-2:111122223333:file-
system/fs-01234567"
      }
    ]
  }
}

```

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen AWS SDKs finden Sie im Folgenden:

- [AWS Befehlszeilenschnittstelle V2](#)
- [AWS SDK for .NET V4](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)

- [AWS SDK for Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS SDK für Kotlin](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK for Ruby V3](#)

DescribeFileSystems

Gibt die Beschreibung eines bestimmten Amazon-EFS-Dateisystems zurück, wenn entweder das Dateisystem `CreationToken` oder die `FileSystemId` angegeben ist. Andernfalls gibt es Beschreibungen aller Dateisysteme zurück, die dem Aufrufer gehören, AWS-Konto auf dem AWS-Region Endpunkt, den Sie aufrufen.

Beim Abrufen aller Dateisystembeschreibungen können Sie optional den Parameter `MaxItems` angeben, um die Anzahl der Beschreibungen in einer Antwort zu begrenzen. Diese Zahl wird automatisch auf 100 gesetzt. Wenn weitere Dateisystembeschreibungen übrig bleiben, gibt Amazon EFS in der Antwort einen `NextMarker`, ein Opaque-Token, zurück. In diesem Fall sollten Sie eine nachfolgende Anforderung senden, bei der der Anforderungsparameter `Marker` auf den Wert `NextMarker` gesetzt ist.

Um eine Liste der Dateisystembeschreibungen abzurufen, wird diese Operation in einem iterativen Prozess verwendet, wobei `DescribeFileSystems` zuerst ohne den `Marker` und dann von der Operation so lange aufgerufen wird, bis die Antwort keine `NextMarker` mehr aufweist, wobei der Parameter `Marker` auf den Wert `NextMarker` aus der vorherigen Antwort gesetzt ist.

Die Reihenfolge der Dateisysteme, die in der Antwort auf einen `DescribeFileSystems`- Aufruf zurückgegeben werden, und die Reihenfolge der Dateisysteme, die in den Antworten einer Iteration mit mehreren Aufrufen zurückgegeben werden, ist nicht angegeben.

Diese Operation erfordert Berechtigungen für die Aktion `elasticfilesystem:DescribeFileSystems`.

Anforderungssyntax

```
GET /2015-02-01/file-systems?  
CreationToken=CreationToken&FileSystemId=FileSystemId&Marker=Marker&MaxItems=MaxItems  
HTTP/1.1
```

URI-Anfrageparameter

Die Anforderung verwendet die folgenden URI-Parameter.

CreationToken

(Optional) Beschränkt die Liste auf das Dateisystem mit diesem Erstellungstoken (Zeichenfolge). Ein Erstellungstoken geben Sie bei der Erstellung eines Amazon-EFS-Dateisystems an.

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 64 Zeichen.

Pattern: . +

FileSystemId

(Optional) ID des Dateisystems, dessen Beschreibung Sie abrufen möchten (Zeichenfolge).

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Marker

(Optional) Opaque-Paginierungstoken, das von einer vorherigen `DescribeFileSystems`-Operation zurückgegeben wurde (Zeichenfolge). Falls vorhanden, gibt es an, dass die Liste an der Stelle fortgesetzt werden soll, an der der Aufruf, der eine Ausgabe zurückgibt, abgebrochen wurde.

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 128 Zeichen.

Pattern: . +

MaxItems

(Optional) Gibt die maximale Anzahl der Dateisysteme an, die in der Antwort zurückgegeben werden können (Ganzzahl). Diese Zahl wird automatisch auf 100 gesetzt. Die Antwort wird mit 100 Dateisystemen pro Seite paginiert, sofern es mehr als 100 Dateisysteme gibt.

Gültiger Bereich: Mindestwert 1.

Anforderungstext

Der Anforderung besitzt keinen Anforderungstext.

Antwortsyntax

```
HTTP/1.1 200
Content-type: application/json

{
```

```

"FileSystems": [
  {
    "AvailabilityZoneId": "string",
    "AvailabilityZoneName": "string",
    "CreationTime": number,
    "CreationToken": "string",
    "Encrypted": boolean,
    "FileSystemArn": "string",
    "FileSystemId": "string",
    "FileSystemProtection": {
      "ReplicationOverwriteProtection": "string"
    },
    "KmsKeyId": "string",
    "LifeCycleState": "string",
    "Name": "string",
    "NumberOfMountTargets": number,
    "OwnerId": "string",
    "PerformanceMode": "string",
    "ProvisionedThroughputInMibps": number,
    "SizeInBytes": {
      "Timestamp": number,
      "Value": number,
      "ValueInArchive": number,
      "ValueInIA": number,
      "ValueInStandard": number
    },
    "Tags": [
      {
        "Key": "string",
        "Value": "string"
      }
    ],
    "ThroughputMode": "string"
  }
],
"Marker": "string",
"NextMarker": "string"
}

```

Antwortelemente

Wenn die Aktion erfolgreich ist, sendet der Service eine HTTP 200-Antwort zurück.

Die folgenden Daten werden vom Service im JSON-Format zurückgegeben.

FileSystems

Ein Array von Dateisystembeschreibungen.

Typ: Array von [FileSystemDescription](#)-Objekten

Marker

Vorhanden, falls vom Aufrufer in der Anforderung angegeben (Zeichenfolge).

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 128 Zeichen.

Pattern: . +

NextMarker

Vorhanden, wenn es mehr Dateisysteme gibt, als in der Antwort zurückgegeben wurden (Zeichenfolge). Sie können NextMarker in einer nachfolgenden Anforderung verwenden, um die Beschreibungen abzurufen.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 128 Zeichen.

Pattern: . +

Fehler

BadRequest

Wird zurückgegeben, wenn die Anfrage fehlerhaft ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden.

Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene `FileSystemId` Wert im Wert des Anforderers nicht vorhanden ist. AWS-Konto

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 500

Beispiele

Abrufen einer Liste von 10 Dateisystemen

Im folgenden Beispiel wird eine GET-Anfrage an den `file-systems` Endpunkt (`elasticfilesystem.us-west-2.amazonaws.com/2015-02-01/file-systems`) gesendet. Die Anforderung gibt einen `MaxItems`-Abfrageparameter an, um die Anzahl der Dateisystembeschreibungen auf 10 zu begrenzen.

Beispielanforderung

```
GET /2015-02-01/file-systems?MaxItems=10 HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20140622T191208Z
Authorization: <...>
```

Beispielantwort

```
HTTP/1.1 200 OK
x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
Content-Type: application/json
Content-Length: 499
{
  "FileSystems": [
    {
      "OwnerId": "251839141158",
      "CreationToken": "MyFileSystem1",
      "FileSystemId": "fs-01234567",
      "PerformanceMode": "generalPurpose",
      "CreationTime": "1403301078",
      "LifecycleState": "created",
      "Name": "my first file system",
      "NumberOfMountTargets": 1,
      "SizeInBytes": {
        "Timestamp": 1403301078,
        "Value": 29313618372,
        "ValueInArchive": 201156,
        "ValueInIA": 675432,
        "ValueInStandard": 29312741784
      }
    }
  ]
}
```

```
}  
]  
}
```

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen finden Sie im AWS SDKs Folgenden:

- [AWS Befehlszeilenschnittstelle V2](#)
- [AWS SDK for .NET V4](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS SDK für Kotlin](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK for Ruby V3](#)

DescribeLifecycleConfiguration

Gibt das aktuelle LifecycleConfiguration Objekt für das angegebene EFS-Dateisystem zurück. Bei der Lebenszyklusverwaltung wird das LifecycleConfiguration-Objekt verwendet, um zu ermitteln, wann Dateien zwischen Speicherklassen verschoben werden müssen. In einem Dateisystem ohne LifecycleConfiguration-Objekt gibt der Aufruf in der Antwort ein leeres Array zurück.

Diese Operation erfordert Berechtigungen für die Operation `elasticfilesystem:DescribeLifecycleConfiguration`.

Anforderungssyntax

```
GET /2015-02-01/file-systems/FileSystemId/lifecycle-configuration HTTP/1.1
```

URI-Anfrageparameter

Die Anforderung verwendet die folgenden URI-Parameter.

FileSystemId

Die ID des Dateisystems, dessen LifecycleConfiguration-Objekt Sie abrufen möchten (Zeichenfolge).

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Erforderlich: Ja

Anforderungstext

Der Anforderung besitzt keinen Anforderungstext.

Antwortsyntax

```
HTTP/1.1 200
Content-type: application/json
```

```
{
  "LifecyclePolicies": [
    {
      "TransitionToArchive": "string",
      "TransitionToIA": "string",
      "TransitionToPrimaryStorageClass": "string"
    }
  ]
}
```

Antwortelemente

Wenn die Aktion erfolgreich ist, sendet der Service eine HTTP 200-Antwort zurück.

Die folgenden Daten werden vom Service im JSON-Format zurückgegeben.

[LifecyclePolicies](#)

Eine Reihe von Richtlinien für das Lebenszyklusmanagement. EFS unterstützt maximal eine Richtlinie pro Dateisystem.

Typ: Array von [LifecyclePolicy](#)-Objekten

Array-Mitglieder: Maximale Anzahl von 3 Elementen.

Fehler

BadRequest

Wird zurückgegeben, wenn die Anfrage fehlerhaft ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden.

Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine

Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können.

Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene FileSystemId Wert im Wert des Anforderers nicht vorhanden ist AWS-Konto.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden.

Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können.

Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden.

Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können.

Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 500

Beispiele

Abrufen der Lebenszykluskonfiguration für ein Dateisystem

Die folgende Anforderung ruft das LifecycleConfiguration-Objekt für das angegebene Dateisystem ab.

Beispielanforderung

```
GET /2015-02-01/file-systems/fs-01234567/lifecycle-configuration HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20181120T221118Z
Authorization: <...>
```

Beispielantwort

```
HTTP/1.1 200 OK
    x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
    Content-Type: application/json
    Content-Length: 86
{
  "LifecyclePolicies": [
    {
      "TransitionToArchive": "AFTER_270_DAYS"
    },
    {
      "TransitionToIA": "AFTER_14_DAYS"
    },
    {
      "TransitionToPrimaryStorageClass": "AFTER_1_ACCESS"
    }
  ]
}
```

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen finden Sie im AWS SDKs Folgenden:

- [AWS Befehlszeilenschnittstelle V2](#)
- [AWS SDK for .NET V4](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS SDK für Kotlin](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK for Ruby V3](#)

DescribeMountTargets

Gibt die Beschreibungen aller aktuellen Mountziele oder eines bestimmten Mountziels für ein Dateisystem zurück. Wenn alle aktuellen Mountingziele angefordert werden, ist die Reihenfolge der Mountingziele, die in der Antwort zurückgegeben werden, nicht angegeben.

Für diesen Vorgang sind Berechtigungen für die Aktion `elasticfilesystem:DescribeMountTargets` erforderlich, entweder für die Dateisystem-ID, die Sie in `FileSystemId` angeben, oder für das Dateisystem des Mountingziels, das Sie in `MountTargetId` angeben.

Der entsprechende CLI-Befehl lautet `describe-mount-targets`.

Anforderungssyntax

```
GET /2015-02-01/mount-targets?  
AccessPointId=AccessPointId&FileSystemId=FileSystemId&Marker=Marker&MaxItems=MaxItems&MountTargetId=MountTargetId  
HTTP/1.1
```

URI-Anfrageparameter

Die Anforderung verwendet die folgenden URI-Parameter.

AccessPointId

(Optional) Die ID des Zugangspunkts, dessen Mountingziele Sie auflisten möchten. Sie muss in der Anforderung enthalten sein, falls keine `FileSystemId` oder `MountTargetId` in der Anforderung enthalten ist. Akzeptiert entweder eine Zugangspunkt-ID oder einen ARN als Eingabe.

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:access-point/fsap-[0-9a-f]{8,40}|fsap-[0-9a-f]{8,40})$`

FileSystemId

(Optional) ID des Dateisystems, dessen Mountingziele Sie auflisten möchten (Zeichenfolge). Sie muss in der Anforderung enthalten sein, falls keine `AccessPointId` oder `MountTargetId` in der Anforderung enthalten ist. Akzeptiert entweder eine Dateisystem-ID oder einen ARN als Eingabe.

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Marker

(Optional) Opaque-Paginierungstoken, das von einer vorherigen `DescribeMountTargets`-Operation zurückgegeben wurde (Zeichenfolge). Falls vorhanden, gibt es an, dass die Liste an der Stelle fortgesetzt werden soll, an der der vorherige Aufruf, der eine Ausgabe zurückgibt, abgebrochen wurde.

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 128 Zeichen.

Pattern: `.+`

MaxItems

(Optional) Die maximale Anzahl der Mountingziele, die in der Antwort zurückgegeben werden können. Derzeit wird diese Anzahl automatisch auf 10 gesetzt, und andere Werte werden ignoriert. Die Antwort wird mit 100 Mountingzielen pro Seite paginiert, sofern es mehr als 100 Mountingziele gibt.

Gültiger Bereich: Mindestwert 1.

MountTargetId

(Optional) ID des Mountingziels, das beschrieben werden soll (Zeichenfolge). Sie muss in der Anforderung enthalten sein, falls keine `FileSystemId` in der Anforderung enthalten ist. Akzeptiert entweder eine Mountingziel-ID oder einen ARN als Eingabe.

Längenbeschränkungen: Mindestlänge von 13. Maximale Länge beträgt 45 Zeichen.

Pattern: `^fsmt-[0-9a-f]{8,40}$`

Anforderungstext

Der Anforderung besitzt keinen Anforderungstext.

Antwortsyntax

HTTP/1.1 200

Content-type: application/json

```
{
  "Marker": "string",
  "MountTargets": [
    {
      "AvailabilityZoneId": "string",
      "AvailabilityZoneName": "string",
      "FileSystemId": "string",
      "IpAddress": "string",
      "Ipv6Address": "string",
      "LifecycleState": "string",
      "MountTargetId": "string",
      "NetworkInterfaceId": "string",
      "OwnerId": "string",
      "SubnetId": "string",
      "VpcId": "string"
    }
  ],
  "NextMarker": "string"
}
```

Antwortelemente

Wenn die Aktion erfolgreich ist, sendet der Service eine HTTP 200-Antwort zurück.

Die folgenden Daten werden vom Service im JSON-Format zurückgegeben.

Marker

Wenn die Anforderung den `Marker` enthält, wird dieser Wert in der Antwort in diesem Feld zurückgegeben.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 128 Zeichen.

Pattern: `.+`

MountTargets

Gibt die Mountingziele des Dateisystems als Array von `MountTargetDescription`-Objekten zurück.

Typ: Array von [MountTargetDescription](#)-Objekten

[NextMarker](#)

Wenn ein Wert vorhanden ist, sind weitere Mountingziele verfügbar, die zurückgegeben werden. In einer nachfolgenden Anforderung können Sie `Marker` angeben, um den nächsten Satz von Mountingzielen abzurufen.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 128 Zeichen.

Pattern: `.+`

Fehler

`AccessPointNotFound`

Wird zurückgegeben, wenn der angegebene `AccessPointId` Wert im Wert des Anforderers nicht vorhanden ist AWS-Konto.

`ErrorCode`

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

`Message`

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

`BadRequest`

Wird zurückgegeben, wenn die Anfrage fehlerhaft ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene FileSystemId Wert im Wert des Anforderers nicht vorhanden ist. AWS-Konto

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 500

MountTargetNotFound

Wird zurückgegeben, wenn kein Mountingziel mit der angegebenen ID im AWS-Konto des Aufrufers gefunden wurde.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

Beispiele

Abrufen von Beschreibungen von Mountingzielen, die für ein Dateisystem erstellt wurden

Die folgende Anforderung ruft Beschreibungen von Mountingzielen ab, die für das angegebene Dateisystem erstellt wurden.

Beispielanforderung

```
GET /2015-02-01/mount-targets?FileSystemId=fs-01234567 HTTP/1.1
```

```
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20140622T191252Z
Authorization: <...>
```

Beispielantwort

```
HTTP/1.1 200 OK
x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
Content-Type: application/json
Content-Length: 357

{
  "MountTargets":[
    {
      "OwnerId":"251839141158",
      "MountTargetId":"fsmt-01234567",
      "FileSystemId":"fs-01234567",
      "SubnetId":"subnet-01234567",
      "LifeCycleState":"added",
      "IpAddress":"10.0.2.42",
      "NetworkInterfaceId":"eni-1bcb7772"
    }
  ]
}
```

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen finden Sie im AWS SDKs Folgenden:

- [AWS Befehlszeilenschnittstelle V2](#)
- [AWS SDK for .NET V4](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS SDK für Kotlin](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK für Python](#)

- [AWS SDK for Ruby V3](#)

DescribeMountTargetSecurityGroups

Gibt die Sicherheitsgruppen zurück, die derzeit für ein Mountziel gültig sind. Sie setzt voraus, dass die Netzwerkschnittstelle des Mountingziels erstellt wurde und der Lebenszyklusstatus des Mountingziels nicht `deleted` lautet.

Diese Operation erfordert außerdem Berechtigungen für die folgenden Aktionen:

- Aktion `elasticfilesystem:DescribeMountTargetSecurityGroups` im Dateisystem des Mountingziels.
- Aktion `ec2:DescribeNetworkInterfaceAttribute` in der Netzwerkschnittstelle des Mountingziels.

Anforderungssyntax

```
GET /2015-02-01/mount-targets/MountTargetId/security-groups HTTP/1.1
```

URI-Anfrageparameter

Die Anforderung verwendet die folgenden URI-Parameter.

MountTargetId

Die ID des Mountingziels, dessen Sicherheitsgruppen Sie abrufen möchten.

Längenbeschränkungen: Mindestlänge von 13. Maximale Länge beträgt 45 Zeichen.

Pattern: `^fsmt-[0-9a-f]{8,40}$`

Erforderlich: Ja

Anforderungstext

Der Anforderung besitzt keinen Anforderungstext.

Antwortsyntax

```
HTTP/1.1 200  
Content-type: application/json
```

```
{  
  "SecurityGroups": [ "string" ]  
}
```

Antwortelemente

Wenn die Aktion erfolgreich ist, sendet der Service eine HTTP 200-Antwort zurück.

Die folgenden Daten werden vom Service im JSON-Format zurückgegeben.

[SecurityGroups](#)

Ein Array von Sicherheitsgruppen.

Typ: Zeichenfolgen-Array

Array-Mitglieder: Maximale Anzahl von 100 Elementen.

Längenbeschränkungen: Mindestlänge von 11. Maximale Länge von 43.

Pattern: `^sg-[0-9a-f]{8,40}`

Fehler

BadRequest

Wird zurückgegeben, wenn die Anfrage fehlerhaft ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden.

Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können.

Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

IncorrectMountTargetState

Wird zurückgegeben, wenn das Mountingziel nicht den richtigen Status für die Operation aufweist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 409

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 500

MountTargetNotFound

Wird zurückgegeben, wenn kein Mountingziel mit der angegebenen ID im AWS-Konto des Aufrufers gefunden wurde.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

Beispiele

Rufen Sie Sicherheitsgruppen ab, die für ein Dateisystem aktiv sind

Im folgenden Beispiel werden die Sicherheitsgruppen, die für die einem Mountingziel zugeordnete Netzwerkschnittstelle gelten, abgerufen.

Beispielanforderung

```
GET /2015-02-01/mount-targets/fsmt-9a13661e/security-groups HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20140620T223513Z
Authorization: <...>
```

Beispielantwort

```
HTTP/1.1 200 OK
x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
Content-Length: 57

{
  "SecurityGroups" : [
    "sg-188d9f74"
  ]
}
```

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen AWS SDKs finden Sie im Folgenden:

- [AWS Befehlszeilenschnittstelle V2](#)
- [AWS SDK for .NET V4](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS SDK für Kotlin](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK for Ruby V3](#)

DescribeReplicationConfigurations

Ruft die Replikationskonfiguration für ein bestimmtes Dateisystem ab. Wenn kein Dateisystem angegeben ist, werden alle Replikationskonfigurationen für das AWS-Konto in AWS-Region an abgerufen.

Anforderungssyntax

```
GET /2015-02-01/file-systems/replication-configurations?  
FileSystemId=FileSystemId&MaxResults=MaxResults&NextToken=NextToken HTTP/1.1
```

URI-Anfrageparameter

Die Anforderung verwendet die folgenden URI-Parameter.

FileSystemId

Sie können die Replikationskonfiguration für ein bestimmtes Dateisystem abrufen, indem Sie dessen Dateisystem-ID angeben. Bei der konto- und regionsübergreifenden Replikation kann ein Konto nur die Replikationskonfiguration für ein Dateisystem in seiner eigenen Region beschreiben.

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

MaxResults

(Optional) Um die Anzahl der in einer Antwort zurückgegebenen Objekte zu begrenzen, können Sie den Parameter `MaxItems` angeben. Der Standardwert lautet 100.

Gültiger Bereich: Mindestwert 1.

NextToken

`NextToken` ist vorhanden, wenn die Antwort paginiert ist. Sie können `NextToken` in einer nachfolgenden Anfrage verwenden, um die nächste Ausgabeseite abzurufen.

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 128 Zeichen.

Pattern: . +

Anforderungstext

Der Anforderung besitzt keinen Anforderungstext.

Antwortsyntax

```
HTTP/1.1 200
Content-type: application/json

{
  "NextToken": "string",
  "Replications": [
    {
      "CreationTime": number,
      "Destinations": [
        {
          "FileSystemId": "string",
          "LastReplicatedTimestamp": number,
          "OwnerId": "string",
          "Region": "string",
          "RoleArn": "string",
          "Status": "string",
          "StatusMessage": "string"
        }
      ],
      "OriginalSourceFileSystemArn": "string",
      "SourceFileSystemArn": "string",
      "SourceFileSystemId": "string",
      "SourceFileSystemOwnerId": "string",
      "SourceFileSystemRegion": "string"
    }
  ]
}
```

Antwortelemente

Wenn die Aktion erfolgreich ist, sendet der Service eine HTTP 200-Antwort zurück.

Die folgenden Daten werden vom Service im JSON-Format zurückgegeben.

NextToken

Sie können NextToken aus der vorherigen Antwort in einer nachfolgenden Anfrage verwenden, um die zusätzlichen Beschreibungen abzurufen.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 128 Zeichen.

Pattern: .+

Replications

Die Sammlung von Replikationskonfigurationen, die zurückgegeben werden.

Typ: Array von [ReplicationConfigurationDescription](#)-Objekten

Fehler

BadRequest

Wird zurückgegeben, wenn die Anfrage fehlerhaft ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene FileSystemId Wert im Wert des Anforderers nicht vorhanden ist. AWS-Konto

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 500

ReplicationNotFound

Wird zurückgegeben, wenn das angegebene Dateisystem keine Replikationskonfiguration aufweist.

ErrorCode

ReplicationNotFound

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

ValidationException

Wird zurückgegeben, wenn der AWS Backup Dienst in dem Land, AWS-Region in dem die Anfrage gestellt wurde, nicht verfügbar ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen AWS SDKs finden Sie im Folgenden:

- [AWS Befehlszeilenschnittstelle V2](#)
- [AWS SDK for .NET V4](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)

- [AWS SDK for Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS SDK für Kotlin](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK for Ruby V3](#)

DescribeTags

Note

VERALTET – Die Aktion `DescribeTags` ist veraltet und wird nicht mehr unterstützt. Verwenden Sie die API-Aktion `ListTagsForResource`, um Tags anzuzeigen, die mit EFS-Ressourcen verknüpft sind.

Gibt die einem Dateisystem zugeordneten Tags zurück. Die Reihenfolge der Tags, die in der Antwort auf einen `DescribeTags`- Aufruf zurückgegeben werden, und die Reihenfolge der Tags, die in den Antworten einer Iteration mit mehreren Aufrufen zurückgegeben werden (bei Verwendung der Paginierung), ist nicht angegeben.

Diese Operation erfordert Berechtigungen für die Aktion `elasticfilesystem:DescribeTags`.

Anforderungssyntax

```
GET /2015-02-01/tags/FileSystemId?Marker=Marker&MaxItems=MaxItems HTTP/1.1
```

URI-Anfrageparameter

Die Anforderung verwendet die folgenden URI-Parameter.

FileSystemId

Die ID des Dateisystems, dessen Tag-Set Sie abrufen möchten.

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Erforderlich: Ja

Marker

(Optional) Ein Opaque-Paginierungstoken, das von einer vorherigen `DescribeTags`-Operation zurückgegeben wurde (Zeichenfolge). Falls vorhanden, gibt es an, dass die Liste an der Stelle fortgesetzt werden soll, an der der vorherige Aufruf abgebrochen wurde.

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 128 Zeichen.

Pattern: .+

[MaxItems](#)

(Optional) Die maximale Anzahl der Dateisystem-Tags, die in der Antwort zurückgegeben werden können. Derzeit wird diese Anzahl automatisch auf 100 gesetzt, und andere Werte werden ignoriert. Die Antwort wird mit 100 Tags pro Seite paginiert, sofern es mehr als 100 Tags gibt.

Gültiger Bereich: Mindestwert 1.

Anforderungstext

Der Anforderung besitzt keinen Anforderungstext.

Antwortsyntax

```
HTTP/1.1 200
Content-type: application/json

{
  "Marker": "string",
  "NextMarker": "string",
  "Tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ]
}
```

Antwortelemente

Wenn die Aktion erfolgreich ist, sendet der Service eine HTTP 200-Antwort zurück.

Die folgenden Daten werden vom Service im JSON-Format zurückgegeben.

[Marker](#)

Wenn die Anfrage einen Marker enthält, wird dieser Wert in der Antwort in diesem Feld zurückgegeben.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 128 Zeichen.

Pattern: . +

[NextMarker](#)

Wenn ein Wert vorhanden ist, sind weitere Tags verfügbar, die zurückgegeben werden. In einer nachfolgenden Anfrage können Sie den Wert von `NextMarker` als Wert des Parameters `Marker` in der nächsten Anfrage angeben, um den nächsten Satz an Tags abzurufen.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 128 Zeichen.

Pattern: . +

[Tags](#)

Gibt Tags, die dem Dateisystem zugeordnet sind, als ein Array von Tag-Objekten zurück.

Typ: Array von [Tag](#)-Objekten

Fehler

BadRequest

Wird zurückgegeben, wenn die Anfrage fehlerhaft ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für Menschen vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können.

Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene FileSystemId Wert im Wert des Anforderers nicht vorhanden ist AWS-Konto.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 500

Beispiele

Abrufen aller einem Dateisystem zugeordneten Tags

Die folgende Anforderung ruft Tags (Schlüssel-Wert-Paare) ab, die dem angegebenen Dateisystem zugeordnet sind.

Beispielanforderung

```
GET /2015-02-01/tags/fs-01234567/ HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20140620T215404Z
Authorization: <...>
```

Beispielantwort

```
HTTP/1.1 200 OK
x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
Content-Type: application/json
Content-Length: 288

{
  "Tags": [
    {
      "Key": "Name",
      "Value": "my first file system"
    },
    {
      "Key": "Fleet",
      "Value": "Development"
    },
    {
      "Key": "Developer",
      "Value": "Alice"
    }
  ]
}
```

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen finden Sie im AWS SDKs Folgenden:

- [AWS Befehlszeilenschnittstelle V2](#)
- [AWS SDK for .NET V4](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS SDK für Kotlin](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK for Ruby V3](#)

ListTagsForResource

Listet alle Tags für eine EFS-Ressource der obersten Ebene auf. Sie müssen die ID der Ressource angeben, für die Sie die Tags abrufen wollen.

Diese Operation erfordert Berechtigungen für die Aktion `elasticfilesystem:DescribeAccessPoints`.

Anforderungssyntax

```
GET /2015-02-01/resource-tags/ResourceId?MaxResults=MaxResults&NextToken=NextToken
HTTP/1.1
```

URI-Anfrageparameter

Die Anforderung verwendet die folgenden URI-Parameter.

[MaxResults](#)

(Optional) Gibt die maximale Anzahl der Tag-Objekte an, die in der Antwort zurückgegeben werden können. Der Standardwert lautet 100.

Gültiger Bereich: Mindestwert 1.

[NextToken](#)

(Optional) Sie können `NextToken` in einer nachfolgenden Anfrage verwenden, um die nächste Seite mit Zugangspunktbeschreibungen abzurufen, wenn die Antwortnutzlast paginiert wurde.

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 128 Zeichen.

Pattern: `.+`

[ResourceId](#)

Gibt die EFS-Ressource an, für die Sie Tags abrufen möchten. Mit diesem API-Endpunkt können Sie Tags für EFS-Dateisysteme und Zugangspunkte abrufen.

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:(access-point/fsap|file-system/fs)-[0-9a-f]{8,40}|fs(ap)?-[0-9a-f]{8,40})$`

Erforderlich: Ja

Anforderungstext

Der Anforderung besitzt keinen Anforderungstext.

Antwortsyntax

```
HTTP/1.1 200
Content-type: application/json

{
  "NextToken": "string",
  "Tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ]
}
```

Antwortelemente

Wenn die Aktion erfolgreich ist, sendet der Service eine HTTP 200-Antwort zurück.

Die folgenden Daten werden vom Service im JSON-Format zurückgegeben.

[NextToken](#)

NextToken ist vorhanden, wenn die Antwort-Payload paginiert ist. Sie können NextToken in einer nachfolgenden Anfrage verwenden, um die nächste Zugangspunktseite abzurufen.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 128 Zeichen.

Pattern: . +

[Tags](#)

Ein Array der Tags für die angegebene EFS-Ressource.

Typ: Array von [Tag](#)-Objekten

Fehler

AccessPointNotFound

Wird zurückgegeben, wenn der angegebene `AccessPointId` Wert im Wert des Anforderers nicht vorhanden ist AWS-Konto.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

BadRequest

Wird zurückgegeben, wenn die Anfrage fehlerhaft ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene `FileSystemId` Wert im Wert des Anforderers nicht vorhanden ist. AWS-Konto

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 500

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen finden Sie im AWS SDKs Folgenden:

- [AWS Befehlszeilenschnittstelle V2](#)
- [AWS SDK for .NET V4](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS SDK für Kotlin](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK for Ruby V3](#)

ModifyMountTargetSecurityGroups

Ändert den Satz der Sicherheitsgruppen, die für ein Mountingziel gültig sind.

Wenn Sie ein Mountingziel erstellen, wird in Amazon EFS auch eine neue Netzwerkschnittstelle erstellt. Weitere Informationen finden Sie unter [CreateMountTarget](#). Durch diese Operation werden die Sicherheitsgruppen, die für die einem Mountingziel zugeordnete Netzwerkschnittstelle gelten, durch die in der Anforderung angegebenen SecurityGroups ersetzt. Sie setzt voraus, dass die Netzwerkschnittstelle des Mountingziels erstellt wurde und der Lebenszyklusstatus des Mountingziels nicht `deleted` lautet.

Die Operation erfordert Berechtigungen für die folgende Dateisystemaktion:

- Aktion `elasticfilesystem:ModifyMountTargetSecurityGroups` im Dateisystem des Mountingziels.
- Aktion `ec2:ModifyNetworkInterfaceAttribute` in der Netzwerkschnittstelle des Mountingziels.

Anforderungssyntax

```
PUT /2015-02-01/mount-targets/MountTargetId/security-groups HTTP/1.1
Content-type: application/json

{
  "SecurityGroups": [ "string" ]
}
```

URI-Anfrageparameter

Die Anforderung verwendet die folgenden URI-Parameter.

[MountTargetId](#)

Die ID des Mountingziels, dessen Sicherheitsgruppen Sie ändern möchten.

Längenbeschränkungen: Mindestlänge von 13. Maximale Länge beträgt 45 Zeichen.

Pattern: `^fsmt-[0-9a-f]{8,40}$`

Erforderlich: Ja

Anforderungstext

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

SecurityGroups

Eine Reihe von VPC-Sicherheitsgruppen IDs.

Typ: Zeichenfolgen-Array

Array-Mitglieder: Maximale Anzahl von 100 Elementen.

Längenbeschränkungen: Mindestlänge von 11. Maximale Länge von 43.

Pattern: `^sg-[0-9a-f]{8,40}`

Erforderlich: Nein

Antwortsyntax

```
HTTP/1.1 204
```

Antwortelemente

Wenn die Aktion erfolgreich ist, gibt der Dienst eine HTTP-204-Antwort mit leerem HTTP-Textinhalt zurück.

Fehler

BadRequest

Wird zurückgegeben, wenn die Anfrage fehlerhaft ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden.

Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

IncorrectMountTargetState

Wird zurückgegeben, wenn das Mountingziel nicht den richtigen Status für die Operation aufweist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 409

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 500

MountTargetNotFound

Wird zurückgegeben, wenn kein Mountingziel mit der angegebenen ID im AWS-Konto des Aufrufers gefunden wurde.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

SecurityGroupLimitExceeded

Wird zurückgegeben, wenn die in der Anfrage `SecurityGroups` angegebene Anzahl von Objekten das Limit überschreitet, das auf dem Kontingent basiert. Löschen Sie entweder einige Sicherheitsgruppen oder fordern Sie eine Erhöhung des Kontokontingents an. Weitere Informationen finden Sie unter [Amazon VPC-Kontingente](#) im Amazon VPC-Benutzerhandbuch (siehe Tabelle mit Sicherheitsgruppen).

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

SecurityGroupNotFound

Wird zurückgegeben, wenn eine der angegebenen Sicherheitsgruppen nicht in der Virtual Private Cloud (VPC) des Subnetzes vorhanden ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

Beispiele

Ersetzen der Sicherheitsgruppen eines Mountingziels

Im folgenden Beispiel werden die Sicherheitsgruppen, die für die einem Mountingziel zugeordnete Netzwerkschnittstelle gelten, ersetzt.

Beispielanforderung

```
PUT /2015-02-01/mount-targets/fsmt-9a13661e/security-groups HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20140620T223446Z
Authorization: <...>
Content-Type: application/json
Content-Length: 57

{
  "SecurityGroups" : [
    "sg-188d9f74"
  ]
}
```

```
}
```

Beispielantwort

```
HTTP/1.1 204 No Content  
x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
```

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen finden Sie im Folgenden AWS SDKs:

- [AWS Befehlszeilenschnittstelle V2](#)
- [AWS SDK for .NET V4](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS SDK für Kotlin](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK for Ruby V3](#)

PutAccountPreferences

Verwenden Sie diesen Vorgang, um die Kontopräferenz in der aktuellen Version so festzulegen AWS-Region , dass IDs für das neue EFS-Dateisystem eine lange Ressource mit 17 Zeichen (63 Bit) oder eine kurze Ressource mit 8 Zeichen (32 Bit) verwendet und Zielressourcen bereitgestellt werden. Alle vorhandenen Ressourcen IDs sind von den Änderungen, die Sie vornehmen, nicht betroffen. Sie können die ID-Präferenz während des Anmeldezeitraums festlegen, wenn EFS auf Long Resource IDs umsteigt. Weitere Informationen finden Sie unter [Amazon EFS-Ressourcen verwalten IDs](#).

Note

Seit Oktober 2021 wird bei dem Versuch, die Ressourcen-ID in der Voreinstellung für das Konto auf das kurze 8-stellige Format zu ändern, eine Fehlermeldung ausgegeben. Wenden Sie sich an den AWS Support, wenn Sie eine Fehlermeldung erhalten und die Abkürzung IDs für Dateisystem und Mount-Zielressourcen verwenden müssen.

Anforderungssyntax

```
PUT /2015-02-01/account-preferences HTTP/1.1
Content-type: application/json
```

```
{
  "ResourceIdType": "string"
}
```

URI-Anfrageparameter

Die Anforderung verwendet keine URI-Parameter.

Anforderungstext

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

[ResourceIdType](#)

Gibt die EFS-Ressourcen-ID-Einstellung an AWS-Konto, die für den Benutzer in der aktuellen Version AWS-Region entweder LONG_ID (17 Zeichen) oder SHORT_ID (8 Zeichen) festgelegt werden soll.

 Note

Seit Oktober 2021 wird eine Fehlermeldung ausgegeben, wenn in der Voreinstellung für das Konto `SHORT_ID` ausgewählt wird. Wenden Sie sich an den AWS Support, wenn Sie eine Fehlermeldung erhalten und die Abkürzung IDs für Dateisystem und Mount-Zielressourcen verwenden müssen.

Typ: Zeichenfolge

Zulässige Werte: `LONG_ID` | `SHORT_ID`

Erforderlich: Ja

Antwortsyntax

```
HTTP/1.1 200
Content-type: application/json

{
  "ResourceIdPreference": {
    "ResourceIdType": "string",
    "Resources": [ "string" ]
  }
}
```

Antwortelemente

Wenn die Aktion erfolgreich ist, sendet der Service eine HTTP 200-Antwort zurück.

Die folgenden Daten werden vom Service im JSON-Format zurückgegeben.

ResourceIdPreference

Beschreibt den Ressourcentyp und seine ID-Präferenz für den AWS-Konto Benutzer in der aktuellen Version AWS-Region.

Typ: ResourceIdPreference Objekt

Fehler

BadRequest

Wird zurückgegeben, wenn die Anfrage fehlerhaft ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 500

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen AWS SDKs finden Sie im Folgenden:

- [AWS Befehlszeilenschnittstelle V2](#)
- [AWS SDK for .NET V4](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS SDK für Kotlin](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK for Ruby V3](#)

PutBackupPolicy

Aktualisiert die Backup-Richtlinie des Dateisystems. Mit dieser Aktion können Sie automatische Backups des Dateisystems starten oder beenden.

Anforderungssyntax

```
PUT /2015-02-01/file-systems/FileSystemId/backup-policy HTTP/1.1
Content-type: application/json

{
  "BackupPolicy": {
    "Status": "string"
  }
}
```

URI-Anfrageparameter

Die Anforderung verwendet die folgenden URI-Parameter.

FileSystemId

Gibt an, für welches EFS-Dateisystem die Backup-Richtlinie aktualisiert werden soll.

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Erforderlich: Ja

Anforderungstext

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

BackupPolicy

Die in der PutBackupPolicy-Anforderung enthaltene Backup-Richtlinie.

Typ: BackupPolicy Objekt

Erforderlich: Ja

Antwortsyntax

```
HTTP/1.1 200
Content-type: application/json

{
  "BackupPolicy": {
    "Status": "string"
  }
}
```

Antwortelemente

Wenn die Aktion erfolgreich ist, sendet der Service eine HTTP 200-Antwort zurück.

Die folgenden Daten werden vom Service im JSON-Format zurückgegeben.

[BackupPolicy](#)

Beschreibt die Backup-Richtlinie des Dateisystems und gibt an, ob automatische Backups aktiviert oder deaktiviert sind.

Typ: [BackupPolicy](#) Objekt

Fehler

BadRequest

Wird zurückgegeben, wenn die Anfrage fehlerhaft ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden.

Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

`FileSystemNotFound`

Wird zurückgegeben, wenn der angegebene `FileSystemId` Wert im Wert des Anforderers nicht vorhanden ist AWS-Konto.

`ErrorCode`

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

`Message`

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

`IncorrectFileSystemLifecycleState`

Wird zurückgegeben, wenn der Lebenszyklusstatus des Dateisystems nicht „verfügbar“ ist.

`ErrorCode`

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

`Message`

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können.

Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 409

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden.

Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können.

Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 500

ValidationException

Wird zurückgegeben, wenn der AWS Backup Dienst in dem Land, AWS-Region in dem die Anfrage gestellt wurde, nicht verfügbar ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden.

Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können.

Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen AWS SDKs finden Sie im Folgenden:

- [AWS Befehlszeilenschnittstelle V2](#)
- [AWS SDK for .NET V4](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS SDK für Kotlin](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK for Ruby V3](#)

PutFileSystemPolicy

Wendet eine `FileSystemPolicy` auf ein Amazon-EFS-Dateisystem an. Eine Dateisystemrichtlinie ist eine auf IAM-Ressourcen basierende Richtlinie und kann mehrere Richtlinianweisungen enthalten. Ein Dateisystem hat immer genau eine Dateisystemrichtlinie. Dabei kann es sich um die Standardrichtlinie oder um eine explizite Richtlinie handeln, die mithilfe dieser API-Operation festgelegt oder aktualisiert wurde. Die Zeichenbeschränkung von EFS-Dateisystemrichtlinien liegt bei 20.000. Wenn eine explizite Richtlinie festgelegt wird, hat diese Vorrang vor der Standardrichtlinie. Weitere Informationen zur Standard-Dateisystemrichtlinie finden Sie unter [EFS-Standarddateisystemrichtlinie](#).

Note

Die Zeichenbeschränkung von EFS-Dateisystemrichtlinien liegt bei 20.000.

Diese Operation erfordert Berechtigungen für die Aktion `elasticfilesystem:PutFileSystemPolicy`.

Anforderungssyntax

```
PUT /2015-02-01/file-systems/FileSystemId/policy HTTP/1.1
Content-type: application/json

{
  "BypassPolicyLockoutSafetyCheck": boolean,
  "Policy": "string"
}
```

URI-Anfrageparameter

Die Anforderung verwendet die folgenden URI-Parameter.

[FileSystemId](#)

Die ID des EFS-Dateisystems, für das Sie die `FileSystemPolicy` erstellen oder aktualisieren möchten.

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Erforderlich: Ja

Anforderungstext

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

[BypassPolicyLockoutSafetyCheck](#)

(Optional) Ein boolescher Wert, der angibt, ob die `FileSystemPolicy`-Sperr Sicherheitsprüfung umgangen werden soll oder nicht. Die Sperr Sicherheitsprüfung bestimmt, ob die Richtlinie in der Anforderung den IAM-Prinzipal, der die Anforderung stellt, sperrt oder daran hindert, zukünftige `PutFileSystemPolicy`-Anforderungen an dieses Dateisystem zu stellen. Setzen Sie `BypassPolicyLockoutSafetyCheck` nur dann auf `True`, wenn Sie verhindern möchten, dass der IAM-Prinzipal, der die Anforderung stellt, nachfolgende `PutFileSystemPolicy`-Anforderungen an dieses Dateisystem stellt. Der Standardwert ist `False`.

Typ: Boolesch

Erforderlich: Nein

[Policy](#)

Die `FileSystemPolicy`, die Sie erstellen. Akzeptiert eine Richtliniendefinition im JSON-Format. Die Zeichenbeschränkung von EFS-Dateisystemrichtlinien liegt bei 20.000. Weitere Informationen zu den Elementen, aus denen sich eine Dateisystemrichtlinie zusammensetzt, finden Sie unter [Ressourcenbasierte Richtlinien in Amazon EFS](#).

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge von 20 000.

Pattern: `[\s\S]+`

Erforderlich: Ja

Antwortsyntax

```
HTTP/1.1 200
Content-type: application/json
```

```
{  
  "FileSystemId": "string",  
  "Policy": "string"  
}
```

Antwortelemente

Wenn die Aktion erfolgreich ist, sendet der Service eine HTTP 200-Antwort zurück.

Die folgenden Daten werden vom Service im JSON-Format zurückgegeben.

FileSystemId

Gibt das EFS-Dateisystem an, für das die FileSystemPolicy gilt.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Policy

Die FileSystemPolicy im JSON-Format für das EFS-Dateisystem.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge von 20 000.

Pattern: `[\s\S]+`

Fehler

BadRequest

Wird zurückgegeben, wenn die Anfrage fehlerhaft ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene `FileSystemId` Wert im Wert des Anforderers nicht vorhanden ist. AWS-Konto

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

IncorrectFileSystemLifeCycleState

Wird zurückgegeben, wenn der Lebenszyklusstatus des Dateisystems nicht „verfügbar“ ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden.

Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 409

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 500

InvalidPolicyException

Wird zurückgegeben, wenn `FileSystemPolicy` falsch formatiert ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter. Wird zurückgegeben, wenn bei der Sicherheitsüberprüfung der Richtlinie ein Fehler aufgetreten ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können.

Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

Beispiele

Erstellen Sie ein EFS FileSystemPolicy

Die folgende Anforderung erstellt eine FileSystemPolicy, die es allen AWS Prinzipalen ermöglicht, das angegebene EFS-Dateisystem mit Lese- und Schreibberechtigungen zu mounten.

Beispielanforderung

```
PUT /2015-02-01/file-systems/fs-01234567/file-system-policy HTTP/1.1
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticfilesystem:ClientMount",
        "elasticfilesystem:ClientWrite"
      ],
      "Principal": {
        "AWS": ["*"]
      },
    }
  ]
}
```

Beispielantwort

```
{
  "Version": "2012-10-17",
  "Id": "1",
  "Statement": [
    {
      "Sid": "efs-statement-abcdef01-1111-bbbb-2222-111122224444",
      "Effect": "Allow",
      "Action": [
        "elasticfilesystem:ClientMount",

```

```
        "elasticfilesystem:ClientWrite"
      ],
      "Principal": {
        "AWS": ["*"]
      },
      "Resource": "arn:aws:elasticfilesystem:us-east-1:1111222233334444:file-
system/fs-01234567"
    }
  ]
}
```

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen finden Sie im AWS SDKs Folgenden:

- [AWS Befehlszeilenschnittstelle V2](#)
- [AWS SDK for .NET V4](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS SDK für Kotlin](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK for Ruby V3](#)

PutLifecycleConfiguration

Verwenden Sie diese Aktion, um den Speicher für Ihr Dateisystem zu verwalten. Eine LifecycleConfiguration besteht aus einem oder mehreren LifecyclePolicy-Objekten, die Folgendes definieren:

- **TransitionToIA**— Wann müssen Dateien im Dateisystem vom primären Speicher (Standard-Speicherklasse) in den IA-Speicher (Infrequent Access) verschoben werden?
- **TransitionToArchive**— Wann werden Dateien im Dateisystem aus ihrer aktuellen Speicherklasse (entweder IA oder Standardspeicher) in den Archivspeicher verschoben.

Dateisysteme können nicht in den Archivspeicher übergehen, bevor sie in den IA-Speicher übergegangen sind. Daher TransitionToArchive muss entweder nicht gesetzt werden oder muss später als TransitionTo IA sein.

Note

Die Speicherklasse Archive ist nur für Dateisysteme verfügbar, die den Elastic Throughput-Modus und den Allzweck-Performance-Modus verwenden.

- **TransitionToPrimaryStorageClass**— Gibt an, ob Dateien im Dateisystem zurück in den Primärspeicher (Standard-Speicherklasse) verschoben werden sollen, nachdem auf sie im IA- oder Archivspeicher zugegriffen wurde.

Weitere Informationen finden Sie unter [Verwalten des Dateisystemspeichers](#).

Jedes Amazon-EFS-Dateisystem unterstützt eine Lebenszykluskonfiguration, die für alle Dateien im Dateisystem gilt. Wenn ein LifecycleConfiguration-Objekt für das angegebene Dateisystem bereits existiert, ändert ein PutLifecycleConfiguration-Aufruf die bestehende Konfiguration. Ein PutLifecycleConfiguration-Aufruf mit einem leeren LifecyclePolicies-Array im Anfragekörper löscht alle vorhandenen LifecycleConfiguration. Geben Sie in der Anfrage Folgendes an:

- Die ID für das Dateisystem, für das Sie das Lifecycle Management aktivieren, deaktivieren oder ändern.

- Ein LifecyclePolicies-Array von LifecyclePolicy-Objekten, die festlegen, wann Dateien in den IA-Speicher, in den Archivspeicher und zurück in den Primärspeicher verschoben werden sollen.

Note

Amazon EFS erfordert, dass jedes LifecyclePolicy-Objekt nur einen einzigen Übergang hat, sodass das LifecyclePolicies-Array mit separaten LifecyclePolicy-Objekten strukturiert werden muss. Weitere Informationen finden Sie in den Beispielanforderungen im folgenden Abschnitt.

Diese Operation erfordert Berechtigungen für die Operation `elasticfilesystem:PutLifecycleConfiguration`.

Um ein LifecycleConfiguration Objekt auf ein verschlüsseltes Dateisystem anzuwenden, benötigen Sie dieselben AWS Key Management Service Berechtigungen wie bei der Erstellung des verschlüsselten Dateisystems.

Anforderungssyntax

```
PUT /2015-02-01/file-systems/FileSystemId/lifecycle-configuration HTTP/1.1
Content-type: application/json
```

```
{
  "LifecyclePolicies": [
    {
      "TransitionToArchive": "string",
      "TransitionToIA": "string",
      "TransitionToPrimaryStorageClass": "string"
    }
  ]
}
```

URI-Anfrageparameter

Die Anforderung verwendet die folgenden URI-Parameter.

[FileSystemId](#)

Die ID des Dateisystems, für das Sie das LifecycleConfiguration-Objekt erstellen (Zeichenfolge).

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Erforderlich: Ja

Anforderungstext

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

[LifecyclePolicies](#)

Ein Array von LifecyclePolicy-Objekten, die das LifecycleConfiguration-Objekt des Dateisystems definieren. Ein LifecycleConfiguration Objekt informiert das Lebenszyklusmanagement über Folgendes:

- **TransitionToIA**— Wann müssen Dateien im Dateisystem vom primären Speicher (Standard-Speicherklasse) in den IA-Speicher (Infrequent Access) verschoben werden?
- **TransitionToArchive**— Wann werden Dateien im Dateisystem aus ihrer aktuellen Speicherklasse (entweder IA oder Standardspeicher) in den Archivspeicher verschoben.

Dateisysteme können nicht in den Archivspeicher übergehen, bevor sie in den IA-Speicher übergegangen sind. Daher TransitionToArchive muss es entweder nicht gesetzt werden oder muss später als TransitionTo IA sein.

Note

Die Speicherklasse Archive ist nur für Dateisysteme verfügbar, die den Elastic Throughput-Modus und den Allzweck-Performance-Modus verwenden.

- **TransitionToPrimaryStorageClass**— Gibt an, ob Dateien im Dateisystem zurück in den Primärspeicher (Standard-Speicherklasse) verschoben werden sollen, nachdem auf sie im IA- oder Archivspeicher zugegriffen wurde.

 Note

Wenn Sie den `put-lifecycle-configuration`-CLI-Befehl oder die `PutLifecycleConfiguration`-API-Aktion verwenden, verlangt Amazon EFS, dass jedes `LifecyclePolicy`-Objekt nur einen einzigen Übergang hat. Das bedeutet, dass `LifecyclePolicies` in einem Anfragetext als ein Array von `LifecyclePolicy`-Objekten strukturiert sein muss, ein Objekt für jeden Speicherübergang. Weitere Informationen finden Sie in den Beispielanforderungen im folgenden Abschnitt.

Typ: Array von [LifecyclePolicy](#)-Objekten

Array-Mitglieder: Maximale Anzahl von 3 Elementen.

Erforderlich: Ja

Antwortsyntax

```
HTTP/1.1 200
Content-type: application/json

{
  "LifecyclePolicies": [
    {
      "TransitionToArchive": "string",
      "TransitionToIA": "string",
      "TransitionToPrimaryStorageClass": "string"
    }
  ]
}
```

Antwortelemente

Wenn die Aktion erfolgreich ist, sendet der Service eine HTTP 200-Antwort zurück.

Die folgenden Daten werden vom Service im JSON-Format zurückgegeben.

[LifecyclePolicies](#)

Eine Reihe von Richtlinien für das Lebenszyklusmanagement. EFS unterstützt maximal eine Richtlinie pro Dateisystem.

Typ: Array von [LifecyclePolicy](#)-Objekten

Array-Mitglieder: Maximale Anzahl von 3 Elementen.

Fehler

BadRequest

Wird zurückgegeben, wenn die Anfrage fehlerhaft ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene `FileSystemId` Wert im Wert des Anforderers nicht vorhanden ist AWS-Konto.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können.

Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

`IncorrectFileSystemLifeCycleState`

Wird zurückgegeben, wenn der Lebenszyklusstatus des Dateisystems nicht „verfügbar“ ist.

`ErrorCode`

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

`Message`

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden.

Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können.

Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 409

`InternalServerError`

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

`ErrorCode`

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

`Message`

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden.

Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können.

Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 500

Beispiele

Erstellung einer Lebenszyklus-Konfiguration

Im folgenden Beispiel wird mithilfe der `PutLifecycleConfiguration`-Aktion ein `LifecyclePolicy`-Objekt erstellt. In diesem Beispiel wird eine Lebenszyklusrichtlinie erstellt, die EFS anweist, Folgendes zu tun:

- Verschieben Sie alle Dateien im Dateisystem, auf die in den letzten 30 Tagen nicht im Standardspeicher zugegriffen wurde, in den IA-Speicher.
- Verschieben Sie alle Dateien im Dateisystem, auf die in den letzten 90 Tagen nicht im Standardspeicher zugegriffen wurde, in den Archivspeicher.
- Verschieben Sie Dateien zurück in den Standardspeicher, nachdem auf sie im IA- oder Archivspeicher zugegriffen wurde. Die Speicherklasse `Archive` ist nur für Dateisysteme verfügbar, die den Elastic Throughput-Modus und den Allzweck-Performance-Modus verwenden.

Weitere Informationen finden Sie unter [EFS-Speicherklassen](#) und [Verwalten von Dateisystemspeicher](#).

Beispielanforderung

```
PUT /2015-02-01/file-systems/fs-0123456789abcdefb/lifecycle-configuration HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20181122T232908Z
Authorization: <...>
Content-type: application/json
Content-Length: 86

{
  "LifecyclePolicies": [
    {
      "TransitionToArchive": "AFTER_90_DAYS"
    },
    {
      "TransitionToIA": "AFTER_30_DAYS"
    },
    {
      "TransitionToPrimaryStorage": "AFTER_1_ACCESS"
    }
  ]
}
```

```
}
```

Beispielantwort

```
HTTP/1.1 200 OK
x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
Content-type: application/json
Content-Length: 86

{
  "LifecyclePolicies": [
    {
      "TransitionToArchive": "AFTER_90_DAYS"
    },
    {
      "TransitionToIA": "AFTER_30_DAYS"
    },
    {
      "TransitionToPrimaryStorage": "AFTER_1_ACCESS"
    }
  ]
}
```

Beispiel für eine put-lifecycle-configuration CLI-Anfrage

Dieses Beispiel veranschaulicht eine Verwendung von PutLifecycleConfiguration.

Beispielanforderung

```
aws efs put-lifecycle-configuration \
  --file-system-id fs-0123456789abcdefb \
  --lifecycle-policies "[{"TransitionToArchive":"AFTER_90_DAYS"},
  {"TransitionToIA":"AFTER_30_DAYS"},
  {"TransitionToPrimaryStorageClass":"AFTER_1_ACCESS"}]"
--region us-west-2 \
--profile adminuser
```

Beispielantwort

```
{
  "LifecyclePolicies": [
```

```
{
  "TransitionToArchive": "AFTER_90_DAYS"
},
{
  "TransitionToIA": "AFTER_30_DAYS"
},
{
  "TransitionToPrimaryStorageClass": "AFTER_1_ACCESS"
}
]
```

Lebenszyklusmanagement deaktivieren

Das folgende Beispiel deaktiviert das Lebenszyklusmanagement für das angegebene Dateisystem.

Beispielanforderung

```
PUT /2015-02-01/file-systems/fs-01234567/lifecycle-configuration HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20181122T232908Z
Authorization: <...>
Content-type: application/json
Content-Length: 86

{
  "LifecyclePolicies": [ ]
}
```

Beispielantwort

```
HTTP/1.1 200 OK
x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
Content-type: application/json
Content-Length: 86

{
  "LifecyclePolicies": [ ]
}
```

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen AWS SDKs finden Sie im Folgenden:

- [AWS Befehlszeilenschnittstelle V2](#)
- [AWS SDK for .NET V4](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS SDK für Kotlin](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK for Ruby V3](#)

TagResource

Erstellt ein Tag für eine EFS-Ressource. Mit diesem API-Vorgang können Sie Tags für EFS-Dateisysteme und Zugangspunkte erstellen.

Diese Operation erfordert Berechtigungen für die Aktion `elasticfilesystem:TagResource`.

Anforderungssyntax

```
POST /2015-02-01/resource-tags/ResourceId HTTP/1.1
```

```
Content-type: application/json
```

```
{
  "Tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ]
}
```

URI-Anfrageparameter

Die Anforderung verwendet die folgenden URI-Parameter.

ResourceId

Die ID, die die EFS-Ressource angibt, für die Sie ein Tag erstellen möchten.

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:(access-point/fsap|file-system/fs)-[0-9a-f]{8,40}|fs(ap)?-[0-9a-f]{8,40})$`

Erforderlich: Ja

Anforderungstext

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

Tags

Ein Array von Tag-Objekten, die hinzugefügt werden sollen. Jedes Tag-Objekt ist ein Schlüssel-Wert-Paar.

Typ: Array von [Tag](#)-Objekten

Erforderlich: Ja

Antwortsyntax

```
HTTP/1.1 200
```

Antwortelemente

Wenn die Aktion erfolgreich ist, gibt der Dienst eine HTTP 200-Antwort mit leerem HTTP-Textinhalt zurück.

Fehler

AccessPointNotFound

Wird zurückgegeben, wenn der angegebene `AccessPointId` Wert im Wert des Anforderers nicht vorhanden ist AWS-Konto.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

BadRequest

Wird zurückgegeben, wenn die Anfrage fehlerhaft ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene FileSystemId Wert im Wert des Anforderers nicht vorhanden ist. AWS-Konto

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 500

Beispiele

Tags in einem Dateisystem erstellen

Die folgende Anforderung erstellt drei Tags ("key1", "key2", und "key3") im angegebenen Dateisystem.

Beispielanforderung

```
POST /2015-02-01/tag-resource/fs-01234567 HTTP/1.1
Host: elasticfilesystem.us-west-2.amazonaws.com
x-amz-date: 20140620T221118Z
Authorization: <...>
Content-Type: application/json
Content-Length: 160
```

```
{
  "Tags": [
    {
      "Key": "key1",
      "Value": "value1"
    },
    {
      "Key": "key2",
      "Value": "value2"
    },
    {
      "Key": "key3",
      "Value": "value3"
    }
  ]
}
```

```
{
  "Key": "key3",
  "Value": "value3"
}
]
```

Beispielantwort

```
HTTP/1.1 204 no content
x-amzn-RequestId: 01234567-89ab-cdef-0123-456789abcdef
```

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen finden Sie im AWS SDKs Folgenden:

- [AWS Befehlszeilenschnittstelle V2](#)
- [AWS SDK for .NET V4](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS SDK für Kotlin](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK for Ruby V3](#)

UntagResource

Entfernt Tags aus einer EFS-Ressource. Mit diesem API-Vorgang können Sie Tags aus EFS-Dateisystemen und Zugangspunkten entfernen.

Diese Operation erfordert Berechtigungen für die Aktion `elasticfilesystem:UntagResource`.

Anforderungssyntax

```
DELETE /2015-02-01/resource-tags/ResourceId?tagKeys=TagKeys HTTP/1.1
```

URI-Anfrageparameter

Die Anforderung verwendet die folgenden URI-Parameter.

ResourceId

Gibt die EFS-Ressource an, von der Sie Tags entfernen möchten.

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:(access-point/fsap|file-system/fs)-[0-9a-f]{8,40}|fs(ap)?-[0-9a-f]{8,40})$`

Erforderlich: Ja

TagKeys

Die Schlüssel der Schlüssel-Wert-Tag-Paare, die Sie aus der angegebenen EFS-Ressource entfernen möchten.

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Die maximale Anzahl beträgt 50 Elemente.

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 128 Zeichen.

Pattern: `^(?![aA]{1}[wW]{1}[sS]{1}:)([\\p{L}\\p{Z}\\p{N}_.:/=+\\-@]+)$`

Erforderlich: Ja

Anforderungstext

Der Anforderung besitzt keinen Anforderungstext.

Antwortsyntax

```
HTTP/1.1 200
```

Antwortelemente

Wenn die Aktion erfolgreich ist, gibt der Dienst eine HTTP 200-Antwort mit leerem HTTP-Textinhalt zurück.

Fehler

AccessPointNotFound

Wird zurückgegeben, wenn der angegebene `AccessPointId` Wert im Wert des Anforderers nicht vorhanden ist AWS-Konto.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

BadRequest

Wird zurückgegeben, wenn die Anfrage fehlerhaft ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene `FileSystemId` Wert im Wert des Anforderers nicht vorhanden ist. AWS-Konto

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden.

Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 500

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen finden Sie im AWS SDKs Folgenden:

- [AWS Befehlszeilenschnittstelle V2](#)
- [AWS SDK for .NET V4](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS SDK für Kotlin](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK for Ruby V3](#)

UpdateFileSystem

Aktualisiert den Durchsatz oder die Menge des bereitgestellten Durchsatzes eines vorhandenen Dateisystems.

Anforderungssyntax

```
PUT /2015-02-01/file-systems/FileSystemId HTTP/1.1
Content-type: application/json

{
  "ProvisionedThroughputInMibps": number,
  "ThroughputMode": "string"
}
```

URI-Anfrageparameter

Die Anforderung verwendet die folgenden URI-Parameter.

FileSystemId

Die ID des Dateisystems, das Sie aktualisieren möchten.

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Erforderlich: Ja

Anforderungstext

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

ProvisionedThroughputInMibps

(Optional) Der Durchsatz, gemessen in Mebibyte pro Sekunde (MiBps), den Sie für ein Dateisystem bereitstellen möchten, das Sie erstellen. Erforderlich, wenn `ThroughputMode` auf `provisioned` festgelegt wird. Gültige Werte sind 1—3414 MiBps, wobei die Obergrenze von der Region abhängt. Um diesen Grenzwert zu erhöhen, wenden Sie sich an [Support](#). Weitere

Informationen finden Sie unter [Amazon-EFS-Kontingente, die Sie erhöhen können](#) im Amazon-EFS-Benutzerhandbuch.

Type: Double

Gültiger Bereich: Mindestwert 1.0.

Erforderlich: Nein

ThroughputMode

(Optional) Aktualisiert den Durchsatzmodus des Dateisystems. Wenn Sie Ihren Durchsatzmodus nicht aktualisieren, müssen Sie diesen Wert in Ihrer Anfrage nicht angeben. Wenn Sie ThroughputMode in provisioned ändern, müssen Sie auch einen Wert für ProvisionedThroughputInMibps festlegen.

Typ: Zeichenfolge

Zulässige Werte: bursting | provisioned | elastic

Erforderlich: Nein

Antwortsyntax

HTTP/1.1 202

Content-type: application/json

```
{
  "AvailabilityZoneId": "string",
  "AvailabilityZoneName": "string",
  "CreationTime": number,
  "CreationToken": "string",
  "Encrypted": boolean,
  "FileSystemArn": "string",
  "FileSystemId": "string",
  "FileSystemProtection": {
    "ReplicationOverwriteProtection": "string"
  },
  "KmsKeyId": "string",
  "LifecycleState": "string",
  "Name": "string",
  "NumberOfMountTargets": number,
  "OwnerId": "string",
```

```
"PerformanceMode": "string",
"ProvisionedThroughputInMibps": number,
"SizeInBytes": {
  "Timestamp": number,
  "Value": number,
  "ValueInArchive": number,
  "ValueInIA": number,
  "ValueInStandard": number
},
"Tags": [
  {
    "Key": "string",
    "Value": "string"
  }
],
"ThroughputMode": "string"
}
```

Antwortelemente

Wenn die Aktion erfolgreich ist, sendet der Service eine HTTP 202-Antwort zurück.

Die folgenden Daten werden vom Service im JSON-Format zurückgegeben.

AvailabilityZoneId

Die eindeutige und konsistente Kennung der Availability Zone, in der sich das Dateisystem befindet. Sie ist nur für One-Zone-Dateisysteme gültig. use1-az1 ist beispielsweise eine Availability Zone ID für die US-East-1 AWS-Region, und sie hat in jedem Fall den gleichen Standort. AWS-Konto

Typ: Zeichenfolge

AvailabilityZoneName

Beschreibt die AWS Availability Zone, in der sich das Dateisystem befindet, und ist nur für One Zone-Dateisysteme gültig. Weitere Informationen finden Sie unter [Verwenden von EFS-Speicherklassen](#) im Amazon-EFS-Benutzerhandbuch.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 64 Zeichen.

Pattern: . +

CreationTime

Die Zeit, zu der das Dateisystem erstellt wurde, in Sekunden (seit 1970-01-01T00:00:00Z).

Typ: Zeitstempel

CreationToken

Die Opaque-Zeichenfolge, die in der Anforderung angegeben wird.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 64 Zeichen.

Pattern: .+

Encrypted

Ein boolescher Wert, der mit True anzeigt, dass das Dateisystem verschlüsselt ist.

Typ: Boolescher Wert

FileSystemArn

Der Amazon-Ressourcenname (ARN) für das EFS-Dateisystem, im Format `arn:aws:elasticfilesystem:region:account-id:file-system/file-system-id`. Beispiel mit Beispieldaten: `arn:aws:elasticfilesystem:us-west-2:1111333322228888:file-system/fs-01234567`

Typ: Zeichenfolge

FileSystemId

Die von Amazon EFS zugewiesene ID des Dateisystems.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

FileSystemProtection

Gibt den Schutz des Dateisystems an.

Typ: [FileSystemProtectionDescription](#) Objekt

KmsKeyId

Die ID eines, das zum Schutz des verschlüsselten Dateisystems AWS KMS key verwendet wird.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 2 048 Zeichen.

Pattern: `^([0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}|mrk-[0-9a-f]{32}|alias/[a-zA-Z0-9/_-]+|(arn:aws[-a-z]*:kms:[a-z0-9-]+:\d{12}:((key/[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12})|(key/mrk-[0-9a-f]{32})|(alias/[a-zA-Z0-9/_-]+))))$`

LifeCycleState

Die Lebenszyklusphase des Dateisystems.

Typ: Zeichenfolge

Zulässige Werte: `creating | available | updating | deleting | deleted | error`

Name

Sie können einem Dateisystem Tags hinzufügen, einschließlich eines Name-Tags. Weitere Informationen finden Sie unter [CreateFileSystem](#). Wenn das Dateisystem über ein Name-Tag verfügt, gibt Amazon EFS den Wert in diesem Feld zurück.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 256 Zeichen.

Pattern: `^([\p{L}\p{Z}\p{N}_.: /+=\ -@]*)$`

NumberOfMountTargets

Die aktuelle Anzahl von Mounting-Zielen, die das Dateisystem aufweist. Weitere Informationen finden Sie unter [CreateMountTarget](#).

Typ: Ganzzahl

Gültiger Bereich: Mindestwert 0.

OwnerId

AWS-Konto Derjenige, der das Dateisystem erstellt hat.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 14 Zeichen.

Pattern: `^(\\d{12})|(\\d{4}-\\d{4}-\\d{4})$`

[PerformanceMode](#)

Der Leistungsmodus des Dateisystems.

Typ: Zeichenfolge

Zulässige Werte: `generalPurpose` | `maxIO`

[ProvisionedThroughputInMibps](#)

Die Menge des bereitgestellten Durchsatzes, gemessen in MiBps, für das Dateisystem. Gültig für Dateisysteme, bei denen `ThroughputMode` auf `provisioned` eingestellt ist.

Type: Double

Gültiger Bereich: Mindestwert 1.0.

[SizeInBytes](#)

Die letzte bekannte gemessene Größe (in Bytes) der im Dateisystem gespeicherten Daten im Feld `Value` und die Zeit, zu der diese Größe ermittelt wurde, im Feld `Timestamp`. Der Wert `Timestamp` ist die ganzzahlige Anzahl der Sekunden seit 1970-01-01T00:00:00Z. Der Wert `SizeInBytes` steht nicht für die Größe eines konsistenten Snapshots des Dateisystems, ist aber letztlich konsistent, wenn keine Schreibvorgänge im Dateisystem vorgenommen werden. Das heißt, `SizeInBytes` steht nur dann für die tatsächliche Größe, wenn das Dateisystem länger als einige Stunden nicht verändert wurde. Andernfalls entspricht der Wert nicht exakt der Größe, die das Dateisystem zu einem beliebigen Zeitpunkt hatte.

Typ: [FileSystemSize](#) Objekt

[Tags](#)

Die Tags, die dem Dateisystem zugeordnet sind, dargestellt als ein Array von Tag-Objekten.

Typ: Array von [Tag](#)-Objekten

[ThroughputMode](#)

Zeigt den Durchsatzmodus des Dateisystems an. Weitere Informationen finden Sie unter [Durchsatzmodi](#) im Amazon-EFS-Benutzerhandbuch.

Typ: Zeichenfolge

Zulässige Werte: `bursting` | `provisioned` | `elastic`

Fehler

BadRequest

Wird zurückgegeben, wenn die Anfrage fehlerhaft ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene `FileSystemId` Wert im Wert des Anforderers nicht vorhanden ist. AWS-Konto

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können.

Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

`IncorrectFileSystemLifecycleState`

Wird zurückgegeben, wenn der Lebenszyklusstatus des Dateisystems nicht „verfügbar“ ist.

`ErrorCode`

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

`Message`

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 409

`InsufficientThroughputCapacity`

Wird zurückgegeben, wenn die Kapazität nicht ausreicht, um zusätzlichen Durchsatz bereitzustellen. Dieser Wert kann zurückgegeben werden, wenn Sie versuchen, ein Dateisystem im Modus „Bereitgestellter Durchsatz“ zu erstellen, wenn Sie versuchen, den bereitgestellten Durchsatz eines vorhandenen Dateisystems zu erhöhen oder wenn Sie versuchen, ein vorhandenes Dateisystem vom Modus „Bursting Throughput“ auf „Bereitgestellter Durchsatz“ umzustellen. Bitte versuchen Sie es später erneut.

`ErrorCode`

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

`Message`

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine

Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können.

Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 503

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert.

Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden.

Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können.

Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 500

ThroughputLimitExceeded

Wird zurückgegeben, wenn der Durchsatzmodus oder die Menge des bereitgestellten Durchsatzes nicht geändert werden können, da die Durchsatzgrenze von 1024 erreicht MiB/s wurde.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert.

Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden.

Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können.

Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

TooManyRequests

Wird zurückgegeben, wenn Sie nicht mindestens 24 Stunden warten, bevor Sie entweder den Durchsatzmodus ändern oder den Wert für den bereitgestellten Durchsatz verringern.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 429

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen finden Sie im AWS SDKs Folgenden:

- [AWS Befehlszeilenschnittstelle V2](#)
- [AWS SDK for .NET V4](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS SDK für Kotlin](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK for Ruby V3](#)

UpdateFileSystemProtection

Aktualisiert den Schutz für ein Dateisystem.

Diese Operation erfordert Berechtigungen für die Aktion `elasticfilesystem:UpdateFileSystemProtection`.

Anforderungssyntax

```
PUT /2015-02-01/file-systems/FileSystemId/protection HTTP/1.1
Content-type: application/json
```

```
{
  "ReplicationOverwriteProtection": "string"
}
```

URI-Anfrageparameter

Die Anforderung verwendet die folgenden URI-Parameter.

[FileSystemId](#)

Die ID des zu aktualisierenden Dateisystems.

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Erforderlich: Ja

Anforderungstext

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

[ReplicationOverwriteProtection](#)

Der Status des Replikationsüberschreibschutzes des Dateisystems.

- **ENABLED** – Das Dateisystem kann nicht als Zieldateisystem in einer Replikationskonfiguration verwendet werden. Das Dateisystem ist beschreibbar. Der Überschreibschutz für die Replikation ist standardmäßig **ENABLED**.

- **DISABLED** – Das Dateisystem kann als Zieldateisystem in einer Replikationskonfiguration verwendet werden. Das Dateisystem ist schreibgeschützt und kann nur durch EFS-Replikation geändert werden.
- **REPLICATING** – Das Dateisystem wird als Zieldateisystem in einer Replikationskonfiguration verwendet. Das Dateisystem ist schreibgeschützt und wird nur durch die EFS-Replikation geändert.

Wenn die Replikationskonfiguration gelöscht wird, wird der Replikationsüberschreibschutz des Dateisystems wieder aktiviert und das Dateisystem wird beschreibbar.

Typ: Zeichenfolge

Zulässige Werte: **ENABLED** | **DISABLED** | **REPLICATING**

Erforderlich: Nein

Antwortsyntax

```
HTTP/1.1 200
Content-type: application/json

{
  "ReplicationOverwriteProtection": "string"
}
```

Antwortelemente

Wenn die Aktion erfolgreich ist, sendet der Service eine HTTP 200-Antwort zurück.

Die folgenden Daten werden vom Service im JSON-Format zurückgegeben.

[ReplicationOverwriteProtection](#)

Der Status des Replikationsüberschreibschutzes des Dateisystems.

- **ENABLED** – Das Dateisystem kann nicht als Zieldateisystem in einer Replikationskonfiguration verwendet werden. Das Dateisystem ist beschreibbar. Der Überschreibschutz für die Replikation ist standardmäßig **ENABLED**.
- **DISABLED** – Das Dateisystem kann als Zieldateisystem in einer Replikationskonfiguration verwendet werden. Das Dateisystem ist schreibgeschützt und kann nur durch EFS-Replikation geändert werden.

- **REPLICATING** – Das Dateisystem wird als Zieldateisystem in einer Replikationskonfiguration verwendet. Das Dateisystem ist schreibgeschützt und wird nur durch EFS-Replikation geändert.

Wenn die Replikationskonfiguration gelöscht wird, wird der Replikationsüberschreibschutz des Dateisystems wieder aktiviert und das Dateisystem wird beschreibbar.

Typ: Zeichenfolge

Zulässige Werte: **ENABLED** | **DISABLED** | **REPLICATING**

Fehler

BadRequest

Wird zurückgegeben, wenn die Anfrage fehlerhaft ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene **FileSystemId** Wert im Wert des Anforderers nicht vorhanden ist. AWS-Konto

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 404

IncorrectFileSystemLifecycleState

Wird zurückgegeben, wenn der Lebenszyklusstatus des Dateisystems nicht „verfügbar“ ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 409

InsufficientThroughputCapacity

Wird zurückgegeben, wenn die Kapazität nicht ausreicht, um zusätzlichen Durchsatz bereitzustellen. Dieser Wert kann zurückgegeben werden, wenn Sie versuchen, ein Dateisystem im Modus „Bereitgestellter Durchsatz“ zu erstellen, wenn Sie versuchen, den bereitgestellten Durchsatz eines vorhandenen Dateisystems zu erhöhen oder wenn Sie versuchen, ein vorhandenes Dateisystem vom Modus „Bursting Throughput“ auf „Bereitgestellter Durchsatz“ umzustellen. Bitte versuchen Sie es später erneut.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 503

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP Status Code: 500

ReplicationAlreadyExists

Wird zurückgegeben, wenn das Dateisystem bereits in einer Replikationskonfiguration enthalten ist. >

ErrorCode

Der Fehlercode ist eine Zeichenfolge, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden.

Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 409

ThroughputLimitExceeded

Wird zurückgegeben, wenn der Durchsatzmodus oder die Menge des bereitgestellten Durchsatzes nicht geändert werden können, da die Durchsatzgrenze von 1024 erreicht MiB/s wurde.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können. Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 400

TooManyRequests

Wird zurückgegeben, wenn Sie nicht mindestens 24 Stunden warten, bevor Sie entweder den Durchsatzmodus ändern oder den Wert für den bereitgestellten Durchsatz verringern.

ErrorCode

Der Fehlercode ist eine Zeichenfolge handelt, die eine Fehlerbedingung eindeutig identifiziert. Er ist für Programme vorgesehen, die Fehler dem Typ nach erkennen und verarbeiten.

Message

Die Fehlermeldung enthält eine generische Beschreibung der Fehlerbedingung in englischer Sprache. Sie ist für dafür vorgesehen, von Menschen gelesen zu werden. Einfache Programme zeigen dem Endbenutzer die Meldung direkt an, wenn sie auf eine

Fehlerbedingung treffen, die sie nicht kennen, oder die sie nicht verarbeiten können.

Komplexere Programme mit einer umfangreicheren Fehlerverarbeitung und einer geeigneten Internationalisierung werden die Fehlermeldung eher ignorieren.

HTTP-Statuscode: 429

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen finden Sie im AWS SDKs Folgenden:

- [AWS Befehlszeilenschnittstelle V2](#)
- [AWS SDK for .NET V4](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS SDK für Kotlin](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK für Python](#)
- [AWS SDK for Ruby V3](#)

Datentypen

Die folgenden Datentypen werden unterstützt:

- [AccessPointDescription](#)
- [BackupPolicy](#)
- [CreationInfo](#)
- [Destination](#)
- [DestinationToCreate](#)
- [FileSystemDescription](#)
- [FileSystemProtectionDescription](#)
- [FileSystemSize](#)

- [LifecyclePolicy](#)
- [MountTargetDescription](#)
- [PosixUser](#)
- [ReplicationConfigurationDescription](#)
- [ResourceIdPreference](#)
- [RootDirectory](#)
- [Tag](#)

AccessPointDescription

Liefert eine Beschreibung eines EFS-Dateisystem-Zugangspunkts.

Inhalt

AccessPointArn

Der eindeutige Amazon-Ressourcenname (ARN), der mit dem Zugangspunkt verbunden ist.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:access-point/fsap-[0-9a-f]{8,40}$`

Erforderlich: Nein

AccessPointId

Die ID des Zugangspunkts, die von Amazon EFS zugewiesen wird.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:access-point/fsap-[0-9a-f]{8,40}|fsap-[0-9a-f]{8,40})$`

Erforderlich: Nein

ClientToken

Die Opaque-Zeichenfolge, die in der Anforderung angegeben wird, um eine idempotente Erstellung zu gewährleisten.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 64 Zeichen.

Pattern: `.+`

Erforderlich: Nein

FileSystemId

Die ID des EFS-Dateisystems, auf das der Zugangspunkt angewendet wird.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Erforderlich: Nein

LifeCycleState

Gibt die Lebenszyklusphase des Zugangspunkts an.

Typ: Zeichenfolge

Zulässige Werte: `creating | available | updating | deleting | deleted | error`

Erforderlich: Nein

Name

Der Name des Zugangspunkts. Dies ist der Wert des Name-Tags.

Typ: Zeichenfolge

Erforderlich: Nein

OwnerId

Identifiziert den AWS-Konto , dem die Access Point-Ressource gehört.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 14 Zeichen.

Pattern: `^(\\d{12})|(\\d{4}-\\d{4}-\\d{4})$`

Erforderlich: Nein

PosixUser

Die vollständige POSIX-Identität, einschließlich der Benutzer-ID, der Gruppen-ID und der sekundären Gruppe IDs auf dem Access Point, die für alle Dateioperationen von NFS-Clients verwendet wird, die den Access Point verwenden.

Typ: [PosixUser](#) Objekt

Erforderlich: Nein

RootDirectory

Das Verzeichnis im EFS-Dateisystem, das der Zugangspunkt als Stammverzeichnis für NFS-Clients verfügbar macht, die den Zugangspunkts verwenden.

Typ: [RootDirectory](#) Objekt

Erforderlich: Nein

Tags

Die mit dem Zugangspunkt verknüpften Tags, dargestellt als ein Array von Tag-Objekten.

Typ: Array von [Tag](#)-Objekten

Erforderlich: Nein

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen finden Sie im Folgenden AWS SDKs:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

BackupPolicy

Die Backup-Richtlinie für das Dateisystem, die zur Erstellung automatischer täglicher Backups verwendet wird. Wenn der Status den Wert **ENABLED** hat, wird das Dateisystem automatisch gesichert. Weitere Informationen finden Sie unter [Automatische Backups](#).

Inhalt

Status

Beschreibt den Status der Backup-Richtlinie des Dateisystems.

- **ENABLED** – EFS erstellt automatisch eine Sicherungskopie des Dateisystems.
- **ENABLING** – EFS schaltet automatische Backups für das Dateisystem ein.
- **DISABLED** – Automatische Backups sind für das Dateisystem ausgeschaltet.
- **DISABLING** – EFS schaltet automatische Backups für das Dateisystem aus.

Typ: Zeichenfolge

Zulässige Werte: **ENABLED** | **ENABLING** | **DISABLED** | **DISABLING**

Erforderlich: Ja

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen AWS SDKs finden Sie im Folgenden:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

CreationInfo

Erforderlich, wenn das angegebene `RootDirectory > Path` nicht vorhanden ist. Gibt die POSIX IDs und die Berechtigungen an, die auf das `RootDirectory > Path` des Access Points angewendet werden sollen. Wenn das Stammverzeichnis des Zugriffspunkts nicht vorhanden ist, erstellt EFS es mit diesen Einstellungen, wenn ein Client eine Verbindung mit dem Zugriffspunkt herstellt. Wenn Sie `CreationInfo` angeben, müssen Sie Werte für alle Eigenschaften einschließen.

Amazon EFS erstellt nur dann ein Stammverzeichnis, wenn Sie `CreationInfo`: `OwnUid`, `ownGID` und Berechtigungen für das Verzeichnis angegeben haben. Wenn Sie diese Informationen nicht angeben, erstellt Amazon EFS das Stammverzeichnis nicht. Wenn das Stammverzeichnis nicht existiert, schlagen Mount-Versuche beim Zugangspunkt fehl.

Important

Wenn Sie `CreationInfo` nicht angeben und das angegebene `RootDirectory` nicht vorhanden ist, schlagen Versuche, das Dateisystem mithilfe des Zugriffspunkts zu mounten, fehl.

Inhalt

OwnerGid

Gibt die POSIX-Gruppen-ID an, die auf die `RootDirectory` angewendet werden soll. Akzeptiert Werte von 0 bis 2^{32} (4294967295).

Type: Long

Gültiger Bereich: Mindestwert 0. Maximaler Wert von 4294967295.

Erforderlich: Ja

OwnerUid

Gibt die POSIX-Benutzer-ID an, die auf die `RootDirectory` angewendet werden soll. Akzeptiert Werte von 0 bis 2^{32} (4294967295).

Type: Long

Gültiger Bereich: Mindestwert 0. Maximaler Wert von 4294967295.

Erforderlich: Ja

Permissions

Gibt die POSIX-Berechtigungen an, die auf `RootDirectory` angewendet werden sollen, im Format einer Oktalzahl, die die Modusbits der Datei darstellt.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 3. Maximale Länge beträgt 4 Zeichen.

Pattern: `^[0-7]{3,4}$`

Erforderlich: Ja

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen finden Sie im Folgenden AWS SDKs:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Destination

Beschreibt das Zielsystem in der Replikationskonfiguration.

Inhalt

FileSystemId

Die ID des Ziel-Amazon-EFS-Dateisystems.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Erforderlich: Ja

Region

Das, AWS-Region in dem sich das Zielsystem befindet.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 64 Zeichen.

Pattern: `^[a-z]{2}-((iso[a-z]{0,1}-)|(gov-)){0,1}[a-z]+-{0,1}[0-9]{0,1}$`

Erforderlich: Ja

Status

Beschreibt den Status der Replikationskonfiguration. Weitere Informationen zum Replikationsstatus finden Sie unter [Replizierungsdetails anzeigen](#) im Amazon EFS-Benutzerhandbuch.

Typ: Zeichenfolge

Zulässige Werte: ENABLED | ENABLING | DELETING | ERROR | PAUSED | PAUSING

Erforderlich: Ja

LastReplicatedTimestamp

Der Zeitpunkt, zu dem die letzte Synchronisierung auf dem Zielsystem erfolgreich abgeschlossen wurde. Alle Änderungen an Daten im Quellsystem, die vor diesem Zeitpunkt vorgenommen wurden, wurden erfolgreich in das Zielsystem repliziert. Alle Änderungen, die nach diesem Zeitpunkt vorgenommen wurden, werden möglicherweise nicht vollständig repliziert.

Typ: Zeitstempel

Erforderlich: Nein

OwnerId

ID des Dateisystems, AWS-Konto in dem sich das Zielsystem befindet.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 14 Zeichen.

Pattern: `^(\d{12})|(\d{4}-\d{4}-\d{4})$`

Erforderlich: Nein

RoleArn

Amazon-Ressourcenname (ARN) der IAM-Rolle im Quellkonto, das es Amazon EFS ermöglicht, die Replikation in seinem Namen durchzuführen. Dies ist optional für die Replikation mit demselben Konto und für die kontenübergreifende Replikation erforderlich.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 2 048 Zeichen.

Pattern: `arn:(aws[a-zA-Z-]*)?:iam::\d{12}:role/?[a-zA-Z_0-9+=,.\@-_/]+`

Erforderlich: Nein

StatusMessage

Nachricht, die Details zur Konfiguration des Replikationsziels PAUSED oder zum ERROR Status der Replikationszielkonfiguration enthält. Weitere Informationen zu Replikationsstatusmeldungen finden Sie unter [Replizierungsdetails anzeigen](#) im Amazon EFS-Benutzerhandbuch.

Typ: Zeichenfolge

Erforderlich: Nein

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen AWS SDKs finden Sie im Folgenden:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

DestinationToCreate

Beschreibt das neue oder vorhandene Zielspeichersystem für die Replikationskonfiguration.

- Wenn Sie in ein neues Speichersystem replizieren möchten, geben Sie nicht die Speichersystem-ID für das Zielspeichersystem an. Amazon EFS erstellt ein neues, leeres Speichersystem. Geben Sie für One Zone Storage die Availability Zone an, in der das Speichersystem erstellt werden soll. Wenn Sie einen anderen AWS Key Management Service Schlüssel als den Standard-KMS-Schlüssel verwenden möchten, geben Sie ihn an. Weitere Informationen finden Sie unter [Konfiguration der Replikation auf ein neues Amazon EFS-Speichersystem](#) im Amazon EFS-Benutzerhandbuch.

Note

Nachdem das Speichersystem erstellt wurde, können Sie den KMS-Schlüssel oder den Leistungsmodus nicht mehr ändern.

- Wenn Sie auf ein vorhandenes Speichersystem replizieren möchten, das sich in demselben Konto wie das Quellspeichersystem befindet, müssen Sie die ID oder den Amazon-Ressourcennamen (ARN) des Speichersystems angeben, auf das repliziert werden soll. Der Replikationsschutz des Speichersystems muss deaktiviert sein. Weitere Informationen finden Sie unter [Replizieren auf ein vorhandenes Speichersystem](#) im Amazon EFS-Benutzerhandbuch.
- Wenn Sie das Speichersystem auf ein Speichersystem replizieren, das sich in einem anderen Konto als das Quellspeichersystem befindet (kontoübergreifende Replikation), müssen Sie den ARN für das Speichersystem und die IAM-Rolle angeben, die es Amazon EFS ermöglicht, die Replikation auf dem Zielkonto durchzuführen. Der Replikationsschutz des Speichersystems muss deaktiviert sein. Weitere Informationen finden Sie unter [Across replizieren AWS-Konten](#) im Amazon EFS-Benutzerhandbuch.

Inhalt

AvailabilityZoneName

Um ein Speichersystem zu erstellen, das One-Zone-Speicher verwendet, geben Sie den Namen der Availability Zone an, in der das Zielspeichersystem erstellt werden soll.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 64 Zeichen.

Pattern: . +

Erforderlich: Nein

FileSystemId

Die ID oder der ARN des Dateisystems, das für das Ziel verwendet werden soll. Für die kontoübergreifende Replikation muss es sich um einen ARN handeln. Die Replikationsüberschreibungsreplikation des Dateisystems muss deaktiviert werden. Wenn keine ID oder ARN angegeben ist, wird ein neues Dateisystem erstellt.

Note

Wenn Sie die Replikation zum ersten Mal in ein vorhandenes Dateisystem konfigurieren, schreibt Amazon EFS Daten in das Zieldateisystem oder entfernt bestehende Daten aus dem Zieldateisystem, sodass sie den Daten im Quelldateisystem entsprechen. Wenn Sie Daten im Zieldateisystem nicht ändern möchten, sollten Sie stattdessen in ein neues Dateisystem replizieren. Weitere Informationen finden Sie unter <https://docs.aws.amazon.com/efs/latest/ug/create-replication.html>.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Erforderlich: Nein

KmsKeyId

Geben Sie den Schlüssel AWS Key Management Service (AWS KMS) an, den Sie zum Verschlüsseln des Zieldateisystems verwenden möchten. Wenn Sie keinen KMS-Schlüssel angeben, verwendet Amazon EFS Ihren Standard-KMS-Schlüssel für Amazon EFS, `/aws/elasticfilesystem`. Diese ID kann eines der folgenden Formate aufweisen:

- Schlüssel-ID – Der eindeutige Bezeichner des Schlüssels, zum Beispiel `1234abcd-12ab-34cd-56ef-1234567890ab`.
- ARN — Zum Beispiel der ARN für den Schlüssel `arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab`.

- Schlüsselalias – Ein zuvor erstellter Anzeigename für einen Schlüssel, z. B. `alias/projectKey1`.
- Schlüsselalias-ARN – Ein ARN für einen Schlüsselalias, z. B. `arn:aws:kms:us-west-2:444455556666:alias/projectKey1`.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 2 048 Zeichen.

Pattern: `^([0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}|mrk-[0-9a-f]{32}|alias/[a-zA-Z0-9/_-]+|(arn:aws[-a-z]*:kms:[a-z0-9-]+:\d{12}:((key/[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12})|(key/mrk-[0-9a-f]{32})|(alias/[a-zA-Z0-9/_-]+))))$`

Erforderlich: Nein

Region

Um ein Dateisystem zu erstellen, das Regional Storage verwendet, geben Sie das an, AWS-Region in dem das Zieldateisystem erstellt werden soll. Die Region muss für die Region aktiviert sein AWS-Konto, der das Quelldateisystem gehört. Weitere Informationen finden Sie AWS-Regionen im AWS Allgemeinen Referenzhandbuch unter [Verwaltung](#).

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 64 Zeichen.

Pattern: `^[a-z]{2}-((iso[a-z]{0,1}-)|(gov-)){0,1}[a-z]+-{0,1}[0-9]{0,1}$`

Erforderlich: Nein

RoleArn

Amazon-Ressourcenname (ARN) der IAM-Rolle im Quellkonto, das es Amazon EFS ermöglicht, die Replikation in seinem Namen durchzuführen. Dies ist optional für die Replikation mit demselben Konto und für die kontenübergreifende Replikation erforderlich.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 2 048 Zeichen.

Pattern: `arn:(aws[a-zA-Z-]*)?:iam::\d{12}:role/?[a-zA-Z_0-9+=,.\@-_/]+`

Erforderlich: Nein

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen AWS SDKs finden Sie im Folgenden:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

FileSystemDescription

Eine Beschreibung des Dateisystems.

Inhalt

CreationTime

Die Zeit, zu der das Dateisystem erstellt wurde, in Sekunden (seit 1970-01-01T00:00:00Z).

Typ: Zeitstempel

Erforderlich: Ja

CreationToken

Die Opaque-Zeichenfolge, die in der Anforderung angegeben wird.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 64 Zeichen.

Pattern: .+

Erforderlich: Ja

FileSystemId

Die von Amazon EFS zugewiesene ID des Dateisystems.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Erforderlich: Ja

LifeCycleState

Die Lebenszyklusphase des Dateisystems.

Typ: Zeichenfolge

Zulässige Werte: `creating` | `available` | `updating` | `deleting` | `deleted` | `error`

Erforderlich: Ja

NumberOfMountTargets

Die aktuelle Anzahl von Mounting-Zielen, die das Dateisystem aufweist. Weitere Informationen finden Sie unter [CreateMountTarget](#).

Typ: Ganzzahl

Gültiger Bereich: Mindestwert 0.

Erforderlich: Ja

OwnerId

Der AWS-Konto , der das Dateisystem erstellt hat.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 14 Zeichen.

Pattern: `^(\d{12})|(\d{4}-\d{4}-\d{4})$`

Erforderlich: Ja

PerformanceMode

Der Leistungsmodus des Dateisystems.

Typ: Zeichenfolge

Zulässige Werte: `generalPurpose` | `maxIO`

Erforderlich: Ja

SizeInBytes

Die letzte bekannte gemessene Größe (in Bytes) der im Dateisystem gespeicherten Daten im Feld `Value` und die Zeit, zu der diese Größe ermittelt wurde, im Feld `Timestamp`. Der Wert `Timestamp` ist die ganzzahlige Anzahl der Sekunden seit 1970-01-01T00:00:00Z. Der Wert `SizeInBytes` steht nicht für die Größe eines konsistenten Snapshots des Dateisystems, ist aber letztlich konsistent, wenn keine Schreibvorgänge im Dateisystem vorgenommen werden. Das

heißt, `SizeInBytes` steht nur dann für die tatsächliche Größe, wenn das Dateisystem länger als einige Stunden nicht verändert wurde. Andernfalls entspricht der Wert nicht exakt der Größe, die das Dateisystem zu einem beliebigen Zeitpunkt hatte.

Typ: [FileSystemSize](#) Objekt

Erforderlich: Ja

Tags

Die Tags, die dem Dateisystem zugeordnet sind, dargestellt als ein Array von Tag-Objekten.

Typ: Array von [Tag](#)-Objekten

Erforderlich: Ja

AvailabilityZoneId

Die eindeutige und konsistente Kennung der Availability Zone, in der sich das Dateisystem befindet. Sie ist nur für One-Zone-Dateisysteme gültig. `use1-az1` ist beispielsweise eine Availability Zone ID für die US-East-1 AWS-Region, und sie hat in jedem Fall den gleichen Standort. AWS-Konto

Typ: Zeichenfolge

Erforderlich: Nein

AvailabilityZoneName

Beschreibt die AWS Availability Zone, in der sich das Dateisystem befindet, und ist nur für One Zone-Dateisysteme gültig. Weitere Informationen finden Sie unter [Verwenden von EFS-Speicherklassen](#) im Amazon-EFS-Benutzerhandbuch.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 64 Zeichen.

Pattern: `.+`

Erforderlich: Nein

Encrypted

Ein boolescher Wert, der mit `True` anzeigt, dass das Dateisystem verschlüsselt ist.

Typ: Boolesch

Erforderlich: Nein

FileSystemArn

Der Amazon-Ressourcenname (ARN) für das EFS-Dateisystem, im Format `arn:aws:elasticfilesystem:region:account-id:file-system/file-system-id`. Beispiel mit Beispieldaten: `arn:aws:elasticfilesystem:us-west-2:1111333322228888:file-system/fs-01234567`

Typ: Zeichenfolge

Erforderlich: Nein

FileSystemProtection

Gibt den Schutz des Dateisystems an.

Typ: [FileSystemProtectionDescription](#) Objekt

Erforderlich: Nein

KmsKeyId

Die ID eines, das zum Schutz des verschlüsselten Dateisystems AWS KMS key verwendet wird.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 2 048 Zeichen.

Pattern: `^([0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}|mrk-[0-9a-f]{32}|alias/[a-zA-Z0-9/_-]+|(arn:aws[-a-z]*:kms:[a-z0-9-]+:\d{12}:((key/[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12})|(key/mrk-[0-9a-f]{32})|(alias/[a-zA-Z0-9/_-]+))))$`

Erforderlich: Nein

Name

Sie können einem Dateisystem Tags hinzufügen, einschließlich eines Name-Tags. Weitere Informationen finden Sie unter [CreateFileSystem](#). Wenn das Dateisystem über ein Name-Tag verfügt, gibt Amazon EFS den Wert in diesem Feld zurück.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 256 Zeichen.

Pattern: `^([\p{L}\p{Z}\p{N}_.: /+=\ -@]*)$`

Erforderlich: Nein

ProvisionedThroughputInMibps

Die Menge des bereitgestellten Durchsatzes, gemessen in MiBps, für das Dateisystem. Gültig für Dateisysteme, bei denen `ThroughputMode` auf `provisioned` eingestellt ist.

Type: Double

Gültiger Bereich: Mindestwert 1.0.

Erforderlich: Nein

ThroughputMode

Zeigt den Durchsatzmodus des Dateisystems an. Weitere Informationen finden Sie unter [Durchsatzmodi](#) im Amazon-EFS-Benutzerhandbuch.

Typ: Zeichenfolge

Zulässige Werte: `bursting` | `provisioned` | `elastic`

Erforderlich: Nein

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen finden Sie im AWS SDKs Folgenden:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

FileSystemProtectionDescription

Beschreibt den Schutz eines Dateisystems.

Inhalt

ReplicationOverwriteProtection

Der Status des Replikationsüberschreibschutzes des Dateisystems.

- **ENABLED** – Das Dateisystem kann nicht als Zieldateisystem in einer Replikationskonfiguration verwendet werden. Das Dateisystem ist beschreibbar. Der Überschreibschutz für die Replikation ist standardmäßig ENABLED.
- **DISABLED** – Das Dateisystem kann als Zieldateisystem in einer Replikationskonfiguration verwendet werden. Das Dateisystem ist schreibgeschützt und kann nur durch EFS-Replikation geändert werden.
- **REPLICATING** – Das Dateisystem wird als Zieldateisystem in einer Replikationskonfiguration verwendet. Das Dateisystem ist schreibgeschützt und wird nur durch EFS-Replikation geändert.

Wenn die Replikationskonfiguration gelöscht wird, wird der Replikationsüberschreibschutz des Dateisystems wieder aktiviert und das Dateisystem wird beschreibbar.

Typ: Zeichenfolge

Zulässige Werte: ENABLED | DISABLED | REPLICATING

Erforderlich: Nein

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen finden Sie im Folgenden AWS SDKs:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

FileSystemSize

Die letzte bekannte gemessene Größe (in Bytes) der im Dateisystem gespeicherten Daten im Feld `Value` und die Zeit, zu der diese Größe ermittelt wurde, im Feld `Timestamp`. Der Wert entspricht nicht der Größe eines konsistenten Snapshot des Dateisystems, aber er ist schließlich konsistent, wenn keine Schreibvorgänge im Dateisystem stattfinden. Das heißt, der Wert entspricht nur dann der tatsächlichen Größe, wenn das Dateisystem über einen Zeitraum von mehr als ein paar Stunden nicht verändert wird. Andernfalls ist der Wert nicht notwendigerweise die genaue Größe, die das Dateisystem zu einem bestimmten Zeitpunkt hatte.

Inhalt

Value

Die letzte bekannte gemessene Größe (in Bytes) der im Dateisystem gespeicherten Daten.

Type: Long

Gültiger Bereich: Mindestwert 0.

Erforderlich: Ja

Timestamp

Der Zeitpunkt, zu dem die Größe der im Feld `Value` zurückgegebenen Daten bestimmt wurde. Der Wert ist die ganzzahlige Anzahl der Sekunden seit 1970-01-01T00:00:00Z.

Typ: Zeitstempel

Erforderlich: Nein

ValueInArchive

Die letzte bekannte gemessene Größe (in Bytes) der in der Speicherklasse Archiv gespeicherten Daten.

Type: Long

Gültiger Bereich: Mindestwert 0.

Erforderlich: Nein

ValueInIA

Die letzte bekannte gemessene Größe (in Bytes) der Daten, die in der Speicherklasse für seltenen Zugriff gespeichert sind.

Type: Long

Gültiger Bereich: Mindestwert 0.

Erforderlich: Nein

ValueInStandard

Die letzte bekannte gemessene Größe (in Bytes) der in der Speicherklasse Standard gespeicherten Daten.

Type: Long

Gültiger Bereich: Mindestwert 0.

Erforderlich: Nein

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen AWS SDKs finden Sie im Folgenden:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

LifecyclePolicy

Beschreibt eine vom Lifecycle Management verwendete Richtlinie, die festlegt, wann Dateien in Speicherklassen und aus Speicherklassen übertragen werden sollen. Weitere Informationen finden Sie unter [Verwalten des Dateisystemspeichers](#).

Note

Wenn Sie den `put-lifecycle-configuration`-CLI-Befehl oder die `PutLifecycleConfiguration`-API-Aktion verwenden, verlangt Amazon EFS, dass jedes `LifecyclePolicy`-Objekt nur einen einzigen Übergang hat. Das bedeutet, dass `LifecyclePolicies` in einem Anfragekörper als Array von `LifecyclePolicy` Objekten strukturiert sein muss, ein Objekt für jeden Übergang. Weitere Informationen finden Sie in den Anfragebeispielen in [PutLifecycleConfiguration](#).

Inhalt

TransitionToArchive

Die Anzahl der Tage nach dem letzten Zugriff auf Dateien im Primärspeicher (Standardspeicherkategorie), innerhalb derer sie in den Archivspeicher verschoben werden sollen. Metadatenoperationen wie die Auflistung der Inhalte eines Verzeichnisses zählen nicht als Dateizugriffe.

Typ: Zeichenfolge

Zulässige Werte: `AFTER_1_DAY` | `AFTER_7_DAYS` | `AFTER_14_DAYS` | `AFTER_30_DAYS` | `AFTER_60_DAYS` | `AFTER_90_DAYS` | `AFTER_180_DAYS` | `AFTER_270_DAYS` | `AFTER_365_DAYS`

Erforderlich: Nein

TransitionToIA

Die Anzahl der Tage nach dem letzten Zugriff auf Dateien im Primärspeicher (der Speicherkategorie Standard), nach der sie in den Speicher für seltenen Zugriff (IA) verschoben werden sollen. Metadatenoperationen wie die Auflistung der Inhalte eines Verzeichnisses zählen nicht als Dateizugriffe.

Typ: Zeichenfolge

Zulässige Werte: AFTER_7_DAYS | AFTER_14_DAYS | AFTER_30_DAYS |
AFTER_60_DAYS | AFTER_90_DAYS | AFTER_1_DAY | AFTER_180_DAYS |
AFTER_270_DAYS | AFTER_365_DAYS

Erforderlich: Nein

TransitionToPrimaryStorageClass

Ob Dateien zurück in den primären (Standard-)Speicher verschoben werden sollen, nachdem auf sie im IA- oder Archivspeicher zugegriffen wurde. Metadatenoperationen wie die Auflistung der Inhalte eines Verzeichnisses zählen nicht als Dateizugriffe.

Typ: Zeichenfolge

Zulässige Werte: AFTER_1_ACCESS

Erforderlich: Nein

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen AWS SDKs finden Sie im Folgenden:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

MountTargetDescription

Stellt eine Beschreibung eines Mounting-Ziels bereit.

Inhalt

FileSystemId

Die ID des Dateisystems, für das das Mounting-Ziel bestimmt ist.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Erforderlich: Ja

LifeCycleState

Der Lebenszyklusstatus des Mounting-Ziels.

Typ: Zeichenfolge

Zulässige Werte: `creating | available | updating | deleting | deleted | error`

Erforderlich: Ja

MountTargetId

Vom System zugewiesene ID für das Mounting-Ziel.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 13 Zeichen. Maximale Länge beträgt 45 Zeichen.

Pattern: `^fsmt-[0-9a-f]{8,40}$`

Erforderlich: Ja

SubnetId

Die ID des Subnetzes des Mounting-Ziels.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 15 Zeichen. Maximale Länge beträgt 47 Zeichen.

Pattern: `^subnet-[0-9a-f]{8,40}$`

Erforderlich: Ja

AvailabilityZoneId

Die eindeutige und konsistente Kennung der Availability Zone, in der sich das Mounting-Ziel befindet. `use1-az1` ist beispielsweise eine AZ-ID für die Region `us-east-1` und sie hat in jeder Region den gleichen Standort. AWS-Konto

Typ: Zeichenfolge

Erforderlich: Nein

AvailabilityZoneName

Der Name der Availability Zone, in der sich das Mounting-Ziel befindet. Availability Zones werden den jeweiligen Namen unabhängig voneinander zugeordnet. AWS-Konto Beispielsweise ist die Availability Zone `us-east-1a` für Sie AWS-Konto möglicherweise nicht derselbe Standort wie `us-east-1a` für eine andere AWS-Konto.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 64 Zeichen.

Pattern: `.*`

Erforderlich: Nein

IpAddress

Die IPv4 Adresse für das Mount-Ziel.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 7 Zeichen. Maximale Länge beträgt 15 Zeichen.

Pattern: `^[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}$`

Erforderlich: Nein

Ipv6Address

Die IPv6 Adresse für das Mount-Ziel.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 3. Die maximale Länge beträgt 39.

Erforderlich: Nein

NetworkInterfaceId

Die ID der Netzwerkschnittstelle, die Amazon EFS bei der Erstellung des Mounting-Ziels erstellt hat.

Typ: Zeichenfolge

Erforderlich: Nein

OwnerId

AWS-Konto ID, der die Ressource gehört.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 14 Zeichen.

Pattern: `^(\d{12})|(\d{4}-\d{4}-\d{4})$`

Erforderlich: Nein

VpcId

Die ID der Virtual Private Cloud (VPC), in der das Mounting-Ziel konfiguriert ist.

Typ: Zeichenfolge

Erforderlich: Nein

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen AWS SDKs finden Sie im Folgenden:

- [AWS SDK for C++](#)

- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

PosixUser

Die vollständige POSIX-Identität, einschließlich der Benutzer-ID, Gruppen-ID und jeder sekundären Gruppe, auf dem Access Point IDs, der für alle Dateisystemoperationen verwendet wird, die von NFS-Clients ausgeführt werden, die den Access Point verwenden.

Inhalt

Gid

Die POSIX-Gruppen-ID, die für alle Dateisystemoperationen verwendet wird, die diesen Zugriffspunkt verwenden.

Type: Long

Gültiger Bereich: Mindestwert 0. Maximaler Wert von 4294967295.

Erforderlich: Ja

Uid

Die POSIX-Benutzer-ID, die für alle Dateisystemoperationen verwendet wird, die diesen Zugriffspunkt verwenden.

Type: Long

Gültiger Bereich: Mindestwert 0. Maximaler Wert von 4294967295.

Erforderlich: Ja

SecondaryGids

Sekundäre POSIX-Gruppe, die für alle Dateisystemoperationen IDs verwendet wird, die diesen Access Point verwenden.

Typ: Array von Longs

Array-Mitglieder: Die Mindestanzahl beträgt 0 Elemente. Die maximale Anzahl beträgt 16 Elemente.

Gültiger Bereich: Mindestwert 0. Maximaler Wert von 4294967295.

Erforderlich: Nein

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen finden Sie im AWS SDKs Folgenden:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ReplicationConfigurationDescription

Beschreibt die Replikationskonfiguration für ein bestimmtes Dateisystem.

Inhalt

CreationTime

Der Zeitpunkt der Erstellung der Replikationskonfiguration.

Typ: Zeitstempel

Erforderlich: Ja

Destinations

Ein Array von Zielobjekten. Es wird nur ein Zielobjekt unterstützt.

Typ: Array von [Destination](#)-Objekten

Erforderlich: Ja

OriginalSourceFileSystemArn

Der Amazon-Ressourcenname (ARN) des ursprünglichen EFS-Dateisystems in der Replikationskonfiguration.

Typ: Zeichenfolge

Erforderlich: Ja

SourceFileSystemArn

Der Amazon-Ressourcenname (ARN) des aktuellen EFS-Dateisystems in der Replikationskonfiguration.

Typ: Zeichenfolge

Erforderlich: Ja

SourceFileSystemId

Die ID des Amazon-EFS-Quelldateisystems, das repliziert wird.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 128 Zeichen.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

Erforderlich: Ja

SourceFileSystemRegion

Das, AWS-Region in dem sich das EFS-Quelldateisystem befindet.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 64 Zeichen.

Pattern: `^[a-z]{2}-((iso[a-z]{0,1}-)|(gov-)){0,1}[a-z]+-{0,1}[0-9]{0,1}$`

Erforderlich: Ja

SourceFileSystemOwnerId

ID des, AWS-Konto in dem sich das Quelldateisystem befindet.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 14 Zeichen.

Pattern: `^(\\d{12})|(\\d{4}-\\d{4}-\\d{4}))$`

Erforderlich: Nein

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen finden Sie im AWS SDKs Folgenden:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

ResourceIdPreference

Beschreibt den Ressourcentyp und seine ID-Präferenz für den AWS-Konto Benutzer in der aktuellen Version AWS-Region.

Inhalt

ResourceIdType

Identifiziert die EFS-Ressourcen-ID-Präferenz, entweder LONG_ID (17 Zeichen) oder SHORT_ID (8 Zeichen).

Typ: Zeichenfolge

Zulässige Werte: LONG_ID | SHORT_ID

Erforderlich: Nein

Resources

Identifiziert die Amazon-EFS-Ressourcen, für die die ID-Präferenzeinstellung gilt, FILE_SYSTEM und MOUNT_TARGET.

Typ: Zeichenfolgen-Array

Zulässige Werte: FILE_SYSTEM | MOUNT_TARGET

Erforderlich: Nein

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen AWS SDKs finden Sie im Folgenden:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

RootDirectory

Gibt das Verzeichnis im Amazon-EFS-Dateisystem an, auf das der Zugriffspunkt Zugriff gewährt. Der Zugriffspunkt macht den angegebenen Dateisystempfad als Stammverzeichnis Ihres Dateisystems für Anwendungen verfügbar, die den Zugriffspunkt verwenden. NFS-Clients, die den Access Point verwenden, können nur auf Daten in den Access Points `RootDirectory` und seinen Unterverzeichnissen zugreifen.

Inhalt

CreationInfo

(Optional) Gibt das POSIX IDs und die Berechtigungen an, die für den Access Point gelten sollen. `RootDirectory` Wenn das angegebene `RootDirectory > Path` nicht vorhanden ist, erstellt EFS das Stammverzeichnis mithilfe der `CreationInfo`-Einstellungen, wenn ein Client eine Verbindung zu einem Zugriffspunkt herstellt. Bei der Angabe der `CreationInfo` müssen Sie Werte für alle Eigenschaften angeben.

Important

Wenn Sie `CreationInfo` nicht angeben und das angegebene `RootDirectory > Path` nicht vorhanden ist, schlagen Versuche, das Dateisystem mithilfe des Zugriffspunkts zu mounten, fehl.

Typ: [CreationInfo](#) Objekt

Erforderlich: Nein

Path

Gibt den Pfad auf dem EFS-Dateisystem an, der als Stammverzeichnis für NFS-Clients verfügbar gemacht werden soll, die über den Zugriffspunkt auf das EFS-Dateisystem zugreifen. Ein Pfad kann bis zu vier Unterverzeichnisse haben. Wenn der angegebene Pfad nicht vorhanden ist, müssen Sie die `CreationInfo` angeben.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 1. Maximale Länge beträgt 100 Zeichen.

Pattern: `^(\\|\\(?!\\.)+[^\$#<>;`|&?{}^*/\n]+){1,4}$`

Erforderlich: Nein

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen finden Sie im AWS SDKs Folgenden:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Tag

Ein Tag ist ein Schlüsselwertpaar. Erlaubte Zeichen sind Buchstaben, Leerzeichen und Zahlen, die in UTF-8 dargestellt werden können, sowie die folgenden Zeichen: + - = . _ : /.

Inhalt

Key

Der Tag-Schlüssel (Zeichenfolge). Der Schlüssel darf nicht mit aws : beginnen.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 128 Zeichen.

Pattern: `^(?![aA]{1}[wW]{1}[sS]{1}:)([p{L}\p{Z}\p{N}_.:/=+\-@]+)$`

Erforderlich: Ja

Value

Der Wert des Tag-Schlüssels.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge beträgt 256 Zeichen.

Pattern: `^([p{L}\p{Z}\p{N}_.:/=+\-@]*)$`

Erforderlich: Ja

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einer der sprachspezifischen Sprachen AWS SDKs finden Sie im Folgenden:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

Dokumentverlauf

- API-Version: 2015-02-01
- Letzte Aktualisierung der Dokumentation: 9. Juni 2025

In der folgenden Tabelle sind wichtige Änderungen am Benutzerhandbuch zu Amazon Elastic File System nach Juli 2018 beschrieben. Um Benachrichtigungen über Dokumentationsaktualisierungen zu erhalten, können Sie den RSS-Feed abonnieren.

Änderung	Beschreibung	Datum
Support für IPv6	Internet Protocol Version 6 (IPv6) wird auf Amazon EFS Service APIs - und Mount-Zielen unterstützt. Weitere Informationen zur Amazon EFS-API finden Sie unter Amazon EFS-API . Weitere Informationen zu Bereitstellungszielen finden Sie unter Bereitstellungsziele verwalten .	9. Juni 2025
Elastische Durchsatzsteigerung bei bestimmten AWS-Regionen	Der maximale Lesedurchsatz für Dateisysteme, die Elastic Throughput verwenden, wurde in den folgenden Regionen von 20 GiBps auf 60 erhöht GiBps: Asien-Pazifik (Mumbai), Asien-Pazifik (Seoul), Asien-Pazifik (Sydney) und Europa (London). Weitere Informationen finden Sie unter Amazon EFS-Kontingente, die Sie erhöhen können .	2. Mai 2025

Elastische Durchsatzsteigerung bei bestimmten AWS-Regionen

Der maximale Lesedurchsatz für Dateisysteme, die Elastic Throughput in bestimmten Regionen verwenden, beträgt GiBps 60. In allen anderen Regionen wurde das Limit für den Lesedurchsatz für Dateisysteme, die Elastic Throughput verwenden, von 10 auf 20 erhöht GiBps. Weitere Informationen finden Sie unter [Amazon EFS-Kontingente, die Sie erhöhen können](#).

10. April 2025

Höheres Access Point-Kontingent

Die maximale Anzahl von Zugriffspunkten, die ein einzelnes Dateisystem haben kann, wurde von 1.000 auf 10.000 erhöht. Sie können auch eine Erhöhung dieses Limits beantragen. Weitere Informationen finden Sie unter [Ressourcenkontingente, die Sie erhöhen können](#).

10. Februar 2025

Verbesserte Sicherungs- und Wiederherstellungsraten

Die Geschwindigkeit der Durchführung von Backups und Wiederherstellungen wurde verbessert. Weitere Informationen finden Sie unter [Backup-Leistung](#).

8. Januar 2025

[Höhere IOPS-Quote auf Anfrage](#)

Sie können jetzt erhöhte IOPS für EFS-Dateisysteme im Elastic Throughput-Modus anfordern. Weitere Informationen finden Sie unter [Amazon EFS-Kontingente, die Sie erhöhen können](#).

26. November 2024

[Support kontenübergreifender Replikation](#)

Amazon EFS unterstützt die kontenübergreifende Replikation. Weitere Informationen finden Sie unter [EFS-Dateisysteme replizieren. AWS-Konten](#)

19. November 2024

[Die bestehende AWS verwaltete Richtlinie wurde aktualisiert](#)

ReplicationRead ,ReplicationWrite , iam:PassRole werden der AmazonElasticFileSystemFullAccess Richtlinie hinzugefügt. Weitere Informationen finden Sie unter [AmazonElasticFileSystemFullAccess](#).

7. November 2024

[Die bestehende AWS verwaltete Richtlinie wurde aktualisiert](#)

ReplicationRead und ReplicationWrite hinzugefügt zuAmazonElasticFileSystemServiceRolePolicy . Weitere Informationen finden Sie unter [AmazonElasticFileSystemServiceRolePolicy](#).

7. November 2024

[Die bestehende AWS verwaltete Richtlinie wurde aktualisiert](#)

Die Genehmigung Replicati onRead wurde zur vorhandenen AmazonElasticFileSystemReadOnlyAccess Richtlinie hinzugefügt. Weitere Informationen finden Sie unter [AmazonElasticFileSystemReadOnlyAccess](#).

[Die elastische Durchsatzgrenze wurde erhöht](#)

Das elastische Durchsatzlimit wurde auf 60 Gibibyte pro Sekunde (GiBps) für bestimmte Regionen AWS-Regionen und auf 10 GiBps für alle anderen Regionen erhöht. Weitere Informationen finden Sie unter [Gesamter elastischer Standarddurchsatz für alle verbundenen Clients in jeder Region](#). AWS-Region

14. Oktober 2024

[Die bestehende AWS verwaltete Richtlinie wurde aktualisiert](#)

Das optionale Element Sid (Statement ID) ist jetzt in der AmazonElasticFileSystemReadOnlyAccess Richtlinienerklärung enthalten. Der Wert von Sid ist ElasticFileSystemReadOnlyAccess . Weitere Informationen zum Sid Richtlinienelement finden Sie unter [IAM-JSON-Richtlinienelemente: Sid](#).

7. August 2024

Das elastische Durchsatzlimit wurde erhöht

Die elastische Durchsatzgrenze wurde für bestimmte Fälle erhöht AWS-Regionen. Weitere Informationen finden Sie unter [Gesamter elastischer Standarddurchsatz für alle verbundenen Clients AWS-Region](#).

31. Juli 2024

Die Quote für Bergeziele wurde erhöht

Die maximale Anzahl von Mount-Zielen für jede Virtual Private Cloud (VPC) wurde von 400 auf 1.400 erhöht. Weitere Informationen finden Sie unter [Amazon-EFS-Ressourcenkontingente, die Sie nicht ändern können](#).

15. Mai 2024

Das kombinierte Durchsatzlimit für Elastic-Dateisysteme wurde erhöht

Der maximale kombinierte Lese- und Schreibdurchsatz beträgt 1.500 MiBps für Dateisysteme, die Elastic Throughput verwenden und mit Version 2.0 oder höher des Amazon EFS-Clients (amazon-efs-utils Version) oder des Amazon EFS CSI-Treibers (aws-efs-csi-driver) gemountet wurden. Weitere Informationen finden Sie in der Tabelle mit der Leistungsübersicht unter [Amazon EFS-Leistung](#).

30. April 2024

[Das elastische Durchsatzlimit wurde erhöht](#)

Die elastische Durchsatzgrenze wurde für bestimmte Fälle erhöht AWS-Regionen. Weitere Informationen finden Sie unter [Gesamter elastischer Standarddurchsatz für alle verbundenen Clients AWS-Region.](#)

13. März 2024

[Mehr IOPS](#)

Dateisysteme, die Elastic Throughput verwenden, können maximal 90.000 Lesevorgänge für Daten ermöglichen, auf die selten zugegriffen wird. Weitere Informationen zur Leistung finden Sie unter [Leistungsübersicht.](#)

22. Januar 2024

[Die bestehende AWS verwaltete Richtlinie wurde aktualisiert](#)

Der bestehenden AmazonElasticFileSystemFullAccess Richtlinie wurde eine Berechtigung elasticfilesystem:UpdateFileSystemProtection hinzugefügt, die es Prinzipalen ermöglicht, den Schutz eines Dateisystems zu aktualisieren. Weitere Informationen finden Sie unter [Amazon EFS-Updates für AWS verwaltete Richtlinien.](#)

27. November 2023

[Replizieren in ein vorhandenes Dateisystem](#)

Dateisysteme können jetzt auf bestehende Dateisysteme repliziert werden, was es einfacher macht, Änderungen zwischen Dateisystemen für Failback-Zwecke zu synchronisieren. Weitere Informationen finden Sie unter [Zielfdateisystem](#).

27. November 2023

[Schutz des Dateisystems hinzugefügt](#)

Der Schutz vor dem Überschreiben der Replikation wurde den Dateisystemen hinzugefügt und ist standardmäßig aktiviert. Der Schutz verhindert, dass Dateisysteme als Ziel in einer Replikationskonfiguration verwendet werden. Weitere Informationen finden Sie unter [Schutz des Dateisystems](#).

8. November 2023

Neue Speicherklasse, Dateisystemtypen und Lebenszyklusrichtlinie

Amazon EFS bietet jetzt die Speicherklasse „EFS Archive“, Dateisystemtypen und die Lebenszyklusrichtlinie „Übergang ins Archiv“. Weitere Informationen finden Sie unter [Dateisystemtypen und Speicherklassen](#).

26. November 2023

[Mehr IOPS](#)

Dateisysteme mit Elastic-Dateisatz unterstützen jetzt maximal 65 000 Lese- und 50 000 Schreibvorgänge pro Sekunde für Daten, auf die selten zugegriffen wird, und unterstützen jetzt 250 000 Lese-IOPS für Daten, auf die häufig zugegriffen wird. Weitere Informationen zur Leistung finden Sie unter [Leistungsübersicht](#).

26. November 2023

[Löschen der Replikationskonfiguration aus dem Quelldateisystem](#)

Replikationskonfigurationen können jetzt aus dem Quelldateisystem gelöscht werden. Weitere Informationen finden Sie unter [Löschen einer Replikationskonfiguration](#).

19. September 2023

[Zusätzliche AWS-Region Unterstützung hinzugefügt](#)

Amazon EFS ist jetzt in der Region Israel (Tel Aviv) verfügbar.

7. August 2023

[Leistungssteigerung von Dateisystemen im Allzweckmodus](#)

Amazon-EFS-Dateisysteme im Allzweckmodus unterstützen jetzt bis zu 55 000 Lesevorgänge pro Sekunde und 25 000 Schreiboperationen. Weitere Informationen finden Sie unter [Kontingente für Amazon-EFS-Dateisysteme](#).

3. August 2023

Das Limit für den bereitgestellten Durchsatz wurde erhöht	Das Limit für den bereitgestellten Durchsatz wurde für bestimmte Fälle erhöht. AWS-Regionen Weitere Informationen finden Sie unter Bereitgestellter Standarddurchsatz für alle verbundenen Clients . AWS-Region	21. Juni 2023
Erweiterte Regionsunterstützung für EFS-Replikation	Die EFS-Replikation ist jetzt in allen Bereichen verfügbar , AWS-Regionen in denen EFS verfügbar ist. Weitere Informationen finden Sie unter Amazon-EFS-Replikation .	28. April 2023
Elastische Erhöhung der Durchsatzgrenze	Die elastische Durchsatzgrenze wurde für bestimmte Fälle erhöht AWS-Regionen. Weitere Informationen finden Sie in der Tabelle Gesamter elastischer Standarddurchsatz für alle verbundenen Clients AWS-Region .	17. April 2023
Elastic ersetzt Bursting als Standard-Durchsatzmodus	Der standardmäßige (und empfohlene) Durchsatzmodus für Dateisysteme ist jetzt Elastic statt Bursting. Weitere Informationen finden Sie unter Durchsatzmodi .	13. April 2023
Zusätzliche AWS-Region Unterstützung hinzugefügt	Amazon EFS ist jetzt in der Region Asien-Pazifik (Melbourne) verfügbar.	12. April 2023

Unterstützung für macOS Ventura hinzugefügt	Amazon EFS kann jetzt auf EC2-Mac-Instances installiert werden, die auf macOS Ventura ausgeführt werden. Weitere Informationen finden Sie unter Unterstützte Distributionen .	10. April 2023
Zusätzliche AWS-Region Unterstützung hinzugefügt	Amazon EFS ist jetzt in der Region Asien-Pazifik (Hyderabad) verfügbar.	16. Februar 2023
Zusätzliche AWS-Region Unterstützung hinzugefügt	Amazon EFS ist jetzt für alle Benutzer in der AWS-Region Europa (Spanien) verfügbar.	19. Januar 2023
Das Zugangspunkt-Limit für Dateisysteme wurde erhöht	Die maximale Anzahl von Zugangspunkten, die ein einzelnes Dateisystem haben kann, wurde von 120 auf 1 000 erhöht. Weitere Informationen finden Sie unter Ressourcenkontingente .	17. Januar 2023
Zusätzliche AWS-Region Unterstützung hinzugefügt	Amazon EFS ist jetzt für alle Benutzer in Europa (Zürich) verfügbar AWS-Region.	15. Dezember 2022
Unterstützung für eintägige Lebenszyklusrichtlinien hinzugefügt	Sie können jetzt einen Tag für die Lebenszyklusrichtlinie „Übergang in IA“ auswählen. Weitere Informationen finden Sie unter Using Lifecycle policies .	27. November 2022

[Reduzierte Lese- und Schreiblatenzen](#)

Die Latenzen beim Lesen und Schreiben von Dateidaten haben sich sowohl bei One-Zone-Speicher- als auch bei Standard-Speicher-Dateisystemen verringert. Weitere Informationen zur Leistung finden Sie unter [Leistungsübersicht](#).

27. November 2022

[Zusätzlicher Durchsatzmodus hinzugefügt](#)

Der elastische Durchsatzmodus wurde als Durchsatzoption für Amazon EFS-Dateisysteme hinzugefügt. Weitere Informationen finden Sie unter [Elastischer Durchsatz](#).

27. November 2022

[Zusätzliche AWS-Region Unterstützung hinzugefügt](#)

Amazon EFS ist jetzt für alle Benutzer in der Region Naher Osten (UAE) verfügbar.

17. Oktober 2022

[Unterstützung für EFS-Replikation hinzugefügt](#)

Amazon EFS hat ein früheres Limit entfernt, bei dem die EFS-Replikation keine Sockets und Named Pipes unterstützt, oder FIFOs.

15. September 2022

[Das Limit für die Anzahl der Dateisperren pro Verbindung wurde erhöht](#)

Die Anzahl der Dateisperren pro Verbindung wurde von 8 192 auf 65 536 erhöht. Weitere Informationen finden Sie unter [Kontingente für NFS-Clients](#).

4. Mai 2022

[Das Limit für Prozesse, die Dateisperren verwenden, wurde entfernt](#)

Amazon EFS hat ein früheres Limit aufgehoben, bei dem maximal 256 Prozesse auf einer einzelnen Instanz Dateisperren gleichzeitig verwenden konnten. Weitere Informationen finden Sie unter [Kontingente für NFS-Clients](#).

4. Mai 2022

[Zusätzliche AWS-Region Unterstützung hinzugefügt](#)

Amazon EFS ist jetzt für alle Benutzer in der Region Asien-Pazifik (Jakarta) verfügbar.

27. Januar 2022

[Unterstützung für EFS-Replikation hinzugefügt](#)

Verwenden Sie die EFS-Replikation, um die Daten und Metadaten eines EFS-Dateisystems in ein anderes EFS-Dateisystem Ihrer Wahl zu replizieren. AWS-Region Weitere Informationen finden Sie unter [Amazon EFS - Replikation](#).

25 Januar 2022

[Dateisystem- und Mount-Zielressourcen verwenden das 17-stellige Ressourcen-ID-Format](#)

Dem neuen Amazon EFS-Dateisystem und den Mount-Zielressourcen werden jetzt 17 Zeichen IDs zugewiesen. Weitere Informationen finden Sie unter [Arbeiten mit Amazon EFS-Ressourcen](#).

22. Oktober 2021

[Unterstützung für EFS Intelligent-Tiering hinzugefügt](#)

EFS Intelligent-Tiering verwendet EFS-Lebenszyklusverwaltung zur Überwachung von Dateizugriffsmustern und ist so konzipiert, dass Dateien automatisch zu und von Ihren entsprechenden Speicherklassen mit Infrequent Access (IA) übertragen werden. Weitere Informationen finden Sie unter [EFS Intelligent-Tiering und Lebenszyklusverwaltung](#).

2. September 2021

[Unterstützung für das Testen des 17-stelligen Ressourcen-ID-Formats hinzugefügt](#)

Amazon EFS stellt am 1. Oktober 2021 von der Verwendung von 8 Zeichen IDs auf 17 Zeichen IDs für Dateisysteme und Mount-Ziele um. Während dieser Umstellung können Sie sich anmelden und beginnen, Ressourcen mit 17 Zeichen pro Person IDs zu verwenden. AWS-Region Weitere Informationen finden Sie unter [Ressource IDs](#).

5. Mai 2021

[Unterstützung für das Mounten von One-Zone-Dateisystemen aus einer anderen Availability Zone mit Amazon-EFS-Mountinghilfe hinzugefügt](#)

Sie können jetzt die EFS-Mountinghilfe verwenden, um ein Amazon-EFS-Dateisystem, das One-Zone-Speicherklasse verwendet, für eine EC2-Instance bereitzustellen, die sich in einer anderen Availability Zone befindet. Sie können die neue az-Option verwenden, um die Availability Zone des Amazon-EFS-Dateisystems anzugeben. Weitere Informationen finden Sie unter [Mounting file systems with One Zone storage classes](#).

6. April 2021

[Unterstützung für EFS-One-Zone-Speicherklassen hinzugefügt](#)

Amazon-EFS-One-Zone-Speicherklassen speichern Daten redundant in einer einzigen Availability Zone in einer AWS-Region. Die EFS-Speicherklassen One Zone und One Zone-Infrequent Access (One Zone-IA) sind eine kostengünstige Option zum Speichern von Daten, für die nicht die Multi-AZ-Resilienz der EFS-Speicherklassen Standard und Standard-IA erforderlich ist. Weitere Informationen finden Sie unter [Verwenden von EFS-Speicherklassen](#).

9. März 2021

[Zusätzliche AWS-Region
Unterstützung hinzugefügt](#)

Amazon EFS ist jetzt für alle Benutzer in der AWS-Region Asien-Pazifik (Osaka) verfügbar.

3. März 2021

[Unterstützung für Amazon-EC2-Mac-Instances, auf denen macOS Big Sur ausgeführt wird, hinzugefügt](#)

Sie können Ihr Amazon-EFS-Dateisystem jetzt von EC2-macOS-Instances aus mounten, auf denen macOS Big Sur ausgeführt wird, indem Sie die EFS-Mountinghilfe oder den NFS-Mount-Befehl verwenden. Weitere Informationen finden Sie unter [Mounten mit der EFS-Mountinghilfe](#) oder [Mounten von Dateisystemen ohne die EFS-Mountinghilfe](#).

23. Februar 2021

[Die neue Amazon EFS-Konsole ist in der AWS GovCloud \(US\) Region verfügbar](#)

Die neue Amazon EFS-Konsole ist jetzt in der verfügbar AWS GovCloud (US) AWS-Region.

10. Februar 2021

[Support für neue Amazon CloudWatch EFS-Metriken hinzugefügt MeteredIO Bytes](#)

Sie können MeteredIO Bytes verwenden, um die Bytezahl für jede Dateisystemoperation zu messen, einschließlich Datenlese-, Datenschreib- und Metadatenoperationen. Lesevorgänge werden mit einem Drittel der Rate anderer Vorgänge gemessen. Weitere Informationen finden Sie unter [CloudWatchAmazon-Metriken für Amazon EFS](#).

28. Januar 2021

[Amazon EFS erhöht den Lesedurchsatz im Dateisystem um 300 %](#)

Amazon-EFS-Dateisysteme messen jetzt Leseanforderungen mit einem Drittel der Rate anderer Anforderungen.

28. Januar 2021

[Support für neue Amazon CloudWatch EFS-Metrik hinzugefügt StorageBytes](#)

Sie können StorageBytes verwenden, um die Größe des Dateisystems in Byte zu messen und zu überwachen, einschließlich der Datenmenge, die in den Speicherklassen Standard und Infrequent Access gespeichert ist. Weitere Informationen finden Sie unter [CloudWatch Amazon-Metriken für Amazon EFS](#).

11. Januar 2021

[Wird AWS Transfer Family für den Zugriff auf Amazon EFS-Dateisysteme verwendet](#)

Sie können AWS Transfer Family verwenden, um Dateien in und aus Ihren Amazon EFS-Dateisystemen zu übertragen. Weitere Informationen finden Sie unter [Verwenden für AWS Transfer Family den Zugriff auf Dateien in Ihrem EFS-Dateisystem](#).

06. Januar 2021

[Wird AWS Systems Manager zur Verwaltung des Amazon EFS-Clients \(amazon-efs-utils\) verwendet](#)

Sie können AWS Systems Manager es verwenden, um die Amazon EFS-Clients (amazon-efs-utils) auf Ihren EC2-Instances automatisch zu installieren oder zu aktualisieren. Weitere Informationen finden Sie unter [Verwenden von AWS Systems Manager zur automatischen Installation oder Aktualisierung von Amazon EFS-Clients.](#)

29. September 2020

[Erzwingen der Erstellung verschlüsselter EFS-Dateisysteme](#)

Sie können den Bedingungschlüssel `elasticfilesystem:Encrypted` AWS Identity and Access Management (IAM) verwenden , um zu erzwingen, dass Benutzer Amazon EFS-Dateisysteme erstellen, die im Ruhezustand verschlüsselt sind. Weitere Informationen finden Sie im [Erzwingung der Erstellung von im Ruhezustand verschlüsselten Amazon-EFS-Dateisystemen.](#)

16. September 2020

[Der Durchsatz pro Client von Amazon EFS stieg um 100 %](#)

EFS unterstützt jetzt einen Durchsatz MB/s von bis zu 500 pro Client, was einer Steigerung von 100% gegenüber dem vorherigen Limit von 250 MB/s entspricht. Weitere Informationen finden Sie unter [Kontingente für Amazon-EFS-Dateisysteme](#).

23. Juli 2020

[Unterstützung für automatische tägliche Sicherungen von Amazon-EFS-Dateisystemen](#)

Automatische tägliche Sicherungen sind jetzt standardmäßig aktiviert, wenn Sie mithilfe der EFS-Konsole ein Dateisystem erstellen. Weitere Informationen finden Sie unter [Verwenden AWS Backup mit Amazon EFS](#).

16. Juli 2020

[Der neue Quick-Create-Workflow vereinfacht die Erstellung von Amazon-EFS-Dateisystemen](#)

Mit der Option „Quick Create“ in der EFS-Konsole können Sie mit einer einzigen Schaltfläche ein EFS-Dateisystem mit den vom Service empfohlenen Einstellungen erstellen. Weitere Informationen finden Sie unter [Erstellen Sie Ihr EFS-Dateisystem](#).

16. Juli 2020

[Neue Amazon-EFS-Konsole ist jetzt verfügbar](#)

Die neue EFS-Konsole erleichtert Ihnen die Verwendung von Amazon EFS und vereinfacht die Verwaltung Ihrer EFS-Dateisysteme.

16. Juli 2020

[Amazon EFS erhöht den Mindestdurchsatz im Dateisystem](#)

Amazon EFS-Dateisysteme, die Bursting-Durchsatz verwenden, haben jetzt einen Mindestdurchsatz von 1 MiB/s. Weitere Informationen finden Sie unter [Durchsatzmodi](#).

30. Juni 2020

[Leistungssteigerung von Dateisystemen im Allzweckmodus](#)

Ab sofort unterstützen Amazon-EFS-Dateisysteme im Allzweckmodus bis zu 35 000 Lesevorgänge pro Sekunde. Das entspricht einem Zuwachs von 400 % gegenüber der bisherigen Obergrenze von 7 000 Vorgängen pro Sekunde. Weitere Informationen finden Sie unter [Kontingente für Amazon-EFS-Dateisysteme](#).

01. April 2020

[Zusätzliche Unterstützung hinzugefügt AWS-Region](#)

Amazon EFS ist jetzt für alle Benutzer in Peking und Ningxia AWS-Regionen verfügbar.

22. Januar 2020

[Unterstützung für die IAM-Autorisierung für NFS-Clients hinzugefügt](#)

Sie können jetzt AWS Identity and Access Management (IAM) verwenden, um den NFS-Zugriff auf ein Amazon EFS-Dateisystem zu verwalten. Weitere Informationen finden Sie unter [Verwenden von AWS IAM zur Steuerung des NFS-Zugriffs auf Amazon EFS](#).

13. Januar 2020

[Unterstützung für EFS-Zugriffspunkte hinzugefügt](#)

Amazon-EFS-Zugangspunkte sind anwendungsspezifische Einstiegspunkte in ein EFS-Dateisystem, die das Verwalten der Anwendungszugriffe auf freigegebene Datensätze erleichtern. Weitere Informationen finden Sie unter [Arbeiten mit Amazon EFS Access Points](#).

13. Januar 2020

[Support für die AWS Backup teilweise Wiederherstellung hinzugefügt.](#)

Sie können jetzt bestimmte Dateien und Verzeichnisse mithilfe einer Teilwiederherstellung wiederherstellen, zusätzlich zum Wiederherstellen eines vollständigen Wiederherstellungspunkts. Weitere Informationen finden Sie unter [Verwenden AWS Backup mit Amazon EFS](#).

13. Januar 2020

[Unterstützung für serviceverknüpfte IAM-Rollen hinzugefügt](#)

Amazon EFS verwendet jetzt eine serviceverknüpfte Rolle basierend auf IAM, die das Einrichten von EFS erleichtert, indem automatisch die erforderlichen Berechtigungen hinzugefügt werden. Weitere Informationen finden Sie unter [Verwenden von serviceverknüpften Rollen für Amazon EFS](#).

10. Dezember 2019

[Zusätzliche AWS-Region Unterstützung hinzugefügt](#)

Amazon EFS ist jetzt für alle Benutzer in Europa (Stockholm) verfügbar AWS-Region.

20. November 2019

Zusätzliche AWS-Region Unterstützung hinzugefügt	Amazon EFS ist jetzt für alle Benutzer im asiatisch-pazifischen Raum (Hongkong) verfügbar AWS-Region.	20. November 2019
Zusätzliche AWS-Region Unterstützung hinzugefügt	Amazon EFS ist jetzt für alle Benutzer in Südamerika (São Paulo) verfügbar AWS-Region.	20. November 2019
Zusätzliche AWS-Region Unterstützung hinzugefügt	Amazon EFS ist jetzt für alle Benutzer im Nahen Osten (Bahrain) verfügbar AWS-Region.	20. November 2019
Neue Lebenszyklusmanagement-Richtlinie nach 7 Tagen hinzugefügt	Das Lebenszyklusmanagement verfügt jetzt über eine zusätzliche Richtlinie, um Daten nach 7 Tagen in die kostengünstige Speicherkategorie „Infrequent Access“ zu verschieben. Weitere Informationen hierzu finden Sie unter EFS-Lebenszyklusverwaltung .	6. November 2019
Unterstützung für Schnittstellen-VPC-Endpunkte hinzugefügt	Sie können eine private Verbindung zwischen Ihrer Virtual Private Cloud und Amazon EFS herstellen, um die EFS-API aufzurufen. Weitere Informationen finden Sie unter Arbeiten mit VPC-Endpunkten .	22. Oktober 2019

[Mounten Sie ein EFS-Dateisystem beim Starten einer neuen EC2-Instance.](#)

Sie können jetzt neue Amazon-EC2-Instances so konfigurieren, dass Ihre EFS-Dateisysteme beim Start im EC2 Launch Instance Wizard gemountet werden. Weitere Informationen finden Sie in [Schritt 2. Erstellen Sie Ihre EC2-Ressourcen und starten Sie Ihre EC2-Instance.](#)

17. Oktober 2019

[Unterstützung für Service-Kontingente hinzugefügt](#)

Sie können jetzt alle Amazon-EFS-Limits in der Konsole für Servicekontingente anzeigen. Weitere Informationen finden Sie unter [Amazon-EFS-Limits.](#)

10. September 2019

[Neue Richtlinien für die Lebenszyklusverwaltung hinzugefügt](#)

Bei der Lebenszyklusverwaltung können Sie nun eine von vier Lebenszyklusrichtlinien auswählen, um festzulegen, wann Dateien in die kostengünstige Infrequent Access-Speicherklasse übertragen werden. Weitere Informationen hierzu finden Sie unter [EFS-Lebenszyklusverwaltung.](#)

9. Juli 2019

[Die EFS-Lebenszyklusverwaltung ist jetzt auf allen EFS-Dateisystemen verfügbar.](#)

Die Funktion für die EFS-Lebenszyklusverwaltung ist jetzt auf allen EFS-Dateisystemen verfügbar. Eine frühere Einschränkung basierend auf der Erstellung des Dateisystems wurde entfernt. Weitere Informationen hierzu finden Sie unter [EFS-Lebenszyklusverwaltung](#).

9. Juli 2019

[Zusätzliche AWS-Region Unterstützung hinzugefügt](#)

Amazon EFS ist jetzt für alle Benutzer in Europa (Paris) verfügbar AWS-Region.

12. Juni 2019

[Zusätzliche AWS-Region Unterstützung hinzugefügt](#)

Amazon EFS ist jetzt für alle Benutzer im asiatisch-pazifischen Raum (Mumbai) verfügbar AWS-Region.

5. Juni 2019

[Zusätzliche AWS-Region Unterstützung hinzugefügt](#)

Amazon EFS ist jetzt für alle Benutzer in Kanada (Central) verfügbar AWS-Region.

1. Mai 2019

[API-Update: Tags sind jetzt Teil der CreateFileSystem Operations-Payload](#)

Sie können jetzt Tags hinzufügen, wenn Sie die AWS API- und CreateFileSystem CLI-Operation verwenden, um ein Amazon EFS-Dateisystem zu erstellen. Weitere Informationen finden Sie unter [CreateFileSystem](#) und [Erstellen eines Dateisystems mit der AWS CLI](#).

19. Februar 2019

[Neue Features: EFS Infrequent Access-Speicherklasse und EFS-Lebenszyklusverwaltung](#)

Amazon EFS Infrequent Access ist eine kostenoptimierte Speicherklasse für selten aufgerufene Dateien. Die EFS-Lebenszyklusverwaltung übergibt Dateien automatisch vom Standard in den Infrequent Access-Speicher. Weitere Informationen finden Sie unter [EFS-Speicherklassen](#).

13. Februar 2019

[Zusätzliche AWS-Region Unterstützung hinzugefügt](#)

Amazon EFS ist jetzt für alle Benutzer in Europa (London) verfügbar AWS-Region.

23. Januar 2019

[AWS Backup Serviceintegration mit Amazon EFS](#)

Amazon EFS-Dateisysteme können mit einem vollständig verwalteten AWS Backup, zentralisierten und automatisierten Backup-Service für die Sicherung von Daten zwischen AWS Services in der Cloud und vor Ort gesichert werden. Weitere Informationen finden Sie unter [AWS Backup und Amazon EFS](#).

16. Januar 2019

[Unterstützung für eine Transit-Gateway-Verbindung zu lokalen Speichersystemen hinzugefügt.](#)

Amazon-EFS-Dateisysteme sind nun über Transit Gateway-Verbindungen zu lokalen Speichersystemen zugänglich. Weitere Informationen finden Sie unter [Mounten von einem anderen Konto oder VPC](#) und [Exemplarische Vorgehensweise: Mounten eines Dateisystems aus einer anderen VPC.](#)

6. Dezember 2018

[EFS File Sync ist jetzt Teil des neuen AWS DataSync Dienstes.](#)

AWS DataSync ist ein verwalteter Datenübertragungsdienst, der die Synchronisierung großer Datenmengen zwischen lokalen Speichersystemen und AWS Speicherdiensten vereinfacht. Weitere Informationen finden Sie unter [Übertragen von Dateien von lokalen Dateisystemen zu Amazon EFS mithilfe von AWS DataSync.](#)

26. November 2018

[Unterstützung für VPN- und interregionale VPC Peering-Verbindung hinzugefügt](#)

Amazon EFS sind nun über VPN-Verbindungen und interregionale VPC-Peering-Verbindungen zugänglich. Weitere Informationen finden Sie unter [Übertragen von Dateien von lokalen Dateisystemen zu Amazon EFS mithilfe von AWS DataSync.](#)

23. Oktober 2018

<u>Unterstützung für VPN- und interregionale VPC Peering-Verbindung hinzugefügt</u>	Amazon-EFS-Dateisysteme sind nun über VPN-Verbindungen und interregionale VPC-Peering-Verbindungen zugänglich. Weitere Informationen finden Sie unter <u>Mounten von einem anderen Konto oder einer anderen VPC</u> aus und <u>So funktioniert Amazon EFS mit Direct Connect und VPNs</u> .	23. Oktober 2018
<u>Zusätzliche AWS-Region Unterstützung hinzugefügt</u>	Amazon EFS ist jetzt für alle Benutzer in der AWS-Region Asien-Pazifik (Singapur) verfügbar.	13. Juli 2018
<u>Einführung in den Durchsatzmodus „Bereitgestellt“</u>	Sie können Durchsatz für neue oder vorhandene Dateisysteme nun mit dem Durchsatzmodus „Bereitgestellt“ bereitstellen. Weitere Informationen finden Sie unter <u>Durchsatzmodi</u> .	12. Juli 2018
<u>Zusätzliche AWS-Region Unterstützung hinzugefügt</u>	Amazon EFS ist jetzt für alle Benutzer in der AWS-Region Asien-Pazifik (Tokio) verfügbar.	11. Juli 2018

In der folgenden Tabelle werden wichtige Änderungen am Benutzerhandbuch zu Amazon Elastic File System vor Juli 2018 beschrieben.

Änderungen	Beschreibung	Änderungsdatum
Zusätzliche AWS-Region Unterstützung hinzugefügt	Amazon EFS ist jetzt in der AWS Region Asien-Pazifik (Seoul) verfügbar.	30. Mai 2018
Unterstützung für CloudWatch metrische Mathematik hinzugefügt	Mit metrischer Mathematik können Sie mehrere CloudWatch Metriken abfragen und mithilfe mathematischer Ausdrücke neue Zeitreihen auf der Grundlage dieser Metriken erstellen. Weitere Informationen finden Sie unter Verwenden von metrischer Mathematik mit CloudWatch Metriken .	4. April 2018
Die amazon-efs-utils - Reihe von Open-Source-Tools und Verschlüsselung bei der Übertragung hinzugefügt	Die amazon-efs-utils -Tools sind eine Reihe von ausführbaren Open-Source-Dateien, die Aspekte der Verwendung von Amazon EFS, wie etwa das Mounten, vereinfachen. Für die Nutzung amazon-efs-utils fallen keine zusätzlichen Kosten an und Sie können diese Tools von heruntergeladenen GitHub. Weitere Informationen finden Sie unter Den Amazon EFS-Client installieren. Außerdem unterstützt Amazon EFS in dieser Version jetzt die Verschlüsselung bei der Übertragung über Transport Layer Security (TLS)-Tunneling. Weitere Informationen finden Sie unter Datenverschlüsselung in Amazon EFS.	4. April 2018
Aktualisierte Dateisystemgrenzwerte pro AWS-Region	Amazon EFS hat die Beschränkung für die Anzahl der Dateisysteme für alle Konten in allen AWS-Regionen erhöht. Weitere Informationen finden Sie unter Amazon-EFS-Ressourcenkontingente, die Sie nicht ändern können .	15. März 2018
Zusätzliche AWS-Region Unterstützung hinzugefügt	Amazon EFS ist jetzt für alle Benutzer im Westen der USA (Nordkalifornien) verfügbar AWS-Region.	14. März 2018

Änderungen	Beschreibung	Änderungsdatum
Datenverschlüsselung im Ruhezustand	Amazon EFS unterstützt nun die Verschlüsselung gespeicherter Daten. Weitere Informationen finden Sie unter Datenverschlüsselung in Amazon EFS .	14. August 2017
Unterstützung für zusätzliche Region hinzugefügt	Amazon EFS ist jetzt für alle Benutzer in der Region Europa (Frankfurt) verfügbar.	20. Juli 2017
Dateisystemnamen unter Verwendung von DNS (Domain Name System)	Amazon EFS unterstützt jetzt DNS-Namen für Dateisysteme. Ein DNS-Name eines Dateisystems wird automatisch in die IP-Adresse eines Mount-Ziels in der Availability Zone für die die Verbindung herstellende Amazon-EC2-Instance aufgelöst. Weitere Informationen finden Sie unter Mounting auf Amazon EC2 mit einem DNS-Namen .	20. Dezember 2016
Stärkere Tag-Unterstützung für Dateisysteme	Amazon EFS unterstützt jetzt 50 Tags pro Dateisystem. Weitere Informationen zu Tags in Amazon EFS finden Sie im Abschnitt Taggen von EFS-Ressourcen .	29. August 2016
Allgemeine Verfügbarkeit	Amazon EFS ist jetzt allgemein in den Regionen USA Ost (Nord-Virginia), USA West (Oregon), Asien-Pazifik (Tokio) und Europa (Irland) verfügbar.	28. Juni 2016
Erhöhung des Dateisystemlimits	Die Anzahl der Amazon-EFS-Dateisysteme, die pro Konto pro AWS-Region erstellt werden können, wurde von 5 auf 10 erhöht.	21. August 2015
Aktualisierte Übung „Erste Schritte“	Die Übung „Erste Schritte“ wurde aktualisiert, um den Einstieg zu erleichtern.	17. August 2015
Neues Handbuch	Dies ist die erste Version des Benutzerhandbuchs zu Amazon Elastic File System.	26. Mai 2015

Die vorliegende Übersetzung wurde maschinell erstellt. Im Falle eines Konflikts oder eines Widerspruchs zwischen dieser übersetzten Fassung und der englischen Fassung (einschließlich infolge von Verzögerungen bei der Übersetzung) ist die englische Fassung maßgeblich.