



Leitfaden

Amazon Elastic File System



Amazon Elastic File System: Leitfaden

Copyright © 2024 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 Ersthelfer von Amazon EFS?	3
Funktionsweise	4
Übersicht	4
So funktioniert Amazon EFS mit Amazon EC2	6
Amazon EFS-Dateisysteme	6
Amazon EFS One Zone-Dateisysteme	7
So funktioniert AWS Direct Connect mit AWS und AWS Managed VPN	9
Wie Amazon EFS mit AWS Backup funktioniert	10
Übersicht über die Implementierung	11
Authentifizierung und Zugriffskontrolle	13
Datenkonsistenz in Amazon EFS	13
Sperrungen von Dateien	13
EFS-Speicherklassen	14
Lebenszyklusmanagement	14
EFS-Replikation	14
Einrichten	16
So melden Sie sich für ein AWS-Konto an	16
Einen Administratorbenutzer erstellen	16
Erste Schritte	18
Annahmen	18
Verwandte Themen	19
Schritt 1: Erstellen Ihres Dateisystems	19
Schritt 2: Erstellen Sie Ihre EC2-Ressourcen und starten Sie eine Instance	21
Schritt 3: Übertragen von Dateien mit DataSync	22
Bevor Sie beginnen	23
Schritt 4: Bereinigen von Ressourcen	24
Dateisystemtypen und Speicherklassen	26
EFS-Dateisystemtypen	26
Unterstützte Availability Zones für One Zone-Dateisysteme	27
EFS-Speicherklassen	29
Optimierung der Speicherkosten	30
Vergleich der Speicherklassen	30
Preisgestaltung der Speicherklassen	31

Anzeigen der Größe der Speicherklasse	32
Arbeiten mit EFS-Ressourcen	35
Ressourcen-IDs	36
Erstellen eines Dateisystems	37
Voraussetzungen	37
Konfigurationsoptionen	38
Erstellen eines Dateisystems mit der Konsole	41
Erstellen eines Dateisystems mithilfe der AWS CLI	46
Löschen eines Dateisystems	49
Verwenden der Konsole	50
Verwenden der -CLI	50
Mount-Ziele und Sicherheitsgruppen	50
Erstellen von Sicherheitsgruppen	58
Erstellen von Sicherheitsgruppen mit der AWS Management Console	59
Erstellen von Sicherheitsgruppen mit der AWS CLI	61
Erstellen von Dateisystemrichtlinien	61
Erstellen und Löschen von Zugangspunkten	64
Löschen eines Zugriffspunkts	69
Markieren der Amazon-EFS-Ressourcen	70
Grundlagen zu Tags (Markierungen)	70
Markieren Ihrer -Ressourcen mit Tags (Markierungen)	71
Tag (Markierung)-Einschränkungen	71
Verwenden von Tags für die Zugriffskontrolle	72
Dateisysteme verwenden	73
Verwandte Themen	73
Verwenden von amazon-efs-utils	74
Übersicht	74
Unterstützte Distributionen	76
So installieren Sie amazon-efs-utils mit AWS Systems Manager	77
Was macht der Amazon EFS-Client während der Installation	77
Von Systems Manager Distributor unterstützte Betriebssysteme	78
So verwenden Sie , AWS Systems Manager um automatisch zu installieren oder zu aktualisieren amazon-efs-utils	78
Manuelles Installieren des Amazon EFS-Clients	80
Installation des Amazon EFS-Clients auf Amazon Linux und Amazon Linux 2	81
Installation des Amazon EFS-Clients auf anderen Linux-Distributionen	82

Installation des EFS-Clients auf EC2-Mac-Instances	87
Installation von botocore	89
Upgraden von botocore	91
Upgraden von stunnel	91
Deaktivieren der Überprüfung des Hostnamens des Zertifikats	93
Aktivieren des Online Certificate Status Protocol	94
Mounting von Dateisystemen	95
Verwenden der EFS-Mountinghilfe	96
Funktionsweise	97
Abrufen von Support-Protokollen	99
Voraussetzungen	100
Mounting auf EC2-Linux	101
Mounting auf EC2 Mac	103
Mounting aus einer anderen Region	105
Mounting von One-Zone-Dateisystemen	106
Mounting mit IAM-Autorisierung	110
Mounting mit EFS-Zugangspunkten	111
Mounting mit On-Premises-Clients	112
Automatisches Mounting von EFS	113
Mounting auf mehreren EC2-Instances	123
Mounting von einem anderen Konto oder einer anderen VPC	124
Zusätzliche Überlegungen zum Mounting	128
Aufheben des Mountings von Dateisystemen	129
Fehlerbehebung für AMI- und Kernel-Versionen	130
Übertragung von Daten	132
AWS DataSyncZur Übertragung von Daten in Amazon EFS verwenden	132
Verwendung AWS Transfer Family mit Amazon EFS	133
Voraussetzungen für die Verwendung AWS Transfer Family mit Amazon EFS	134
Konfiguration Ihres Amazon EFS-Dateisystems für die Verwendung mit AWS Transfer Family	134
Verwalten von Dateisystemen	140
Verwalten der Mountingziele	141
Erstellen oder Löschen von Mountingzielen in einer VPC	143
Ändern der VPC für Ihr Mounting-Ziel	144
Aktualisieren der Konfiguration von Mountingzielen	145
Verwalten des Durchsatzes	146

Verwaltung des Dateisystemspeichers	148
Lebenszyklus-Richtlinien	148
Dateisystemoperationen für die Lebenszyklusverwaltung	149
Verwaltung von Lebenszyklusrichtlinien für ein Dateisystem	149
Zugriffsverwaltung auf verschlüsselte Dateisysteme	153
Ausführen von administrativen Aktionen für Amazon-EFS-KMS-Schlüssel	154
Verwandte Themen	155
Messen eines Dateisystems	155
Messen von Objekten	155
Gemessene Größe eines Dateisystems	157
Mess-Durchsatz	159
Verwalten der Dateisystemkosten mitAWSBudgets	159
Voraussetzungen	160
Erstellen eines monatlichen Kostenbudgets für ein EFS-Dateisystem	160
Status des Dateisystems	161
Überwachung von EFS	163
Überwachungstools	164
Automatisierte Tools	164
Manuelle Überwachungstools	165
Überwachung mit CloudWatch	165
Amazon- CloudWatch Metriken für Amazon EFS	166
Wie verwende ich die Amazon-EFS-Metriken?	172
Verwenden von Metrikberechnungen mit Amazon EFS	173
Überwachung des Erfolgs- oder Fehlerstatus des Mount-Versuchs	179
Zugreifen auf CloudWatch Metriken	181
Erstellen von Alarmen	183
Protokollieren von Amazon EFS-API-Aufrufen mit AWS CloudTrail	185
Amazon EFS-Informationen in CloudTrail	185
Grundlegendes zu Amazon EFS-Protokolldateieinträgen	186
Amazon EFS-Protokolldateieinträge für encrypted-at-rest Dateisysteme	193
Leistung	195
Zusammenfassung der Leistung	195
Speicherklassen	197
Leistungsmodi	197
Durchsatzmodi	198
Auswählen eines Durchsatzmodus	198

Elastischer Durchsatz	199
Bereitgestellter Durchsatz	200
Einschränkungen beim Umschalten des Durchsatzes und Ändern der bereitgestellten Menge	203
Tipps zur Leistung	203
Durchschnittliche E/A-Größe	203
Optimierung von Workloads, die einen hohen Durchsatz und IOPS erfordern	203
Gleichzeitige Verbindungen	204
Anforderungsmodell	204
NFS-Client-Mount-Einstellungen	205
Optimierung der Leistung kleiner Dateien	205
Optimieren der Verzeichnisleistung	206
Optimierung der NFS-Größe von <code>read_ahead_kb</code>	206
Sichern von Dateisystemen	208
Inkrementelle Sicherungen	208
Backup-Konsistenz	209
Backup-Leistung	209
Zeitfenster für den Abschluss der Sicherung	209
EFS-Speicherklassen	210
IAM-Berechtigungen zum Erstellen und Wiederherstellen von Sicherungen	210
On-Demand-Backups	210
Gleichzeitige Sicherungen	210
Automatische Sicherungen	211
Ein- oder Ausschalten automatischer Sicherungen für bestehende Dateisysteme	211
Manuelle Konfiguration von Sicherungen	213
Wiederherstellen eines Wiederherstellungspunkts	214
Löschen eines Backups	215
Replizieren von Dateisystemen	216
Replikationskonfiguration	217
Replizieren in ein neues Dateisystem	217
Replizieren in ein vorhandenes Dateisystem	218
Schutz des Dateisystems	219
Erforderliche Berechtigungen	220
Kosten	221
Leistung	221
Mounten eines Zieldateisystems	222

Failover und Failback des Dateisystems	222
Erstellen von Replikationskonfigurationen	223
Anzeigen von Replikationskonfigurationen	226
Löschen von Replikationskonfigurationen	229
Überwachung des Replikationsstatus	230
Anleitungen	233
Exemplarische Anleitung: Erstellen und mounten Sie ein Dateisystem mit derAWS CLI	233
Bevor Sie beginnen	234
Einrichten von AWS CLI	235
Schritt 1: Erstellen Amazon EC2 EC2-Ressourcen	236
Schritt 2: Erstellen von Amazon EFS-Ressourcen	242
Schritt 3: Mounting und Testen des -Dateisystems	246
Schritt 4: Bereinigen	250
Komplettlösung: Einrichten eines Apache-Web-Servers und Bereitstellen einer Verbindung mit Ihrer Linux-Umgebung	251
Einzelne EC2-Instance, die Dateien bereitstellt	252
Mehrere EC2-Instanzen, die Dateien bereitstellen	255
Komplettlösung: Schreibbare Unterverzeichnisse pro Benutzer erstellen	260
Automatische Remountion beim Neustart	262
Exemplarische Vorgehensweise: Bereitstellen von EFS auf einem lokalen Client	262
Bevor Sie beginnen	264
Schritt 1: Erstellen Sie Ihre Amazon Elastic File System-Ressourcen	265
Schritt 2: Installieren Sie den NFS-Client	267
Schritt 3: Mounten Sie das Amazon EFS-Dateisystem auf Ihrem lokalen Client	267
Schritt 4: Bereinigen Sie Ressourcen und schützen Sie Ihr AWS Konto	269
Optional: Verschlüsseln von Daten während der Übertragung	270
Exemplarische Vorgehensweise: Mounten eines Dateisystems aus einer anderen VPC	273
Bevor Sie beginnen	274
Schritt 1: Ermitteln der Availability Zone-ID des EFS-Mounting-Ziels	275
Schritt 2: Bestimmen der IP-Adresse des Mounting-Ziels	276
Schritt 3: Hinzufügen eines Hosteintrags für das Mounting-Ziel	277
Schritt 4: Mounting des Dateisystems mithilfe der EFS-Mountinghilfe	277
Schritt 5: Bereinigen Sie Ressourcen und schützen Sie IhrAWS Konto	279
Exemplarische Anleitung: Erzwingen der Verschlüsselung auf einem Amazon EFS-Dateisystem im Ruhezustand	280
Erzwingen von Verschlüsselung im Ruhezustand	281

Root-Squashing mit IAM für NFS aktivieren	284
Sicherheit	288
Datenverschlüsselung in Amazon EFS	289
Verwendung von Verschlüsselung	289
Verschlüsseln von Daten im Ruhezustand	290
Verschlüsseln von Daten während der Übertragung	296
Identity and Access Management	298
Zielgruppe	299
Authentifizierung mit Identitäten	299
Verwalten des Zugriffs mit Richtlinien	303
So funktioniert Amazon Elastic File System mit IAM	306
Beispiele für identitätsbasierte Richtlinien	314
Beispiele für eine ressourcenbasierte Richtlinie	319
Von AWS verwaltete Richtlinien	322
Verwenden von Tags mit Amazon EFS	329
Verwendung von serviceverknüpften Rollen für Amazon EFS	333
Fehlerbehebung	338
Steuern des Datenzugriffs auf das Dateisystem	340
Standard-Dateisystemrichtlinie	341
EFS-Aktionen für Clients	341
EFS-Bedingungsschlüssel für Clients	341
Beispiele für Dateisystemrichtlinien	342
Kontrolle des Netzwerkzugriffs	342
Verwendung von VPC-Sicherheitsgruppen für Amazon EC2-Instance und Mounting-Ziele ...	343
Quell-Ports	344
Sicherheitsüberlegungen für den Netzwerkzugriff	345
Arbeiten mit VPC-Endpunkten	346
Benutzer, Gruppen und Berechtigungen auf NFS-Ebene	348
Datei- und Verzeichnisberechtigungen	349
Beispiel für Amazon EFS-Dateisystem-Nutzungsfälle und Berechtigungen	349
Benutzer- und Gruppen-ID-Berechtigungen für Dateien und Verzeichnisse in einem Dateisystem	351
Kein Root-Squashing	352
Zwischenspeichern von Berechtigungen	353
Ändern des Besitzes an Dateisystemobjekten	353
EFS-Zugangspunkte	353

Arbeiten mit Zugriffspunkten	353
Erstellen eines Zugriffspunkts	354
Montage mit Access Points	354
Erzwingen einer Benutzeridentität	355
Erzwingen eines Stammverzeichnisses erzwingen einer Stammliste	356
Verwenden von Access Points in IAM-Richtlinien	358
Blockieren des öffentlichen Zugriffs	359
Blockieren des öffentlichen ZugriffsAWS Transfer Family	359
Die Bedeutung von „öffentlich“	360
Compliance-Validierung	362
Ausfallsicherheit	363
Netzwerkisolierung	364
EFS-Kontingente	365
Amazon-EFS-Kontingente, die Sie erhöhen können	365
Beantragen einer Kontingenterhöhung	367
Amazon-EFS-Ressourcenkontingente, die Sie nicht ändern können	367
Kontingente für NFS-Clients	369
Kontingente für Amazon-EFS-Dateisysteme	369
Nicht unterstützte Funktionen von NFSv4.0 und 4.1	370
Weitere Überlegungen	372
Fehlerbehebung bei Amazon EFS	373
Fehlerbehebung bei allgemeinen Problemen	373
Ein EFS-Dateisystem kann nicht erstellt werden	374
Zugriff auf zulässige Dateien im NFS-Dateisystem verweigert	374
Fehler beim Zugriff auf die Amazon EFS-Konsole	375
Amazon EC2-Instance hängt sich auf	375
Anwendung, die große Datenmengen schreibt, bleibt hängen	376
Schlechte Leistung beim Öffnen vieler Dateien gleichzeitig	376
Benutzerdefinierte NFS-Einstellungen verursachen Schreibverzögerungen	377
Die Erstellung von Sicherungen mit Oracle Recovery Manager ist langsam	378
Fehlerbehebung bei Fehlern mit Dateivorgängen	378
Der Befehl schlägt mit dem Fehler „Disk quota exceeded“ fehl	379
Befehl schlägt mit „E/A-Fehler“ fehl	379
Befehl schlägt mit der Fehlermeldung „Dateiname ist zu lang“ fehl	380
Befehl schlägt fehl mit dem Fehler „Datei nicht gefunden“	380
Befehl schlägt mit der Fehlermeldung „Zu viele Links“ fehl	380

Befehl schlägt mit der Fehlermeldung „Datei zu groß“ fehl	381
Beheben von AMI- und Kernel-Problemen	381
Eigentümerschaft kann nicht geändert werden	381
Aufgrund des Client-Bug wiederholt das Dateisystem Vorgänge immer wieder	382
Blockierter Client	382
Das Auflisten von Dateien in einem großen Verzeichnis dauert zu lange	382
Beheben von Mountingproblemen	383
Das Dateisystem-Mounting auf der Windows Instance schlägt fehl	383
Zugriff vom Server verweigert	384
Automatisches Mounting schlägt fehl und die Instance reagiert nicht	384
Mounting mehrerer Amazon EFS-Dateisysteme in /etc/fstab schlägt fehl	384
Mounting-Befehl schlägt mit der Fehlermeldung „falscher fs-Typ“ fehl	386
Der Mounting-Befehl schlägt mit der Fehlermeldung „Inkorrekte Mounting-Option“ fehl	386
Mounting mit Zugangspunkt schlägt fehl	387
Das Mounting des Dateisystems schlägt sofort nach der Erstellung des Dateisystems fehl ..	387
Das Mounting des Dateisystems hängt und schlägt dann mit einem Timeout-Fehler fehl	387
Mounting eines Dateisystems mit NFS unter Verwendung eines DNS-Namens schlägt fehl ..	388
Das Mounting des Dateisystems schlägt mit der Fehlermeldung „nfs reagiert nicht“	389
Der Lebenszyklusstatus des Mounting-Ziels hängt fest	390
Der Lebenszyklusstatus des Mounting-Ziels zeigt einen Fehler an	390
Mounting reagiert nicht	390
Gemounteter Client wird nicht mehr verbunden	391
Operationen auf einem neu gemounteten Dateisystem geben den Fehler „bad file handle“ zurück	392
Unmounten eines Dateisystems schlägt fehl	392
Fehlerbehebung bei der Verschlüsselung	393
Mounting mit Verschlüsselung der Daten während der Übertragung schlägt fehl	393
Mounting mit Verschlüsselung der Daten während der Übertragung wird unterbrochen	393
Ein ncrypted-at-rest Dateisystem kann nicht erstellt werden	394
Nicht verwendbares verschlüsseltes Dateisystem	394
Amazon-EFS-API	396
API-Endpunkt	396
API-Version	397
Verwandte Themen	397
Arbeiten mit der Abfrage-API-Anforderungsrate für Amazon EFS	397
Abrufen	398

Wiederholungsversuche oder Stapelverarbeitung	398
Berechnen des Schlafintervalls	398
Aktionen	398
CreateAccessPoint	401
CreateFileSystem	409
CreateMountTarget	425
CreateReplicationConfiguration	437
CreateTags	444
DeleteAccessPoint	447
DeleteFileSystem	449
DeleteFileSystemPolicy	453
DeleteMountTarget	456
DeleteReplicationConfiguration	460
DeleteTags	463
DescribeAccessPoints	466
DescribeAccountPreferences	471
DescribeBackupPolicy	474
DescribeFileSystemPolicy	477
DescribeFileSystems	481
DescribeLifecycleConfiguration	487
DescribeMountTargets	491
DescribeMountTargetSecurityGroups	497
DescribeReplicationConfigurations	501
DescribeTags	505
ListTagsForResource	510
ModifyMountTargetSecurityGroups	514
PutAccountPreferences	518
PutBackupPolicy	521
PutFileSystemPolicy	524
PutLifecycleConfiguration	530
TagResource	539
UntagResource	543
UpdateFileSystem	546
UpdateFileSystemProtection	554
Datentypen	558
AccessPointDescription	559

BackupPolicy	562
CreationInfo	563
Destination	565
DestinationToCreate	567
FileSystemDescription	570
FileSystemProtectionDescription	575
FileSystemSize	576
LifecyclePolicy	578
MountTargetDescription	580
PosixUser	583
ReplicationConfigurationDescription	585
ResourceIdPreference	587
RootDirectory	588
Tag	590
Zusätzliche Informationen	591
Sichern mit AWS Data Pipeline	591
Leistung für Amazon-EFS-Backups mit AWS Data Pipeline	592
Überlegungen zu Amazon-EFS-Sicherungen mithilfe von AWS Data Pipeline	593
Annahmen für Amazon-EFS-Backup mit AWS Data Pipeline	594
So sichern Sie ein Amazon-EFS-Dateisystem mit AWS Data Pipeline	595
Weitere Sicherungsressourcen	603
Mounten von Dateisystemen ohne die EFS-Mountinghilfe	610
NFS-Support	611
Installieren des NFS-Clients	612
NFS-Mounting-Optionen	615
Mounting auf Amazon EC2 mit einem DNS-Namen	617
Mounting mit einer IP-Adresse	620
Dokumentverlauf	623
.....	dcxlvii

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 Petabyte skaliert werden kann. Es kann durch hinzufügen oder entfernen von Daten vergrößert oder verkleinert werden, ohne andere Softwareanwendungen zu stören. Da Amazon EFS über eine einfache Web-Service-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 Protokoll Network File System Version 4 (NFSv4.1 und NFSv4.0), so dass Ihre Anwendungen und Tools nahtlos mit Amazon EFS zusammenarbeiten. Amazon EFS ist für die meisten Arten von Amazon Web Services-Compute-Instances zugänglich, einschließlich Amazon EC2, 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 über mehrere geografisch getrennte Availability Zones innerhalb einer AWS-Region. Das Speichern von Daten über mehrere Availability Zones hinweg bietet kontinuierliche Verfügbarkeit der Daten, auch wenn eine oder mehrere Availability Zones in einer nicht verfügbaren AWS-Region sind.
- **One Zone** – One Zone-Dateisysteme speichern Daten innerhalb einer einzigen Availability Zone in einer AWS-Region. Das Speichern von Daten in einer einzigen Availability Zone ermöglicht eine kontinuierliche Verfügbarkeit der Daten. In dem unwahrscheinlichen Fall, dass die Availability Zone ganz oder teilweise verloren geht oder beschädigt wird, können in diesen Dateisystemen gespeicherte Daten jedoch 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 . Dabei handelt es sich um den Allzweck-Leistungsmodus und die Elastic-Durchsatzmodi.

- Allzweck – Der Allzweck-Leistungsmodus ist ideal für latenzempfindliche Anwendungen wie Web-Serving-Umgebungen, Content-Management-Systeme, Basisverzeichnisse und allgemeine Dateibereitstellung.
- Elastic – Der Elastic-Durchsatzmodus ist so konzipiert, dass die Durchsatzleistung automatisch nach oben oder unten skaliert wird, um die Anforderungen Ihrer Workload-Aktivität zu erfüllen.

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

Amazon EFS bietet file-system-access Semantik, wie starke Datenkonsistenz und Dateisperre. 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 [Sicherheit 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 Amazon 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-Clientzugriff auf 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 Elastic File System](#) und [Steuerung des Netzwerkzugriffs auf Amazon EFS-Dateisysteme für NFS-Clients](#).

 Note

Die Verwendung von Amazon EFS mit Microsoft Windows-basierten Amazon EC2-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 [Amazon EFS – Funktionsweise](#).
3. Versuchen Sie die einführenden Übungen:
 - [Erste Schritte](#)
 - [Anleitungen](#)

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

- [Arbeiten mit Amazon-EFS-Ressourcen](#)
- [Verwalten von Amazon-EFS-Dateisystemen](#)
- [Amazon-EFS-API](#)

Amazon EFS – Funktionsweise

Im Folgenden finden Sie eine Beschreibung der Funktionsweise von Amazon EFS, Details zur Implementierung und Sicherheitsüberlegungen.

Themen

- [Übersicht](#)
- [So funktioniert Amazon EFS mit Amazon EC2](#)
- [So funktioniert AWS Direct Connect mit AWS und AWS Managed VPN](#)
- [Wie Amazon EFS mit AWS Backup funktioniert](#)
- [Übersicht über die Implementierung](#)
- [Authentifizierung und Zugriffskontrolle](#)
- [Datenkonsistenz in Amazon EFS](#)
- [EFS-Speicherklassen](#)
- [EFS-Replikation](#)

Übersicht

Amazon Elastic File System bietet ein einfaches, Serverless- set-and-forget Elastic-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 Amazon EFS-Dateisystem in Ihrer Virtual Private Cloud (VPC) über das Protokoll Network File System Version 4.0 und 4.1 (NFSv4) mounten. Wir empfehlen die Verwendung eines Linux NFSv4.1-Clients der aktuellen Generation, wie er in den neuesten Amazon Linux, Amazon Linux 2, Red Hat, Ubuntu und macOS Big Sur AMIs zu finden ist, in Verbindung mit der Amazon EFS-Mountinghilfe. Anweisungen finden Sie unter [Verwenden der amazon-efs-utils Tools](#).

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. Anweisungen finden Sie unter [Installieren des NFS-Clients](#).

Sie können von mehreren NFS-Clients gleichzeitig auf Ihr Amazon EFS-Dateisystem zugreifen, so dass Anwendungen, die über eine einzelne Verbindung hinaus skalieren, auf ein Dateisystem zugreifen können. Amazon EC2 und andere AWS-Compute Instances, die in mehreren Availability

Zones innerhalb derselben AWS-Region laufen, können auf das Dateisystem zugreifen, so dass viele Benutzer auf eine gemeinsame Datenquelle zugreifen und diese gemeinsam nutzen können.

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

Um auf Ihr Amazon EFS-Dateisystem in einer VPC zuzugreifen, erstellen Sie ein oder mehrere Mounting-Hilfe(n) in der VPC.

- Für regionale Dateisysteme können Sie ein Mountinghilfe in jeder Availability Zone in der AWS-Region.
- Für One Zone-Dateisysteme erstellen Sie nur ein einziges Mounting-Ziel, das sich in der gleichen Availability Zone wie das Dateisystem befindet.

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

Ein Mounting-Ziel bietet eine IP-Adresse für einen NFSv4-Endpunkt, an dem Sie ein Amazon 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 Amazon EFS-Dateisystem kann Mounting-Ziele in jeweils nur einer VPC haben.

Die Mounting-Ziele 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.

Nach dem Mounting des Dateisystems mithilfe des DNS-Namens verwenden Sie es wie jedes andere POSIX-kompatible Dateisystem. Für Informationen zu Berechtigungen auf NFS-Ebene und dazugehörige Überlegungen vgl. [Mit Benutzern, Gruppen und Berechtigungen auf Network File System-\(NFS-\)Level arbeiten](#).

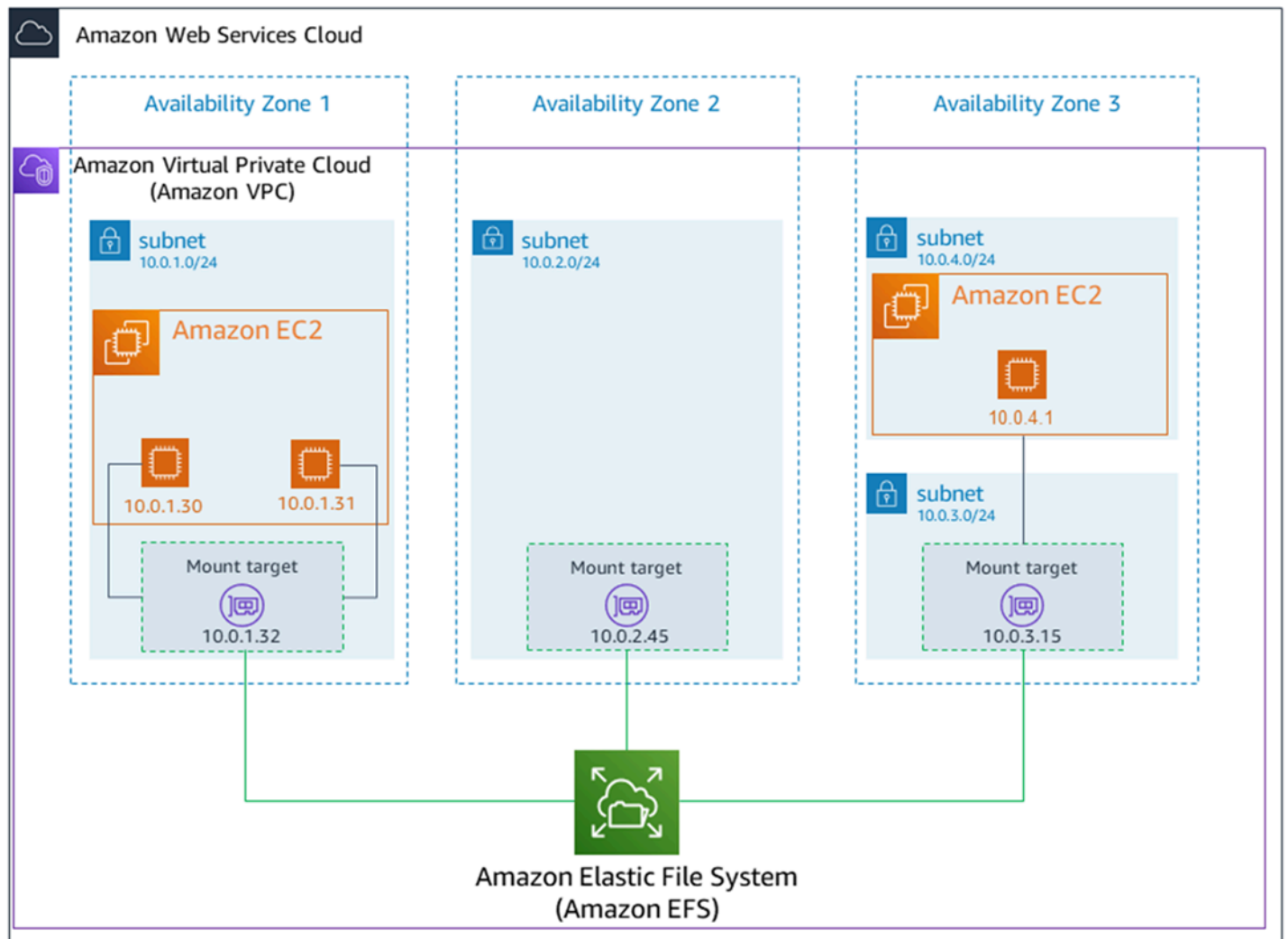
Sie können Ihre Amazon EFS-Dateisysteme auf Ihren Servern im On-Premises-Rechenzentrum mounten, wenn Sie mit AWS Direct Connect oder AWS VPN mit Ihrer Amazon VPC verbunden sind. Sie können Ihre EFS-Dateisysteme auf On-Premises-Servern mounten, um Datensätze zu EFS zu migrieren, Cloud Bursting-Szenarien zu ermöglichen oder Ihre On-Premises-Daten in Amazon EFS zu sichern.

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.

Amazon EFS-Dateisysteme

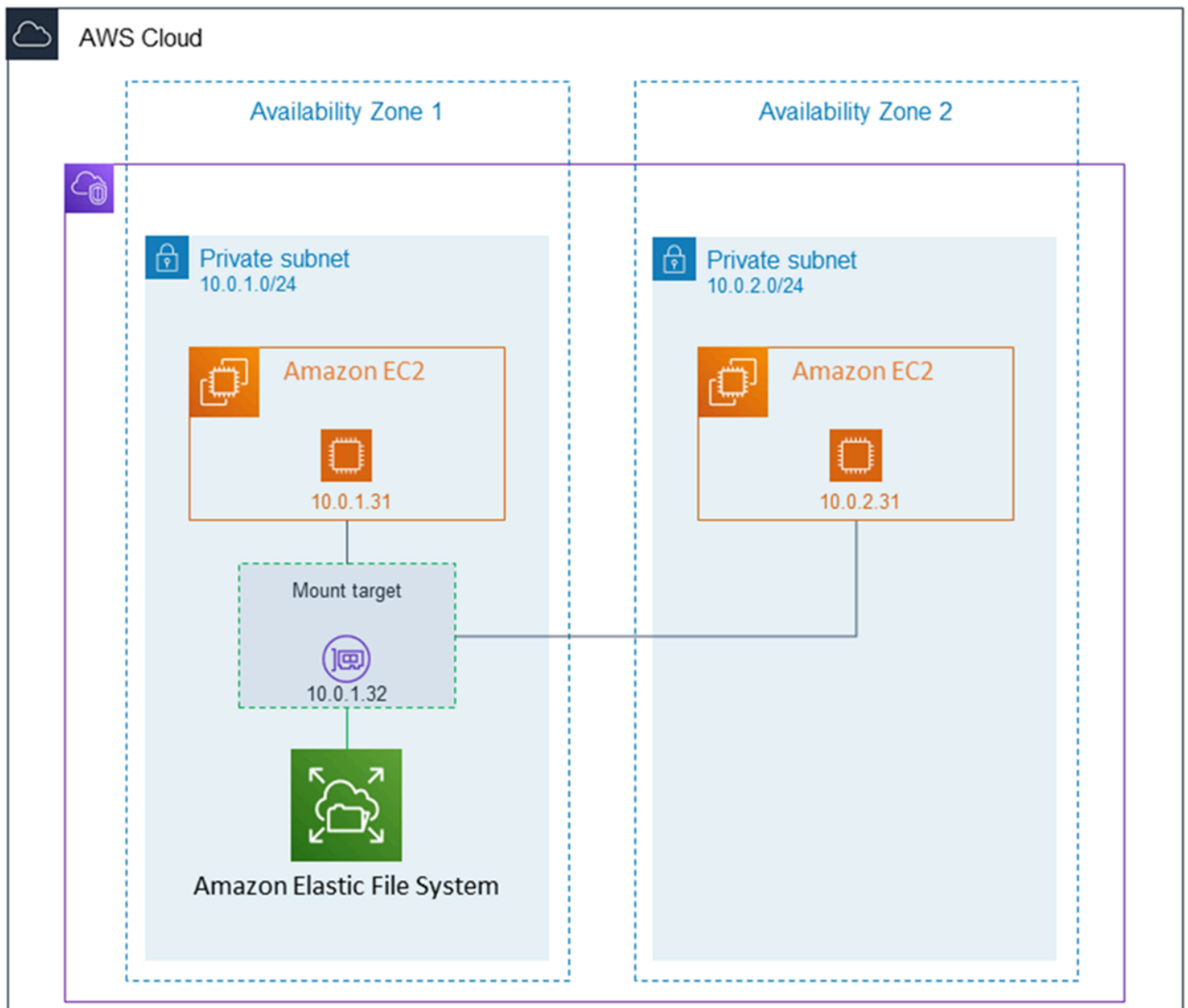
Die folgende Abbildung zeigt mehrere EC2-Instances, die auf ein Amazon EFS-Dateisystem zugreifen, das für mehrere Availability Zones in einer AWS-Region.



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 [Verwenden der EFS-Mountinghilfe zum Mouneten von EFS-Dateisystemen](#) [Mounting auf Amazon EC2-Linux-Instances mithilfe der EFS-Mountinghilfe](#).

Amazon EFS One Zone-Dateisysteme

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



In dieser Abbildung hat die VPC zwei Availability Zones mit jeweils einem Subnetz. Da der Dateisystemtyp Eine 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 AWS Direct Connect mit AWS und AWS Managed VPN

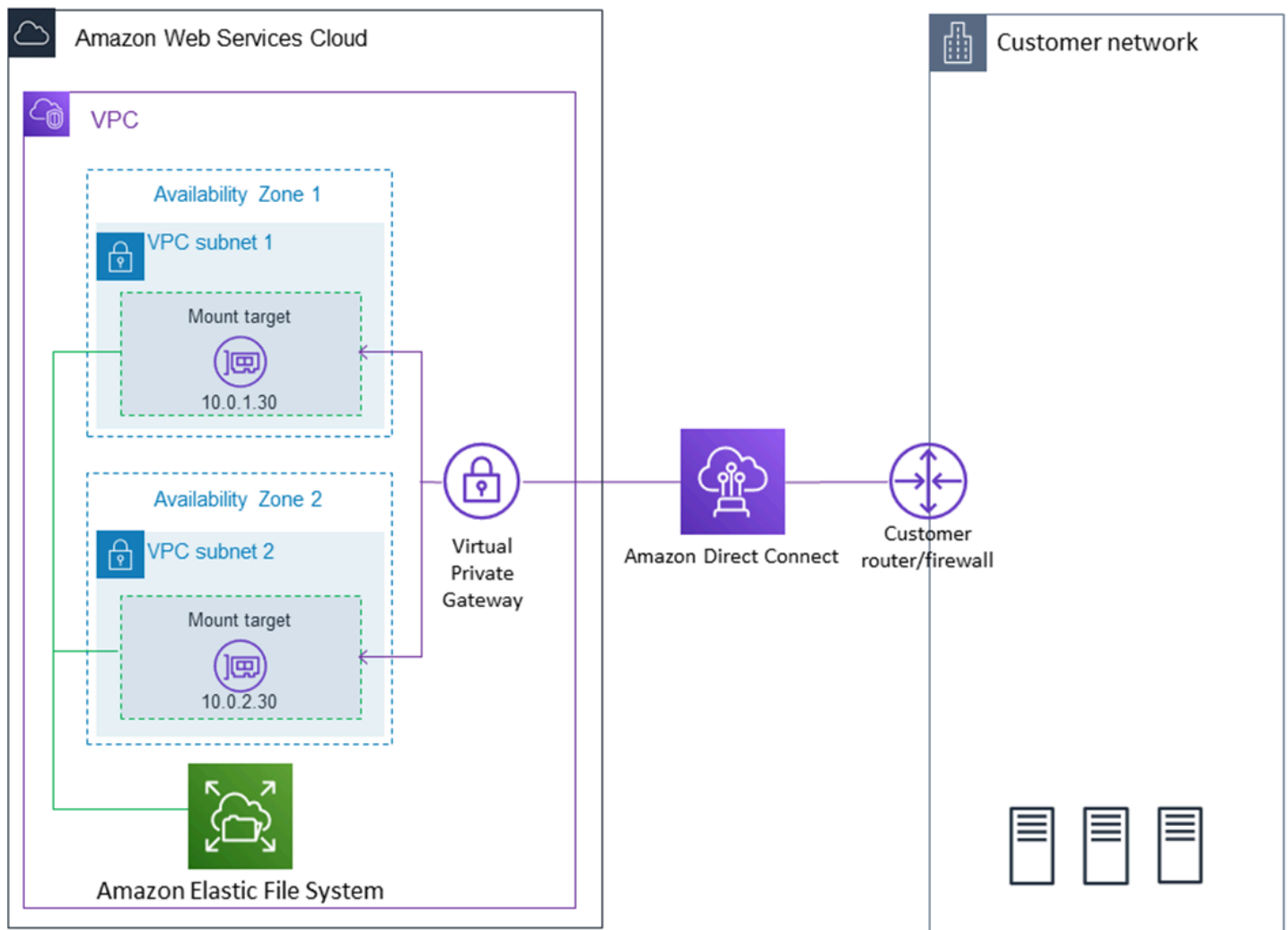
Wenn Sie ein Amazon EFS-Dateisystem verwenden, das auf einem On-Premises-Server gemountet ist, können Sie On-Premises-Daten in die in einem Amazon EFS-Dateisystem gehostete AWS Cloud migrieren. Sie können außerdem die Vorteile des Burstings nutzen. Mit anderen Worten: Sie können Daten von Ihren On-Premises-Servern 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 AWS Direct Connect-Verbindung zu Ihrer Amazon VPC wird Ihnen in Rechnung gestellt. Weitere Informationen finden Sie unter [AWS 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).



Verwenden Sie ein beliebiges Mounting-Ziel in Ihrer VPC, wenn Sie mithilfe einer AWS Direct Connect-Verbindung zwischen Ihrem On-Premises-Server und VPC das Subnetz des Mounting-Ziels 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 [Exemplarische Vorgehensweise: Erstellen und Bereitstellen eines lokalen Dateisystems mit VPN AWS Direct Connect](#).

Wie Amazon EFS mit AWS Backup funktioniert

Für eine umfassende Backup-Implementierung für Ihre Dateisysteme können Sie Amazon EFS mit AWS Backup verwenden. AWS Backup ist ein vollständig verwalteter Backup-Service, mit dem Sie die Datensicherung über AWS-Services in der Cloud und On-Premises ganz einfach zentralisieren und automatisieren können. Mit AWS Backup können Sie Backup-Richtlinien zentral

konfigurieren und die Backup-Aktivität für Ihre AWS-Ressourcen überwachen. Amazon EFS priorisiert Dateisystemoperationen immer vor Sicherungsvorgängen. Weitere Informationen zum Sichern von EFS-Dateisystemen mit AWS Backup finden Sie unter [Sichern Ihrer Amazon-EFS-Dateisysteme](#).

Übersicht über die Implementierung

In Amazon EFS ist ein Dateisystem die primäre Ressource. Jedes Dateisystem verfügt über Eigenschaften, wie z. B. ID, Erstellungstoken,stellungszeit, Dateisystemgröße in Byte, Anzahl der für das Dateisystem erstellten Mounting-Ziele sowie Lebenszyklusstatus des Dateisystems. Weitere Informationen finden Sie unter [CreateFileSystem](#).

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 [Erstellen und Verwalten von Mount-Zielen und Sicherheitsgruppen](#). Zuerst müssen Sie den NFS-Client auf Ihrer EC2-Instance installieren. Die [Erste Schritte](#) Übung enthält step-by-step Anweisungen.

- **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 Amazon EFS Access Points](#).

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 [Exemplarische Anleitung: Erstellen eines Amazon EFS-Dateisystems und das Mounten auf einer Amazon EC2 EC2-Instance mithilfe der AWS CLI](#).
- Sie können diese Ressourcen auch wie folgt programmgesteuert verwalten:
 - Verwenden Sie die AWS SDKs – Die AWS SDKs vereinfachen Ihre Programmieraufgaben, indem sie die zugrunde liegende Amazon EFS-API umhüllen. 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 aus Ihrer Anwendung auf – Wenn Sie die SDKs aus irgendeinem Grund nicht verwenden können, können Sie die Amazon EFS-API-Aufrufe direkt aus Ihrer Anwendung 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](#).

Authentifizierung und Zugriffskontrolle

Sie müssen über gültige Anmeldeinformationen verfügen, um Amazon EFS-API-Anforderungen, etwa zum Erstellen eines Dateisystems, durchzuführen. Darüber hinaus müssen Sie über die erforderlichen Berechtigungen zum Erstellen von Ressourcen bzw. zum Zugriff darauf verfügen.

Benutzer und Rollen, die Sie in AWS Identity and Access Management (IAM) erstellen, müssen die Berechtigung erhalten, Ressourcen zu erstellen oder darauf zuzugreifen. Weitere Informationen zu Berechtigungen finden Sie unter [Identitäts- und Zugriffsmanagement für Amazon Elastic File System](#).

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 Dateisystemdatenzugriffs](#).

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 bieten als close-to-open Semantik. Anwendungen, die synchronen Datenzugriff und nicht anhängende Schreibvorgänge durchführen, haben read-after-write Konsistenz für den Datenzugriff.

Sperrungen 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 beratende Sperren und Lese-/Schreiboperationen werden vor der Ausführung nicht auf kollidierende Sperren geprüft. Um beispielsweise Probleme mit der Dateisynchronisierung bei atomaren Operationen zu vermeiden, muss Ihre Anwendung die NFS-Semantik (z. B. close-to-open Konsistenz) kennen.
- Jede einzelne Datei kann in allen verbundenen Instances bis zu 512 Locks und auf die Datei zugreifende Benutzer verfügen.

EFS-Speicherklassen

Amazon EFS bietet verschiedene Speicherklassen für unterschiedliche Datenspeicheranforderungen. Standard ist die erste Speicherklasse, in die Daten geschrieben werden und ist die Speicherklasse für Daten, auf die häufig zugegriffen wird. Für Dateien, auf die weniger häufig zugegriffen wird, bietet Amazon EFS die Speicherklassen EFS Infrequent Access (IA) und EFS Archive. Die Speicherklasse IA ist kostenoptimiert für Daten, auf die ein paar Mal pro Quartal zugegriffen wird, und die Speicherklasse Archiv ist kostenoptimiert für Daten, auf die nur ein paar Mal pro Jahr oder weniger zugegriffen wird. Weitere Informationen über Amazon EFS-Speicherklassen finden Sie unter [EFS-Speicherklassen](#).

Lebenszyklusmanagement

Um Ihre Dateisysteme so zu verwalten, dass sie während ihres gesamten Lebenszyklus kostengünstig gespeichert werden, verwenden Sie Lebenszyklusmanagement.

Lebenszyklusmanagement überträgt Daten automatisch gemäß der Lebenszykluskonfiguration, die für das Dateisystem definiert ist, zwischen Speicherklassen. Die Lebenszykluskonfiguration ist eine Reihe von Lebenszyklusrichtlinien, die festlegen, wann die Dateisystemdaten in eine andere Speicherklasse überführt werden sollen.

Weitere Informationen finden Sie unter [Verwaltung des Dateisystemspeichers](#).

EFS-Replikation

Sie können mithilfe der Replikation ein Replikat Ihres Amazon-EFS-Dateisystems in der AWS-Region Ihrer Wahl erstellen. Die Replikation repliziert die Daten und Metadaten auf Ihrem EFS-Dateisystem automatisch und transparent in ein neues Ziel-EFS-Dateisystem, das in einer von AWS-Region Ihnen ausgewählten erstellt wird.

Bei der Replikation synchronisiert EFS die Quell- und Zieldateisysteme 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 [Replizieren von Dateisystemen](#).

Einrichten

Bevor Sie Amazon EFS zum ersten Mal verwenden, führen Sie die folgenden Aufgaben aus:

1. [So melden Sie sich für ein AWS-Konto an](#)
2. [Einen Administratorbenutzer erstellen](#)

So melden Sie sich für ein AWS-Konto an

Wenn Sie kein AWS-Konto haben, führen Sie die folgenden Schritte zum Erstellen durch.

Anmeldung für ein AWS-Konto

1. Öffnen Sie <https://portal.aws.amazon.com/billing/signup>.
2. Folgen Sie den Online-Anweisungen.

Bei der Anmeldung müssen Sie auch einen Telefonanruf entgegennehmen und einen Verifizierungscode über die Telefontasten eingeben.

Wenn Sie sich für ein AWS-Konto anmelden, wird ein Root-Benutzer des AWS-Kontos erstellt. Der Root-Benutzer hat Zugriff auf alle AWS-Services und Ressourcen des Kontos. Als bewährte Sicherheitsmethode weisen Sie einem [Administratorbenutzer Administratorzugriff](#) zu und verwenden Sie nur den Root-Benutzer, um [Aufgaben auszuführen, die Root-Benutzerzugriff](#) erfordern.

AWS sendet Ihnen eine Bestätigungs-E-Mail, sobald die Anmeldung abgeschlossen ist. Sie können jederzeit Ihre aktuelle Kontoaktivität anzeigen und Ihr Konto verwalten. Rufen Sie dazu <https://aws.amazon.com/> auf und klicken Sie auf Mein Konto.

Einen Administratorbenutzer erstellen

Wenn Sie sich für AWS-Konto registriert haben, sichern Sie Root-Benutzer des AWS-Kontos, aktivieren Sie AWS IAM Identity Center erstellen Sie einen Administratorbenutzer, damit Sie nicht den Root-Benutzer für alltägliche Aufgaben verwenden.

Schützen Ihres Root-Benutzer des AWS-Kontos

1. Melden Sie sich bei [AWS Management Console](#) als Kontobesitzer an, indem Sie Stammbenutzer auswählen und Ihre AWS-Konto-E-Mail-Adresse eingeben. Geben Sie auf der nächsten Seite Ihr Passwort ein.

Hilfe bei der Anmeldung mit dem Root-Benutzer finden Sie unter [Anmelden als Root-Benutzer](#) im AWS-AnmeldungBenutzerhandbuch zu .

2. Aktivieren Sie die Multi-Faktor-Authentifizierung (MFA) für den Root-Benutzer.

Anweisungen dazu finden Sie unter [Aktivieren eines virtuellen MFA-Geräts für den Root-Benutzer Ihres AWS-Konto \(Konsole\)](#) im IAM-Benutzerhandbuch.

Erstellen eines Administratorbenutzers

1. IAM Identity Center aktivieren.

Eine genaue Anleitung finden Sie unter [Aktivierung von AWS IAM Identity Center](#) im AWS IAM Identity Center-Benutzerhandbuch.

2. Gewähren Sie im IAM Identity Center einem Administratorbenutzer Administratorzugriff.

Ein Tutorial zur Verwendung von IAM-Identity-Center-Verzeichnis als Identitätsquelle finden Sie unter [Benutzerzugriff mit der Standardeinstellung konfigurieren IAM-Identity-Center-Verzeichnis](#) im AWS IAM Identity CenterBenutzerhandbuch.

Als Administratorbenutzer anmelden

- Um sich mit Ihrem IAM-Identity-Center-Benutzer anzumelden, verwenden Sie die Anmelde-URL, die an Ihre E-Mail-Adresse gesendet wurde, als Sie den IAM-Identity-Center-Benutzer erstellt haben.

Hilfe bei der Anmeldung mit einem IAM-Identity-Center-Benutzer finden Sie unter [Anmelden beim AWS-Zugangsportal](#) im AWS-Anmeldung Benutzerhandbuch zu.

Erste Schritte mit Amazon Elastic File System

In dieser „Erste Schritte“-Übung erfahren Sie, wie Sie schnell ein Amazon Elastic File System (Amazon EFS)-Dateisystem erstellen. Im Rahmen dieses Prozesses mounten Sie Ihr Dateisystem auf einer Amazon Elastic Compute Cloud (Amazon EC2)-Instance in Ihrer Virtual Private Cloud (VPC). Sie testen auch die end-to-end Einrichtung.

Für die Erstellung und Verwendung Ihres ersten Amazon-EFS-Dateisystems müssen vier Schritte ausgeführt werden:

- Erstellen Sie ein Amazon-EFS-Dateisystem.
- Erstellen Sie Ihre Amazon-EC2-Ressourcen, starten Sie Ihre Instance und mounten Sie das Dateisystem.
- Übertragen Sie Dateien in Ihr EFS-Dateisystem mit AWS DataSync.
- Bereinigen Sie Ihre Ressourcen und schützen Sie Ihr AWS-Konto.

Themen

- [Annahmen](#)
- [Verwandte Themen](#)
- [Schritt 1: Erstellen eines Amazon-EFS-Dateisystems](#)
- [Schritt 2: Erstellen Sie Ihre EC2-Ressourcen und starten Sie Ihre EC2-Instance](#)
- [Schritt 3: Übertragen von Dateien zu Amazon EFS mithilfe von AWS DataSync](#)
- [Schritt 4: Bereinigen Sie Ihre Ressourcen und schützen Sie Ihr AWS-Konto](#)

Annahmen

Für diese Übung gehen wir von den folgenden Annahmen aus:

- Sie sind bereits mit der Verwendung der Amazon-EC2-Konsole zum Starten von Instances vertraut.
- Ihre Amazon-VPC-, Amazon-EC2- und Amazon-EFS-Ressourcen befinden sich alle in derselben AWS-Region. In dieser Anleitung wird die Region USA West (Oregon) (us-west-2) verwendet.
- Sie haben eine Standard-VPC in der AWS-Region, die Sie für diese „Erste Schritte“-Übung verwenden. Wenn Sie keine Standard-VPC haben oder das Mounting Ihres Dateisystems von einer

neuen VPC mit neuen oder vorhandenen Sicherheitsgruppen aus durchführen möchten, können Sie diese „Erste Schritte“-Übung dennoch verwenden. Konfigurieren Sie hierfür [Verwendung von VPC-Sicherheitsgruppen für Amazon EC2-Instance und Mounting-Ziele](#).

- Sie haben die Standardregel für eingehenden Datenverkehr für die Standardsicherheitsgruppe nicht geändert.
- Sie haben in Ihrem AWS-Konto einen Administratorbenutzer erstellt und verwenden die Anmeldeinformationen für diesen Benutzer, um die Ressourcen in Ihrem Konto zu verwalten. Weitere Informationen finden Sie unter [Einrichten](#).

Verwandte Themen

Diese Anleitung zeigt außerdem eine exemplarische Vorgehensweise zum Durchführen einer ähnlichen „Erste Schritte“-Übung mit AWS Command Line Interface (AWS CLI)-Befehlen zur Durchführung der Amazon-EFS-API-Aufrufe. Weitere Informationen finden Sie unter [Exemplarische Anleitung: Erstellen eines Amazon EFS-Dateisystems und das Mounten auf einer Amazon EC2 EC2-Instance mithilfe der AWS CLI](#).

Schritt 1: Erstellen eines Amazon-EFS-Dateisystems

In diesem Schritt verwenden Sie die Amazon-EFS-Konsole, um ein Amazon-EFS-Dateisystem mit den vom Service empfohlenen Einstellungen zu erstellen.

Wenn Sie ein Dateisystem mit einer benutzerdefinierten Konfiguration erstellen möchten, finden Sie weitere Informationen unter [Erstellen eines Dateisystems mit benutzerdefinierten Einstellungen mithilfe der Amazon-EFS-Konsole](#).

So erstellen Sie ein Amazon-EFS-Dateisystem

1. Melden Sie sich bei der AWS Management Console an 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:

- Automatische Sicherungen sind aktiviert. Weitere Informationen finden Sie unter [Sichern Ihrer Amazon-EFS-Dateisysteme](#).
- Mountingziele, die mit den folgenden Einstellungen konfiguriert sind:
 - Werden in jeder Availability Zone in der AWS-Region erstellt, in der das Dateisystem erstellt wird.
 - Befinden sich in den Standardsubnetzen der von Ihnen ausgewählten VPC.
 - Verwenden der Standardsicherheitsgruppe der VPC: Sie können Sicherheitsgruppen verwalten, nachdem das Dateisystem erstellt wurde.

Weitere Informationen finden Sie unter [Verwalten der Netzwerkzugänglichkeit des Dateisystems](#).

- Regionaler Dateisystemtyp: Weitere Informationen finden Sie unter [EFS-Dateisystemtypen](#).
- Allgemeine Zwecke: Weitere Informationen finden Sie unter [Leistungsmodi](#).
- Elastic Throughput: Weitere Informationen finden Sie unter [Durchsatzmodi](#).
- Verschlüsselung von Daten im Ruhezustand mit Ihrem Standardschlüssel für Amazon EFS (aws/elasticfilesystem) – Weitere Informationen finden Sie unter [Verschlüsseln von Daten im Ruhezustand](#).
- Lebenszyklusverwaltung – Amazon EFS erstellt das Dateisystem mit den folgenden Lebenszyklusrichtlinien:
 - Übergang in IA ist auf 30 Tage seit dem letzten Zugriff festgelegt.
 - TransitionToArchive auf 90 Tage seit dem letzten Zugriff festgelegt.
 - Übergang zum Standard ist auf Keine gesetzt.

Weitere Informationen finden Sie unter [Verwaltung des Dateisystemspeichers](#).

Nachdem Sie das Dateisystem erstellt haben, können Sie die Einstellungen des Dateisystems mit Ausnahme der Verfügbarkeit und Zuverlässigkeit, der Verschlüsselung und des Leistungsmodus anpassen.

Auf der Seite Dateisysteme wird oben ein Banner angezeigt, 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 [Status des Dateisystems](#).

Schritt 2: Erstellen Sie Ihre EC2-Ressourcen und starten Sie Ihre EC2-Instance

Note

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

In diesem Schritt erstellen Sie eine neue Amazon-EC2-Instance, auf der Amazon Linux 2 ausgeführt wird, und konfigurieren sie so, dass das EFS-Dateisystem, das Sie gerade in [Schritt 1](#) erstellt haben, automatisch gemountet wird.

Bevor Sie eine Amazon-EC2-Instance starten und sich mit ihr verbinden können, müssen Sie ein Schlüsselpaar erstellen, es sei denn, es ist bereits eines vorhanden. Sie können ein Schlüsselpaar mithilfe der Amazon-EC2-Konsole erstellen und dann Ihre EC2-Instance starten.

So erstellen Sie ein Schlüsselpaar

- Folgen Sie den Schritten unter [Einrichten von Amazon EC2](#) im Amazon-EC2-Benutzerhandbuch für Linux-Instances, um ein Schlüsselpaar zu erstellen. Wenn Sie bereits über ein Schlüsselpaar verfügen, müssen Sie kein neues erstellen. Sie können Ihr vorhandenes Schlüsselpaar für diese Übung verwenden.

Starten der EC2-Instance und Mounten eines EFS-Dateisystems

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 Machine Image (AMI) ein Amazon Linux 2-AMI oben in der Liste und klicken Sie auf 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):
 - Lassen Sie Anzahl der Instances auf 1 stehen.
 - Belassen Sie Kaufoption auf der Standardeinstellung.

- Wählen Sie für Network (Netzwerk) den Eintrag für dieselbe VPC aus, die Sie sich beim Erstellen Ihres EFS-Dateisystems in [Schritt 1: Erstellen eines Amazon-EFS-Dateisystems](#) notiert haben.
 - Wählen Sie für Subnet (Subnetz) ein Standardsubnetz in einer beliebigen Availability Zone aus.
 - Stellen Sie unter File systems (Dateisysteme) sicher, dass das EFS-Dateisystem, das Sie in [Schritt 1: Erstellen eines Amazon-EFS-Dateisystems](#) erstellt haben, ausgewählt ist. Der Pfad neben der Dateisystem-ID ist der Mounting-Punkt, den die EC2-Instance verwendet, die Sie ändern können.
 - Die Benutzerdaten enthalten automatisch die Befehle zum Mounten Ihres Amazon-EFS-Dateisystems.
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.
 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).

Sobald die EC2-Instance erstellt und verfügbar ist, wird sie in Ihr EFS-Dateisystem gemountet. Zu diesem Zeitpunkt können Sie Dateien in Ihr EFS-Dateisystem übertragen.

Schritt 3: Übertragen von Dateien zu Amazon EFS mithilfe von AWS DataSync

Nachdem Sie nun ein funktionierendes EFS-Dateisystem erstellt haben, können Sie AWS DataSync zum Übertragen von Dateien aus einem vorhandenen Dateisystem an Amazon EFS verwenden. AWS DataSync ist ein Datenübertragungsservice, der das Verschieben und Replizieren von Daten zwischen On-Premises-Speichersystemen und AWS-Speicherservices über das Internet oder AWS

Direct Connect vereinfacht, automatisiert und beschleunigt. AWS DataSync kann Ihre Dateidaten und auch Dateisystem-Metadaten wie Eigentümerschaft, Zeitstempel und Zugriffsberechtigungen übertragen.

Bevor Sie beginnen

In diesem Schritt wird Folgendes vorausgesetzt:

- Ein NFS-Quelldateisystem, von dem Sie Dateien übertragen können. Dieses Quellsystem muss über NFS Version 3, Version 4 oder 4.1 zugreifbar sein. Beispiele für Dateisysteme sind Dateisysteme in einem On-Premises-Rechenzentrum, selbstverwaltete Dateisysteme in der Cloud und Amazon-EFS-Dateisysteme.
- Ein EFS-Dateisystem, in das Dateien übertragen werden können. Wenn Sie nicht über ein EFS-Dateisystem verfügen, erstellen Sie eines. Weitere Informationen finden Sie unter [Erste Schritte mit Amazon Elastic File System](#).
- Ihr Server und Netzwerk erfüllt die AWS DataSync-Anforderungen. Weitere Informationen finden Sie in den [AWS DataSync-Anforderungen](#).

Führen Sie folgende Schritte aus, um Dateien mit AWS DataSync von einem Quellspeicherort an einen Zielspeicherort zu übertragen:

- Laden Sie einen Agent herunter, stellen Sie ihn in Ihrer Umgebung bereit, und aktivieren Sie ihn.
- Erstellen und konfigurieren Sie einen Quell- und Zielspeicherort.
- Erstellen und konfigurieren Sie eine Aufgabe.
- Führen Sie die Aufgabe aus, um Dateien von der Quelle zum Ziel zu übertragen.

Weitere Informationen zum Übertragen von Dateien von einem vorhandenen On-Premises-Dateisystem an Ihr EFS-Dateisystem finden Sie unter [Erste Schritte mit AWS DataSync](#) im AWS DataSync-Benutzerhandbuch. Weitere Informationen zum Übertragen von Dateien aus einem bestehenden In-Cloud-Dateisystem an Ihr EFS-Dateisystem finden Sie unter [Bereitstellen des AWS DataSync-Agenten als Amazon-EC2-Instance](#) im AWS DataSync-Benutzerhandbuch und unter [Amazon EFS AWS DataSync In-Cloud Transfer Quick Start and Scheduler](#).

Schritt 4: Bereinigen Sie Ihre Ressourcen und schützen Sie Ihr AWS-Konto

Diese Anleitung enthält exemplarische Vorgehensweisen, mit denen Sie Amazon EFS weiter kennenlernen können. Bevor Sie diesen Schritt für die Bereinigung durchführen, können Sie die erstellten Ressourcen verwenden, zu denen Sie in dieser „Erste Schritte“-Anleitung eine Verbindung hergestellt haben. Weitere Informationen finden Sie unter [Anleitungen](#). Nachdem Sie die exemplarischen Vorgehensweisen abgeschlossen haben, oder wenn Sie sie nicht nutzen möchten, befolgen Sie diese Schritte, um Ihre Ressourcen zu bereinigen und Ihr AWS-Konto zu schützen.

So bereinigen Sie Ihre Ressourcen und schützen Ihr Konto

1. Stellen Sie eine Verbindung zu Ihrer Amazon-EC2-Instance her.
2. Heben Sie das Mounting des EFS-Dateisystems mit dem folgenden Befehl auf.

```
$ sudo umount efs
```

3. Öffnen Sie die EFS-Konsole unter <https://console.aws.amazon.com/efs/>.
4. Wählen Sie das EFS-Dateisystem, das Sie löschen möchten, aus der Liste der Dateisysteme aus.
5. Klicken Sie bei Aktionen auf Dateisystem löschen.
6. 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.
7. Öffnen Sie die Amazon-EC2-Konsole unter <https://console.aws.amazon.com/ec2/>.
8. Wählen Sie die Amazon-EC2-Instance aus der Liste der Instances aus, die Sie beenden möchten.
9. Klicken Sie unter Actions (Aktionen) auf Instance State (Instance-Status) und dann auf Terminate (Beenden).
10. Klicken Sie unter Terminate Instances (Instances beenden) auf Yes, Terminate (Ja, beenden), um die Instance zu beenden, die Sie für diese „Erste Schritte“-Übung erstellt haben.
11. Wählen Sie im Navigationsbereich Security Groups (Sicherheitsgruppen) aus.
12. Wählen Sie den Namen der Sicherheitsgruppe, die Sie für diese „Erste Schritte“-Übung in [Schritt 2: Erstellen Sie Ihre EC2-Ressourcen und starten Sie Ihre EC2-Instance](#) im Rahmen des Amazon-EC2-Assistenten zum Starten von Instances erstellt haben.

 **Warning**

Löschen Sie nicht die Standardsicherheitsgruppe für Ihre VPC.

13. Wählen Sie für Actions (Aktionen) die Option Delete Security Group (Sicherheitsgruppe löschen) aus.
14. Klicken Sie unter Delete Security Group (Sicherheitsgruppe löschen) auf Yes, Delete (Ja, löschen), um die Sicherheitsgruppe zu löschen, die Sie für diese „Erste Schritte“-Übung erstellt haben.

Amazon-EFS-Dateisystemtypen und -Speicherklassen

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 über mehrere geografisch getrennte Availability Zones innerhalb einer AWS-Region. Das Speichern von Daten über mehrere Availability Zones hinweg ermöglicht eine kontinuierliche Verfügbarkeit der Daten, auch wenn eine oder mehrere Availability Zones in einer nicht verfügbar AWS-Region sind.
- **One Zone** – One Zone-Dateisysteme speichern Daten innerhalb einer einzigen Availability Zone in einem AWS-Region. Das Speichern von Daten in einer einzigen Availability Zone ermöglicht eine kontinuierliche Verfügbarkeit der Daten. In dem unwahrscheinlichen Fall, dass die Availability Zone ganz oder teilweise verloren geht oder beschädigt wird, können in diesen Dateisystemen gespeicherte Daten jedoch verloren gehen.

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

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

Note

One-Zone-Dateisysteme sind nur für bestimmte Availability Zones verfügbar. Eine Tabelle mit den 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.

Dateisyst emtyp	Konzipiert für	Zuverlässigkeit (Auslegung)	Verfügbar keit	Availability Zones	Weitere Überlegun gen
Regional	Daten, die ein Höchstmaß an Zuverlässigkeit und Verfügbarkeit erfordern.	99,999999 999 % (11x9)	99,99 %	>=3	None
One Zone	Daten, für die keine höchste Zuverlässigkeit und Verfügbarkeit erforderlich ist.	99,999999 999 % (11x9)	99,99 %	1	Nicht widerstan dsfähig gegenüber dem Verlust der Availability Zone

Unterstützte Availability Zones für One Zone-Dateisysteme

One-Zone-Dateisysteme 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 für Ihre AWS Ressourcen](#) im AWS Resource Access Manager-Benutzerhandbuch.

Availability Zones, die One Zone-Dateisysteme unterstützen

AWS-Region Name	AWS-Region Code	Unterstützte 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
Asien-Pazifik (Hongkong)	ap-east-1	ape1-az1, ape1-az2, ape1-az3
Asien-Pazifik (Mumbai)	ap-south-1	aps1-az1, aps1-az2, aps1-az3
Asien-Pazifik (Osaka)	ap-northeast-3	apne3-az1, apne3-az2, apne3-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

AWS-Region Name	AWS-Region Code	Unterstützte AZ-IDs
Europa (Irland)	eu-west-1	euw1-az1, euw1-az2, euw1-az3
Europa (London)	eu-west-2	euw2-az1, euw2-az2
Europa (Mailand)	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
Naher Osten (Bahrain)	me-south-1	mes1-az1, mes1-az2, mes1-az3
South America (São Paulo)	sa-east-1	sae1-az1, sae1-az2, sae1-az3
AWS GovCloud (USA-Ost)	us-gov-east-1	usge1-az1, usge1-az2, usge1-az3
AWS GovCloud (USA-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-Standard-Speicherklasse geschrieben und können dann mithilfe des Lebenszyklusmanagements auf die EFS-Speicherklassen Infrequent Access und EFS Archive gestaffelt 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 Elastic-Durchsatz unterstützt. Sie können den Durchsatz Ihres Dateisystems nicht auf „Bursting“ oder „Bereitgestellt“ aktualisieren, sobald das Dateisystem Daten in der Archive-Speicherklasse enthält.

Optimierung der Speicherkosten

Die Speicherklassen IA und Archive sind für Dateien kostenoptimiert, die nicht die Latenzleistung des Standardspeichers erfordern. Die First-Byte-Latenz beim Lesen aus der oder Schreiben in die IA-Speicherklasse ist höher als die für die Standard-Speicherklasse.

Mithilfe des Lebenszyklusmanagements können Sie die Speicherkosten optimieren, indem Sie Daten basierend auf den Zugriffsmustern Ihres Workloads automatisch zwischen Speicherklassen verteilen. Sie können Dateien aus den Speicherklassen IA oder Archive in die Speicherklasse Standard verschieben, indem Sie die Lebenszyklusrichtlinie Übergang in den Standard in Ihrem Dateisystem festlegen. Mit dieser Einstellung werden Dateien beim Zugriff von IA oder Archive wieder in den Standard überführt. Wenn Sie möchten, dass Ihre Dateien in der häufig aufgerufenen Standardspeicherklasse verbleiben, deaktivieren Sie das Lebenszyklusmanagement auf dem Dateisystem. Weitere Informationen finden Sie unter [Verwaltung des Dateisystemspeichers](#).

Vergleich der Speicherklassen

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

Speicherklasse	Konzipiert für	Latenz beim Lesen des ersten Bytes	Haltbarkeit (ausgelegt) ¹	Verfügbarkeits-SLA	Availability Zones	Mindestabrechnungsbetrag pro Datei ²	Mindesteiche
EFS Standard	Aktive Daten, die eine schnelle Latenzleistung unter einer Millisekunde erfordern	Unter einer Millisekunde	99,999999999 % (11 9)	99,99 % (regional) 99,9 % (One Zone)	=>3 (regional) 1 (eine Zone)	Nicht zutreffend	Nicht zutreffend

Speicherklasse	Konzipiert für	Latenz beim Lesen des ersten Bytes	Haltbarkeit (ausgelegt) ¹	Verfügbarkeits-SLA	Availability Zones	Mindestabrechnungsbetrag pro Datei ²	Minde...
EFS Infrequent Access	Inaktive Daten, auf die nur wenige Male pro Quartal zugegriffen wird.	Zehn Millisekunden				128 KiB	Nicht zutreffend
EFS Archive	Inaktive Daten, auf die einige Male pro Jahr zugegriffen wird	Zehn Millisekunden		99,9 % (regional)	=>3 (regional)	128 KiB	90 Tage

Note

¹Da One-Zone-Dateisysteme Daten in einer einzigen AWS Availability Zone speichern, können Daten, die in diesen Dateisystemtypen gespeichert sind, im Notfall oder bei einem anderen Fehler verloren gehen, der alle Kopien der Daten innerhalb der Availability Zone betrifft, oder im Falle einer Löschung der Availability Zone.

²Lebenszyklusrichtlinien, die am oder nach dem 24:00 Uhr PT aktualisiert wurden, werden am 26. November 2023 Dateien von < 128 KiB in die IA-Klasse gestuft. Weitere Informationen darüber, wie Amazon EFS einzelne Dateien und Metadaten misst und abrechnet, finden Sie unter [Messung: Wie Amazon EFS die Größe von Dateisystemen und Objekten meldet](#).

Preisgestaltung der Speicherklasse

Ihre Abrechnung erfolgt gemäß der in jeder Speicherklasse gespeicherten Datenmenge. Außerdem werden Ihnen Datenzugriffsgebühren in Rechnung gestellt, wenn Dateien im IA- oder Archive-Speicher gelesen werden, oder für Daten, die mithilfe des Lebenszyklusmanagements zwischen Speicherklassen übergehen. 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 gilt für die Speicherklassen Infrequent Access (IA) und Archive eine Mindestabrechnungsgebühr pro Datei von 128 KiB. Unterstützung für Dateien kleiner als 128 KiB ist nur für Lebenszyklusrichtlinien verfügbar, die am oder nach dem 24:00 Uhr ME, 26. November 2023, aktualisiert wurden. Weitere Informationen darüber, wie Amazon EFS einzelne Dateien und Metadaten misst und abrechnet, finden Sie unter [Messung: Wie Amazon EFS die Größe von Dateisystemen und Objekten meldet](#).

Für Dateisysteme, die den Bereitgestellt- oder Bursting-Durchsatz verwenden, fallen zusätzliche Preise an.

- 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 EFS-Durchsatzmodi finden Sie unter [Durchsatzmodi](#).

Note

Es fallen keine Datenzugriffsgebühren an, wenn Sie verwenden, AWS Backup um EFS-Dateisysteme mit aktiviertem Lebenszyklusmanagement zu sichern. Weitere Informationen zu AWS Backup und Lebenszyklusmanagement finden Sie unter [EFS-Speicherklassen](#).

Anzeigen der Größe der Speicherklasse

Sie können mithilfe der Amazon-EFS-Konsole, der oder der EFS-API anzeigen AWS CLI, wie viele Daten in jeder Speicherklasse Ihres Dateisystems gespeichert sind.

Anzeigen der Speicherdatengröße in der Amazon-EFS-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 128KiB sind, werden auf 128KiB.
- Größe in Archive ist die Größe (in Binärbyte) der in der „EFS Archive“-Speicherklasse gespeicherten Daten. Dateien, die kleiner als 128KiB sind, werden auf 128KiB.

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](#).

Anzeigen der Speicherdatengröße mithilfe der AWS CLI

Sie können mithilfe der AWS CLI oder der 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 angezeigt wird `Value`.

```
{
  "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 Amazon-EFS-Dateisystemobjekten](#).

Arbeiten mit Amazon-EFS-Ressourcen

Amazon EFS bietet elastischen, gemeinsam genutzten und mit POSIX kompatiblen 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 in der aus zugegriffen werden AWS-Region , in der es erstellt wird.

Sie können ein Amazon-EFS-Dateisystem auf EC2-Instances in Ihrer Virtual Private Cloud (VPC) basierend auf der Amazon VPC mithilfe der Network File System-Versionen 4.0 und 4.1 (NFSv4) mounten. Weitere Informationen finden Sie unter [Amazon EFS – Funktionsweise](#).

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.

Wenn Sie noch nicht mit Amazon EFS vertraut sind, empfehlen wir Ihnen, die folgenden Übungen auszuprobieren, die Ihnen eine Erfahrung mit end-to-end der Verwendung eines Amazon-EFS-Dateisystems bieten:

- [Erste Schritte](#) – In der Übung „Erste Schritte“ werden Sie Schritt für Schritt durch die Erstellung eines Dateisystems mit den vom Service empfohlenen Einstellungen geführt. In dieser Übung

erstellen Sie ein Dateisystem mithilfe des Amazon-EFS-Quick-Create-Assistenten, mounten es auf einer EC2-Instance und testen die Einrichtung. Die Konsole kümmert sich um viele Dinge und hilft Ihnen, das end-to-end Erlebnis schnell einzurichten.

- [Exemplarische Anleitung: Erstellen eines Amazon EFS-Dateisystems und das Mounten auf einer Amazon EC2 EC2-Instance mithilfe der AWS CLI](#) – Die Anleitung ähnelt der Übung „Erste Schritte“, verwendet jedoch die AWS Command Line Interface (AWS CLI), um die meisten Aufgaben auszuführen. Da die AWS CLI Befehle eng mit der Amazon-EFS-API verknüpft sind, kann Ihnen die exemplarische Vorgehensweise dabei helfen, sich mit den Amazon-EFS-API-Operationen vertraut zu machen.

Weitere Informationen über das Erstellen von EFS-Ressourcen und den Zugriff auf ein Dateisystem finden Sie in den folgenden Themen.

Themen

- [Ressourcen-IDs](#)
- [Erstellen eines Amazon-EFS-Dateisystems](#)
- [Löschen eines Amazon-EFS-Dateisystems](#)
- [Erstellen und Verwalten von Mount-Zielen und Sicherheitsgruppen](#)
- [Erstellen von Sicherheitsgruppen](#)
- [Erstellen von Dateisystemrichtlinien](#)
- [Erstellen und Löschen von Zugangspunkten](#)
- [Markieren der Amazon-EFS-Ressourcen](#)

Ressourcen-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 den Ziffern 0–9 und Kleinbuchstaben a–f.

Vor Oktober 2021 wurden für die IDs, die neu erstellten Dateisystem- und Mount-Ziel-Ressourcen zugewiesen wurden, 8 Zeichen nach dem Bindestrich verwendet (z. B. fs-12345678). Von Mai 2021 bis Oktober 2021 änderten wir die IDs dieser Ressourcentypen so, dass nach dem Bindestrich 17 Zeichen verwendet werden (z. B. fs-1234567890abcdef0). Je nachdem, wann Ihr Konto erstellt wurde, verfügen Sie möglicherweise über Dateisystem- und Mount-Ziel-Ressourcen

mit kurzen IDs, obwohl alle neuen Ressourcen dieser Typen die längeren IDs erhalten: Die IDs vorhandener EFS-Ressourcen ändern sich nie.

Erstellen eines Amazon-EFS-Dateisystems

Im Folgenden erfahren Sie, wie Sie ein Amazon-EFS-Dateisystem mithilfe der AWS Management Console und der erstellen AWS CLI.

Wenn Sie Amazon EFS noch nicht kennen, empfehlen wir Ihnen, die Übung „Erste Schritte“ durchzugehen. Diese Übung enthält konsolenbasierte end-to-end Anweisungen zum Erstellen und Zugreifen auf ein Dateisystem in Ihrer Virtual Private Cloud (VPC). Weitere Informationen finden Sie unter [Erste Schritte](#).

Themen

- [Voraussetzungen](#)
- [Konfigurationsoptionen beim Erstellen eines Dateisystems](#)
- [Erstellen eines Dateisystems mit benutzerdefinierten Einstellungen mithilfe der Amazon-EFS-Konsole](#)
- [Erstellen eines Dateisystems mithilfe der AWS CLI](#)

Voraussetzungen

In diesem Abschnitt werden die Anforderungen und Voraussetzungen für die Erstellung von Amazon-EFS-Dateisystemen beschrieben.

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-Anforderungstoken ist eine eindeutige Zeichenfolge, die Sie beim Senden einer API-Anforderung angeben.

Ein Client-Anforderungstoken kann eine beliebige Zeichenfolge sein, die bis zu 64 ASCII-Zeichen enthält. Wenn Sie ein Client-Anforderungstoken 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 Service empfohlenen Einstellungen zu erstellen, hat das Erstellungstoken das folgende Format:

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

Erforderliche Berechtigungen

Um EFS-Ressourcen zu erstellen, z. B. ein Dateisystem und Zugriffspunkte, müssen Sie über AWS Identity and Access Management (IAM)-Berechtigungen für die entsprechende API-Operation und -Ressource verfügen.

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 serviceverknüpfte IAM-Rolle, die Berechtigungen enthält, die zum Aufrufen anderer AWS-Services in Ihrem Namen erforderlich sind. Weitere Informationen zum Verwalten von Berechtigungen für die API-Operationen finden Sie unter [Identitäts- und Zugriffsmanagement für Amazon Elastic File System](#).

Konfigurationsoptionen beim Erstellen eines Dateisystems

Sie können ein Dateisystem mithilfe der Amazon-EFS-Konsole oder mit der AWS Command Line Interface (AWS CLI) erstellen. Sie können Dateisysteme auch programmgesteuert erstellen, indem Sie - AWS SDKs oder die Amazon-EFS-API direkt verwenden. Wenn Sie die Amazon-EFS-API oder ein AWS -SDK verwenden, können Sie die `CreateFileSystem` EFS-API-Aktion verwenden, um Dateisystemrichtlinien zu erstellen.

Wenn Sie ein Amazon-EFS-Dateisystem mithilfe des benutzerdefinierten Erstellungsablaufs in der Konsole oder in der AWS CLI erstellen, können Sie Einstellungen für die folgenden Dateisystemfunktionen und Konfigurationsoptionen wählen.

Dateisystemtyp

Der Dateisystemtyp bestimmt die Verfügbarkeit und Haltbarkeit, mit der ein Amazon-EFS-Dateisystem Daten in einer AWS-Region speichert. Sie haben folgende Möglichkeiten für Ihren Dateisystemtyp:

- Wählen Sie **Regional** aus, um ein Dateisystem zu erstellen, das Daten und Metadaten redundant in allen Availability Zones innerhalb einer AWS-Region speichert. Sie können in jeder Availability Zone ein Mount-Ziel in einer AWS-Region erstellen. Regional bietet ein Höchstmaß an Verfügbarkeit und Haltbarkeit.
- Wählen Sie **One Zone** aus, um ein Dateisystem zu erstellen, das Daten und Metadaten redundant innerhalb einer Availability Zone speichert. Dateisysteme, die Speicherklassen verwenden, können nur ein einziges Mount-Ziel haben. Das Mount-Ziel muss sich in derselben Availability Zone befinden, in der das Dateisystem erstellt wird.

Automatische Sicherungen

Automatische Sicherungen sind standardmäßig immer aktiviert, wenn Sie mithilfe der Konsole ein Dateisystem erstellen. Wenn Sie die CLI oder API verwenden, um ein Dateisystem zu erstellen, sind automatische Sicherungen standardmäßig nur aktiviert, wenn Sie Dateisysteme erstellen, die One-Zone-Dateisysteme verwenden. Weitere Informationen finden Sie unter [Automatische Sicherungen](#).

Lebenszyklusrichtlinie

Bei der Lebenszyklusverwaltung werden Lebenszyklusrichtlinien verwendet, um Dateien basierend auf Zugriffsmustern automatisch in und aus der kostengünstigeren Speicherklasse Infrequent Access (IA) zu verschieben. Wenn Sie ein Dateisystem mithilfe der erstellen AWS Management Console, wird die Lebenszyklusrichtlinie des Dateisystems mit den folgenden Standardeinstellungen konfiguriert:

- Übergang in IA ist auf 30 Tage seit dem letzten Zugriff festgelegt.
- TransitionToArchive auf 90 Tage seit dem letzten Zugriff festgelegt.
- Übergang zum Standard ist auf Keine gesetzt.

Wenn Sie ein Dateisystem mithilfe der AWS CLI, der Amazon-EFS-API oder AWS SDKs erstellen, 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. Weitere Informationen finden Sie unter [Verwaltung des Dateisystemspeichers](#).

Verschlüsselung

Sie können beim Erstellen eines Dateisystems die Verschlüsselung im Ruhezustand aktivieren. Wenn Sie diese Aktivierung vornehmen, werden alle in Ihrem Dateisystem gespeicherten Daten und Metadaten verschlüsselt. Sie können die Verschlüsselung während der Übertragung später aktivieren, wenn Sie das Dateisystem mounten. Weitere Informationen zur Amazon-EFS-Verschlüsselung finden Sie unter [Datenverschlüsselung in Amazon EFS](#).

Zur Erstellung der Dateisystem-Mounting-Ziele in Ihrer VPC müssen Sie VPC-Subnetze angeben. Die Konsole füllt vorab die Liste der VPCs in Ihrem Konto, die sich in der ausgewählten AWS-Region befinden. Zuerst wählen Sie Ihre VPC, dann listet die Konsole die darin befindlichen Availability Zones auf. Für jede Availability Zone können Sie ein Subnetz aus der Liste auswählen oder das Standard-Subnetz verwenden, falls vorhanden. Nachdem Sie ein Subnetz ausgewählt haben, können Sie eine in dem Subnetz verfügbare IP-Adresse auswählen oder Amazon EFS automatisch eine Adresse auswählen lassen.

Durchsatzmodi

Es stehen drei Durchsatzmodi zur Auswahl:

- Elastisch (empfohlen) – Bietet einen Durchsatz, der automatisch in Echtzeit hoch- und herunterskaliert wird, um den Leistungsanforderungen Ihres Workloads gerecht zu werden.

Note

Der elastische Durchsatz ist nur für Dateisysteme mit dem Allzweck-Leistungsmodus verfügbar.

- Bereitgestellt – Stellt den von Ihnen angegebenen Durchsatz bereit, unabhängig von der Größe des Dateisystems.
- Bursting – Stellt einen Durchsatz bereit, der mit der Datenmenge im Standardspeicher skaliert.

Weitere Informationen finden Sie unter [Durchsatzmodi](#).

 Note

In Verbindung mit der Nutzung der Durchsätze „Elastisch“ und „Bereitgestellt“ fallen zusätzliche Gebühren an. Weitere Informationen finden Sie unter [Amazon EFS – Preise](#).

Leistungsmodi

Beim Erstellen eines Dateisystems können Sie auch einen Leistungsmodus wählen. Es stehen zwei Leistungsmodi zur Auswahl Allzweck und Max. I/O.

- Der Allzweckmodus hat die niedrigste Latenz pro Operation und wird für alle Dateisysteme empfohlen.
- Max. E/A ist ein Leistungstyp der vorherigen Generation, der für stark parallelisierte Workloads entwickelt wurde, die höhere Latenzen tolerieren können als der Allzweckmodus. Der Modus "Max. E/A" wird von One-Zone-Dateisystemen oder Dateisystemen, die den elastischen Durchsatzmodus 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](#).

Erstellen eines Dateisystems mit benutzerdefinierten Einstellungen mithilfe der Amazon-EFS-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 mit den vom Service empfohlenen Einstellungen finden Sie unter [Schritt 1: Erstellen eines Amazon-EFS-Dateisystems](#).

Das Erstellen eines Amazon-EFS-Dateisystems mit benutzerdefinierten Einstellungen mithilfe der Konsole ist ein Prozess mit 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 Management Console 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 aus, um ein benutzerdefiniertes Dateisystem zu erstellen, anstatt ein Dateisystem mithilfe der vom Service 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. Wählen Sie unter Dateisystemtyp eine Verfügbarkeitsoption aus:
 - Wählen Sie Regional aus, um ein Dateisystem zu erstellen, das Dateisystemdaten und Metadaten redundant in allen Availability Zones innerhalb einer AWS-Region speichert. Regional bietet ein Höchstmaß an Verfügbarkeit und Haltbarkeit.
 - Wählen Sie One Zone aus, um ein Dateisystem zu erstellen, das Dateisystemdaten und Metadaten redundant innerhalb einer 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. Weitere Informationen finden Sie unter [EFS-Speicherklassen](#).
 - c. Automatische Backups sind standardmäßig nicht aktiviert. Sie können automatische Backups ausschalten, indem Sie das Kontrollkästchen deaktivieren. Weitere Informationen finden Sie unter [Sichern Ihrer Amazon-EFS-Dateisysteme](#).
 - d. Ändern Sie für das Lebenszyklusmanagement 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 (Übergang in Archiv) – 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 Dateisystemspeichers](#).

- e. Für die Verschlüsselung ist die Verschlüsselung von Daten im Ruhezustand standardmäßig aktiviert. Amazon EFS verwendet standardmäßig Ihren AWS Key Management Service (AWS KMS) EFS-Serviceschlüssel (aws/elasticfilesystem). 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 Erstellen eines AWS KMS key aus, 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.

5. Für Leistungseinstellungen nehmen Sie folgendes vor:

- a. Für den Durchsatzmodus ist standardmäßig der Modus Elastisch 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 aus.

Amazon-EFS-Dateisysteme messen Leseanforderungen mit einem Drittel der Rate anderer Anforderungen. Nachdem Sie den Durchsatzmodus eingegeben haben, wird

eine Schätzung der monatlichen Kosten für das Dateisystem angezeigt. Sie können den Durchsatzmodus ändern, sobald 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 Netzwerkzugänglichkeit des Dateisystems](#).
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-Adresse – Standardmäßig wählt Amazon EFS die IP-Adresse automatisch aus den verfügbaren Adressen im Subnetz aus. Sie können auch eine bestimmte IP-Adresse eingeben, die sich im Subnetz befindet. Mount-Ziele haben zwar eine einzige IP-Adresse, sind aber redundante, hochverfügbare Netzwerkressourcen.
- Sicherheitsgruppen – Sie können eine oder mehrere Sicherheitsgruppen für das Mount-Ziel angeben. Weitere Informationen finden Sie unter [Verwendung von VPC-Sicherheitsgruppen für Amazon EC2-Instance und Mounting-Ziele](#).

Um eine weitere Sicherheitsgruppe hinzuzufügen oder die Sicherheitsgruppe zu ändern, wählen Sie Sicherheitsgruppen auswählen aus und fügen Sie eine weitere Sicherheitsgruppe aus der Liste hinzu. Wenn Sie die Standardsicherheitsgruppe nicht verwenden möchten, können Sie sie löschen. Weitere Informationen finden Sie unter [Erstellen von 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 Dateisystemdatenzugriffs](#).

1. In den Richtlinienoptionen können Sie eine beliebige Kombination der verfügbaren vorkonfigurierten Richtlinien auswählen:
 - Standardmäßig den Root-Zugriff verhindern
 - Standardmäßig nur Lesezugriff erzwingen
 - Verschlüsselung während der Übertragung für alle Clients erzwingen
2. Verwenden Sie den Richtlinieneditor, um eine vorkonfigurierte Richtlinie anzupassen oder Ihre eigene Richtlinie zu erstellen. Wenn Sie eine der vorkonfigurierten Richtlinien auswählen, wird die JSON-Richtliniendefinition im Richtlinieneditor angezeigt. Sie können das JSON bearbeiten, um eine Richtlinie Ihrer Wahl zu erstellen. Um Ihre Änderungen rückgängig zu machen, wählen Sie Löschen aus.

Die vorkonfigurierten Richtlinien werden in den Richtlinienoptionen wieder verfügbar.

3. 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 Erstellen aus, 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 eines Dateisystems mithilfe der 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. Informationen zur finden Sie AWS CLI unter [Installieren der AWS CLI](#) im AWS Command Line Interface -Benutzerhandbuch.

- Um ein verschlüsseltes Dateisystem zu erstellen, das die EFS-Archive-Speicherklassen verwendet und automatische Sicherungen aktiviert hat, verwenden Sie den `Amazon-create-file-systemEFS-CLI-Befehl` (die entsprechende Operation ist [CreateFileSystem](#)), wie im Folgenden dargestellt.

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

```
--profile adminuser
```

Beispielsweise erstellt der folgende `create-file-system`-Befehl ein Dateisystem in der `us-west-2` AWS-Region. Der Befehl gibt `MyFirstFS` als Erstellungstoken an. Eine Liste der AWS-Regionen, mit denen Sie ein Amazon-EFS-Dateisystem erstellen können, finden Sie unter [Allgemeine Amazon Web Services-Referenz](#).

```
aws efs create-file-system \  
--creation-token MyFirstFS \  
--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.

```
{  
  "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"  
    }  
  ]  
}
```

```
]
}
```

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

```
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 in Ihrem zurück, die in der angegebenen Region AWS-Konto erstellt wurden.

Löschen eines Amazon-EFS-Dateisystems

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 Mounting-Ziele und dann das Dateisystem.

Verwenden der -CLI

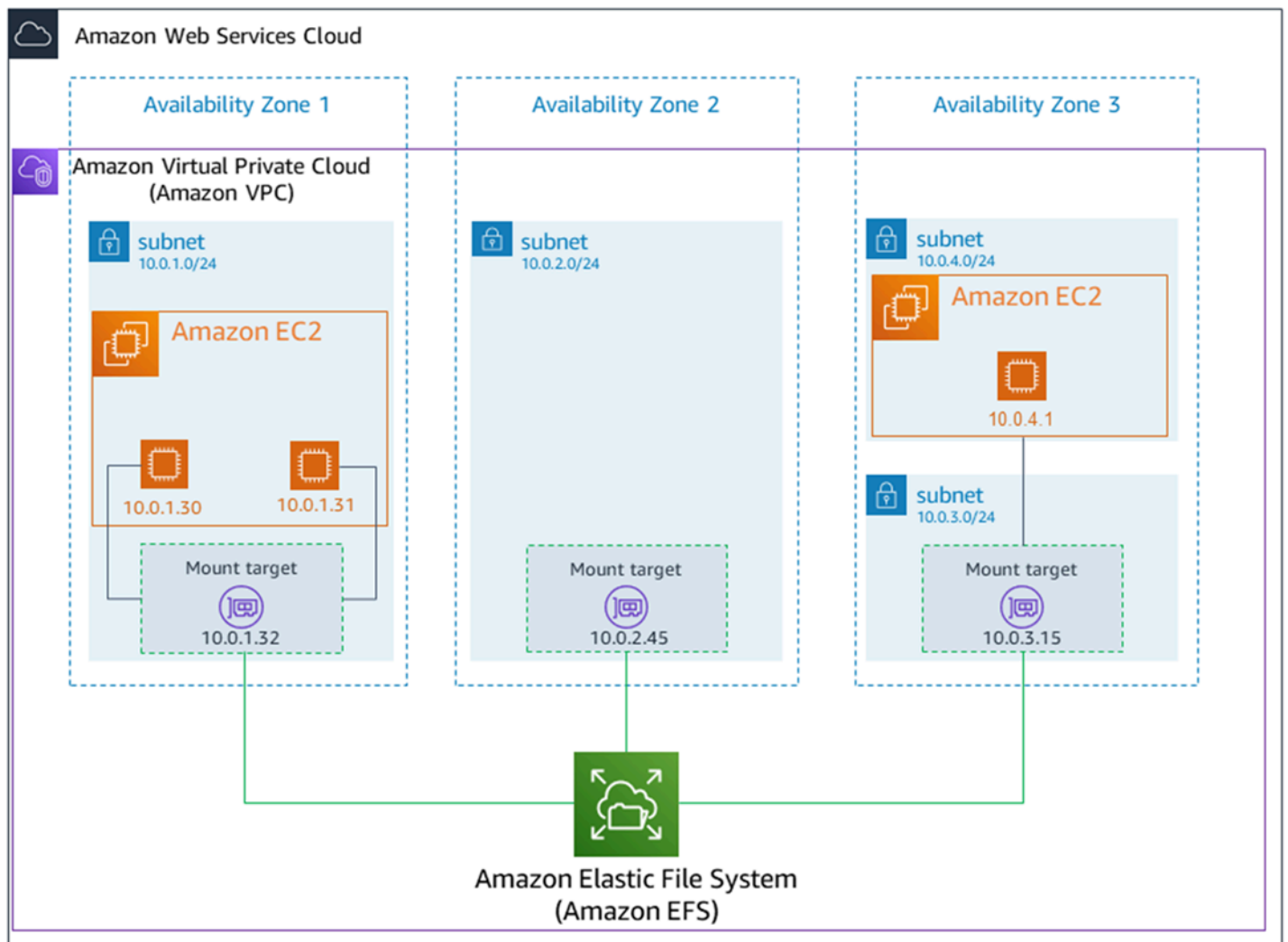
Bevor Sie den AWS CLI Befehl verwenden können, um ein Dateisystem zu löschen, müssen Sie alle Mount-Ziele und Zugriffspunkte löschen, die für das Dateisystem erstellt wurden.

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

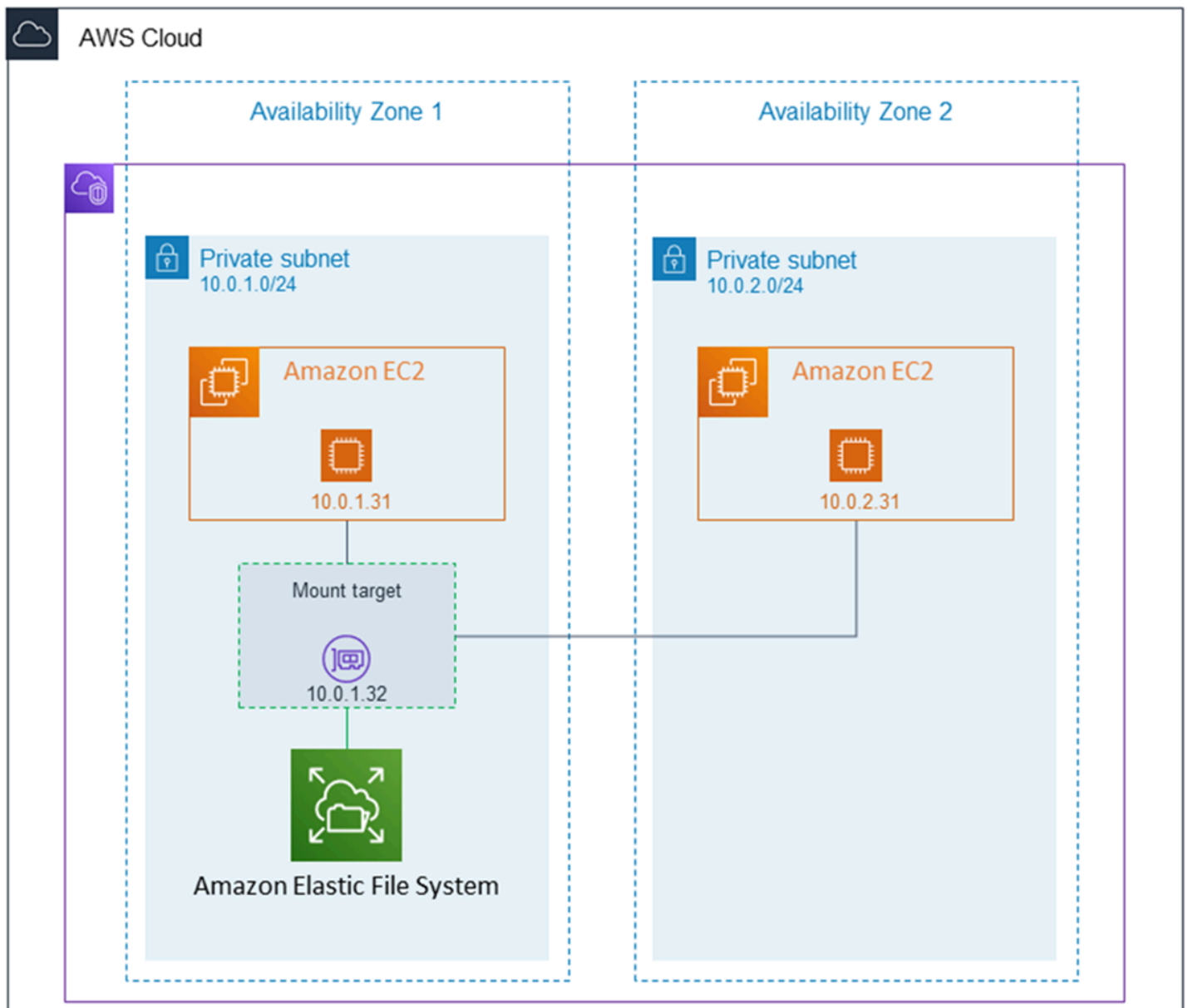
Erstellen und Verwalten von Mount-Zielen und Sicherheitsgruppen

Nachdem Sie ein Amazon-EFS-Dateisystem erstellt haben, können Sie Mount-Ziele erstellen. Für Amazon-EFS-Dateisysteme, die regionale Speicherklassen verwenden, können Sie ein Mount-Ziel in jeder Availability Zone in einer AWS-Region festlegen. 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 Datenverarbeitungs-Instances wie Amazon EC2, Amazon ECS und AWS Lambda in Ihrer Virtual Private Cloud (VPC) mounten.

Das folgende Diagramm zeigt ein Amazon-EFS-Dateisystem, das Standard-Speicherklassen verwendet, wobei Mount-Ziele in allen Availability Zones in der VPC erstellt wurden.



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 Mounting-Ziel 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 verfügt nur der Root-Benutzer auf der Amazon EC2-Instance über 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 [Mit Benutzern, Gruppen und Berechtigungen auf Network File System-\(NFS-\)Level arbeiten](#).

Sie können Mount-Ziele für ein Dateisystem mithilfe der AWS Management Console AWS CLI oder programmgesteuert mithilfe der - AWS SDKs erstellen. Wenn Sie die Konsole verwenden, können Sie Mount-Ziele beim ersten Erstellen eines Dateisystems oder nach der Erstellung des Dateisystems erstellen.

Anweisungen zum Erstellen von Mount-Zielen mithilfe der Amazon-EFS-Konsole beim Erstellen eines neuen Dateisystems finden Sie unter [Schritt 2: Konfigurieren des Netzwerkzugriffs](#).

Verwalten von Mount-Zielen mit der Amazon-EFS-Konsole

Gehen Sie wie folgt vor, um Mount-Ziele für ein vorhandenes Amazon-EFS-Dateisystem hinzuzufügen oder zu ändern.

Gehen Sie wie folgt vor, um Mount-Ziele auf einem Amazon-EFS-Dateisystem (Konsole) zu verwalten:

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die Amazon-EFS-Konsole unter <https://console.aws.amazon.com/efs/>.
2. Wählen Sie im linken Navigationsbereich die Option Dateisysteme aus. Auf der Seite Dateisysteme werden die EFS-Dateisysteme in Ihrem Konto angezeigt.
3. Wählen Sie das Dateisystem aus, für das Sie Mount-Ziele verwalten möchten, indem Sie dessen Namen oder die Dateisystem-ID auswählen, um die Seite mit den Dateisystemdetails anzuzeigen.
4. Wählen Sie Netzwerk aus, um die Liste der vorhandenen Mount-Ziele anzuzeigen.
5. Wählen Sie Verwalten aus, um die Seite Availability Zone aufzurufen und Änderungen vorzunehmen.

Auf dieser Seite können Sie für bestehende Mount-Ziele Sicherheitsgruppen hinzufügen und entfernen oder das Mount-Ziel löschen. Sie können auch neue Mount-Ziele erstellen.

 Note

Für One-Zone-Dateisysteme können Sie nur ein einziges Mount-Ziel erstellen, das sich in der gleichen Availability Zone wie das Dateisystem befindet.

- Um eine Sicherheitsgruppe aus einem Mount-Ziel zu entfernen, wählen Sie X neben der Sicherheitsgruppen-ID aus.
- Um einem Mount-Ziel eine Sicherheitsgruppe hinzuzufügen, wählen Sie Sicherheitsgruppen auswählen, um eine Liste der verfügbaren Sicherheitsgruppen anzuzeigen. Oder geben Sie eine Sicherheitsgruppen-ID in das Suchfeld oben in der Liste ein.
- Um ein Mount-Ziel zum Löschen in die Warteschlange zu stellen, wählen Sie Entfernen aus.

 Note

Bevor Sie ein Mount-Ziel löschen, müssen Sie das Mounting des Dateisystems aufheben.

- Um ein Mount-Ziel hinzuzufügen, wählen Sie Mount-Ziel hinzufügen aus. Diese Option ist nur für Dateisysteme verfügbar, die regionale EFS-Speicherklassen verwenden, und wenn Mount-Ziele nicht bereits in jeder Availability Zone für die AWS-Region vorhanden sind.

6. Wählen Sie Speichern aus, um Ihre Änderungen zu speichern.

Gehen Sie wie folgt vor, um die VPC für ein Amazon-EFS-Dateisystem (Konsole) zu ändern:

Um die VPC für die Netzwerkkonfiguration eines Dateisystems zu ändern, müssen Sie alle vorhandenen Mount-Ziele des Dateisystems löschen.

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. Auf der Seite Dateisysteme werden die EFS-Dateisysteme in Ihrem Konto angezeigt.
3. Wählen Sie für das Dateisystem, für das Sie die VPC ändern möchten, den Namen oder die Dateisystem-ID aus. Die Detailseite des Dateisystems wird angezeigt.
4. Wählen Sie Netzwerk aus, um die Liste der vorhandenen Mount-Ziele anzuzeigen.
5. Wählen Sie Manage (Verwalten). Die Seite Availability Zone wird angezeigt.

6. Entfernen Sie alle Mount-Ziele, die auf der Seite angezeigt werden.
7. Wählen Sie Speichern aus, um die Änderungen zu speichern und die Mount-Ziele zu löschen. Auf der Registerkarte Netzwerk wird der Status der Mount-Ziele als gelöscht angezeigt.
8. Wenn alle Status der Mount-Ziele als gelöscht angezeigt werden, wählen Sie Verwalten aus. Die Seite Availability Zone wird angezeigt.
9. Wählen Sie die neue VPC aus der Virtual Private Cloud (VPC)-Liste aus.
10. Um ein neues Mount-Ziel hinzuzufügen, wählen Sie Mount-Ziel hinzufügen aus. Geben Sie für jedes Mount-Ziel, das Sie hinzufügen, Folgendes ein:
 - Eine Availability Zone
 - Eine Subnetz-ID
 - Eine IP-Adresse oder lassen Sie sie auf Automatisch eingestellt
 - Eine oder mehrere Sicherheitsgruppen
11. Wählen Sie Speichern aus, um die VPC und Änderungen des Mount-Ziels zu speichern.

Verwaltung der Mount-Zielen mithilfe der AWS CLI

Note

Für One-Zone-Dateisysteme können Sie nur ein einziges Mount-Ziel erstellen, das sich in der gleichen Availability Zone wie das Dateisystem befindet.

Gehen Sie wie folgt vor, um ein Mount-Ziel (CLI) zu erstellen:

- Verwenden Sie zum Erstellen eines Mount-Ziels den CLI-Befehl `create-mount-target` (die entsprechende Operation ist [CreateMountTarget](#)), wie im Folgenden dargestellt:

```
$ 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 \  
--profile adminuser
```

Das folgende Beispiel zeigt den Befehl mit Beispieldaten.

```
$ aws efs create-mount-target \  
--file-system-id fs-0123467 \  
--subnet-id subnet-b3983dc4 \  
--security-group sg-01234567 \  
--region us-east-2 \  
--profile adminuser
```

Nach der erfolgreichen Erstellung des Mounting-Ziels gibt Amazon EFS die Beschreibung des Mounting-Ziels als JSON wie im folgenden Beispiel gezeigt aus.

```
{  
  "MountTargetId": "fsmt-f9a14450",  
  "NetworkInterfaceId": "eni-3851ec4e",  
  "FileSystemId": "fs-b6a0451f",  
  "LifeCycleState": "available",  
  "SubnetId": "subnet-b3983dc4",  
  "OwnerId": "23124example",  
  "IpAddress": "10.0.1.24"  
}
```

Gehen Sie wie folgt vor, um eine Liste von Mount-Zielen für ein Dateisystem (CLI) abzurufen:

- Sie können auch eine Liste der für ein Dateisystem erstellten Mount-Ziele mit dem [describe-mount-targets](#)-CLI-Befehl abrufen (die entsprechende Operation ist [DescribeMountTargets](#)), wie nachfolgend gezeigt.

```
$ aws efs describe-mount-targets --file-system-id fs-a576a6dc
```

```
{  
  "MountTargets": [  
    {  
      "OwnerId": "111122223333",  
      "MountTargetId": "fsmt-48518531",  
      "FileSystemId": "fs-a576a6dc",  
      "SubnetId": "subnet-88556633",  
      "LifeCycleState": "available",  
      "IpAddress": "172.31.25.203",  
      "NetworkInterfaceId": "eni-0123456789abcdef1",  
      "AvailabilityZoneId": "use2-az2",  
    }  
  ]  
}
```

```

        "AvailabilityZoneName": "us-east-2b"
    },
    {
        "OwnerId": "111122223333",
        "MountTargetId": "fsmt-5651852f",
        "FileSystemId": "fs-a576a6dc",
        "SubnetId": "subnet-44223377",
        "LifeCycleState": "available",
        "IpAddress": "172.31.46.181",
        "NetworkInterfaceId": "eni-0123456789abcdefa",
        "AvailabilityZoneId": "use2-az3",
        "AvailabilityZoneName": "us-east-2c"
    },
    {
        "OwnerId": "111122223333",
        "MountTargetId": "fsmt-5751852e",
        "FileSystemId": "fs-a576a6dc",
        "SubnetId": "subnet-a3520bcb",
        "LifeCycleState": "available",
        "IpAddress": "172.31.12.219",
        "NetworkInterfaceId": "eni-0123456789abcdef0",
        "AvailabilityZoneId": "use2-az1",
        "AvailabilityZoneName": "us-east-2a"
    }
]
}

```

Gehen Sie wie folgt vor, um ein vorhandenes Mount-Ziel (CLI) zu löschen:

- 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 \  

```

Gehen Sie wie folgt vor, um die Sicherheitsgruppe eines vorhandenen Mount-Ziels zu ändern:

- Um Sicherheitsgruppen zu ändern, die für ein Mountingziel gültig sind, verwenden Sie den `modify-mount-target-security-group` AWS CLI Befehl (die entsprechende Operation ist [ModifyMountTargetSecurityGroups](#)), um alle vorhandenen Sicherheitsgruppen zu ersetzen, wie im Folgenden gezeigt.

```
$ 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
```

Weitere Informationen finden Sie unter [Exemplarische Anleitung: Erstellen eines Amazon EFS-Dateisystems und das Mounten auf einer Amazon EC2 EC2-Instance mithilfe der AWS CLI](#).

Erstellen von Sicherheitsgruppen

Note

Der folgende Abschnitt gilt speziell für Amazon EC2 und beschreibt, wie Sie Sicherheitsgruppen erstellen. So können Sie Secure Shell (SSH) verwenden, um eine

Verbindung mit Instances herzustellen, die Amazon-EFS-Dateisysteme gemountet haben. Wenn Sie kein SSH zum Herstellen der Verbindung mit den Amazon-EC2-Instances verwenden, können Sie diesen Abschnitt überspringen.

Sowohl einer Amazon-EC2-Instance, als auch einem Mount-Ziel müssen Sicherheitsgruppen zugewiesen sein. Diese Sicherheitsgruppen fungieren als virtuelle Firewall zur Steuerung des Datenverkehrs zwischen ihnen. Wenn Sie beim Erstellen eines Mount-Ziels keine Sicherheitsgruppe bereitstellen, weist Amazon EFS die Standardsicherheitsgruppe der VPC zu.

Unabhängig davon müssen zur Ermöglichung des Datenverkehrs zwischen einer EC2-Instance und einem Mounting-Ziel (und damit dem Dateisystem) die folgenden Regeln in diesen Sicherheitsgruppen konfigurieren:

- Die Sicherheitsgruppen, die Sie einem Mounting-Ziel zuweisen, müssen den eingehenden Zugriff für das TCP-Protokoll auf dem NFS-Port von allen EC2-Instances aus erlauben, auf denen Sie das Dateisystem mounten möchten.
- Jede EC2-Instance, die das Dateisystem mountet, muss eine Sicherheitsgruppe haben, die den ausgehenden Zugriff auf das Mounting-Ziel auf dem NFS-Port erlaubt.

Informationen zum Ändern der Sicherheitsgruppen, die den Mount-Zielen Ihrer EFS-Dateisysteme zugeordnet sind, finden Sie unter [Erstellen und Verwalten von Mount-Zielen und Sicherheitsgruppen](#).

Weitere Informationen zu Sicherheitsgruppen finden Sie in [Amazon EC2-Sicherheitsgruppen](#) im Amazon EC2-Benutzerhandbuch für Linux-Instances.

Erstellen von Sicherheitsgruppen mit der AWS Management Console

Sie können die verwenden AWS Management Console , um Sicherheitsgruppen in Ihrer VPC zu erstellen. Zur Verbindung Ihres Amazon EFS-Dateisystems mit Ihrer Amazon-EC2-Instance müssen Sie zwei Sicherheitsgruppen erstellen: eine für Ihre Amazon-EC2-Instance und eine andere für Ihr Amazon-EFS-Mount-Ziel.

1. Erstellen Sie zwei Sicherheitsgruppen in Ihrer VPC. Eine Anleitung finden Sie unter [Erstellen einer Sicherheitsgruppe](#) im Amazon-VPC-Benutzerhandbuch.
2. Überprüfen Sie in der VPC-Konsole die Standardregeln für diese Sicherheitsgruppen. Beide Sicherheitsgruppen sollten nur über eine Regel verfügen, die ausgehenden Datenverkehr zulässt.

3. Sie müssen wie folgt zusätzlichen Zugriff auf die Sicherheitsgruppen autorisieren:
 - a. Fügen Sie der EC2-Sicherheitsgruppe eine Regel hinzu, um SSH-Zugriff auf die Instance auf Port 22 wie folgt zu ermöglichen. Dies ist nützlich, wenn Sie einen SSH-Client wie PuTTY verwenden möchten, um eine Verbindung mit Ihrer EC2-Instance über eine Terminalschnittstelle herzustellen und zu verwalten. Optional können Sie die Adresse der Source (Quelle) einschränken.

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Anywhere	0.0.0.0/0, ::/0

Add Rule

NOTE: Any edits made on existing rules will result in the edited rule being deleted and a new rule created with the new details. This will cause traffic that depends on that rule to be dropped for a very brief period of time until the new rule can be created.

Cancel Save

Anweisungen dazu finden Sie unter [Hinzufügen, Entfernen und Aktualisieren von Regeln](#) von Regeln im Amazon-VPC-Benutzerhandbuch.

- b. Fügen Sie eine Regel für die Sicherheitsgruppe des Mount-Ziels hinzu, um den eingehenden Zugriff von der EC2-Sicherheitsgruppe auf TCP-Port 2049 zu erlauben. Die Sicherheitsgruppe in der Spalte Quelle ist die Sicherheitsgruppe, die der EC2-Instance zugeordnet ist.

Type	Protocol	Port Range	Source	Description
NFS	TCP	2049	Custom	sg-xxxxxxxxxxxx

Add Rule

NOTE: Any edits made on existing rules will result in the edited rule being deleted and a new rule created with the new details. This will cause traffic that depends on that rule to be dropped for a very brief period of time until the new rule can be created.

Cancel Save

Um die Sicherheitsgruppen anzuzeigen, die Ihren Dateisystem-Mount-Zielen zugeordnet sind, wählen Sie in der EFS-Konsole auf der Seite mit den Dateisystemdetails die Registerkarte Netzwerk aus. Weitere Informationen finden Sie unter [Erstellen und Verwalten von Mount-Zielen und Sicherheitsgruppen](#).

 Note

Sie müssen keine ausgehende Regel hinzufügen, da die Standardausgangsregel jeden Datenverkehr nach außen zulässt. (Wenn Sie diese Standardausgangsregel entfernt haben, fügen Sie eine ausgehende Regel hinzu, um eine TCP-Verbindung auf dem NFS-Port zu öffnen, und identifizieren Sie die Sicherheitsgruppe des Mount-Ziels als Ziel.

- Überprüfen Sie, ob beide Sicherheitsgruppen jetzt den eingehenden und ausgehenden Zugriff erlauben, wie in diesem Abschnitt beschrieben.

Erstellen von Sicherheitsgruppen mit der AWS CLI

Ein Beispiel, das zeigt, wie Sicherheitsgruppen mithilfe der erstellt werden AWS CLI, finden Sie unter [Schritt 1: Erstellen Amazon EC2 EC2-Ressourcen](#).

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 - AWS SDKs oder die Amazon-EFS-API direkt verwenden. 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 Dateisystemdatenzugriffs](#).

 Note

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

Erstellen einer Dateisystemrichtlinie (Konsole)

- Öffnen Sie die Amazon Elastic File System-Konsole unter <https://console.aws.amazon.com/efs/>.
- 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. Die Detailseite für dieses Dateisystem wird angezeigt.
4. Wählen Sie Dateisystemrichtlinie und dann Bearbeiten aus. Die Seite File system policy (Dateisystemrichtlinie) wird angezeigt.

File system policy

Policy options

Select one or more of these common policy options, or create a custom policy using the editor. [Learn more](#)

☐ Prevent root access by default*
☐ Enforce read-only access by default*
☐ Prevent anonymous access
☐ Enforce in-transit encryption for all clients

* Identity-based policies can override these default permissions.

▼ **Grant additional permissions**

Grant file system permissions to additional AWS IAM principals. [Learn more](#)

Principal ARN	Permissions	
<input type="text" value="Principal ARN"/>	<input type="text" value="Read Access"/>	<input type="button" value="x"/>

Policy editor (JSON)

```

1 {
2   "Version": "2012-10-17",
3   "Id": "efs-policy-wizard-a5ab3f12-0036-457f-92fe-4047cb9bf354",
4   "Statement": [
5     {
6       "Sid": "efs-statement-9251bda-3e99-4a9b-875a-a9fe9302b6d8",
7       "Effect": "Allow",
8       "Principal": {
9         "AWS": "*"
10      },
11      "Action": [
12        "elasticfilesystem:ClientRootAccess",
13        "elasticfilesystem:ClientWrite",
14        "elasticfilesystem:ClientMount"
15      ],
16      "Condition": {
17        "Bool": {
18          "elasticfilesystem:AccessedViaMountTarget": "true"
19        }
20      }
21    },
22    {
23      "Sid": "efs-statement-7371b922-c09e-46ce-a61f-44f90309c28e",
24      "Effect": "Allow",
25      "Principal": {
26        "AWS": "*"
27      },
28      "Action": [
29        "elasticfilesystem:ClientMount"
30      ]
31    }
32  ]
33 }
```

Manual changes will prevent the use of the policy options on the left until the editor is cleared.

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 zusätzlichen IAM-Prinzipalen, einschließlich einer anderen, Dateisystemberechtigungen zu erteilen 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.

Erstellen einer Dateisystemrichtlinie (CLI)

Im folgenden Beispiel erstellt der [put-file-system-policy](#) CLI-Befehl eine Dateisystemrichtlinie, die dem angegebenen AWS-Konto schreibgeschützten Zugriff auf das EFS-Dateisystem erlaubt. 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"
      }
    }
  ]
}'
```

```
{
  "FileSystemId": "fs-01234567",
  "Policy": "{
    \"Version\" : \"2012-10-17\",
    \"Id\" : \"1\",
    \"Statement\" : [
      {
```

```
    "Sid" : "efs-statement-7c8d8687-1c94-4fdc-98b7-555555555555",
    "Effect" : "Allow",
    "Principal" : {
        "AWS" : "arn:aws:iam::111122223333:root"
    },
    "Action" : [
        "elasticfilesystem:ClientMount"
    ],
    "Resource" : "arn:aws:elasticfilesystem:us-east-2:555555555555:file-system/
fs-01234567"
    }
  ]
}
```

Erstellen und Löschen von Zugangspunkten

Sie können Amazon-EFS-Zugriffspunkte mithilfe der AWS Management Console oder der erstellen AWS CLI. Sie können Zugriffspunkte auch programmgesteuert erstellen, indem Sie die - AWS SDKs oder die Amazon-EFS-API direkt verwenden. Sie können einen Zugangspunkt nicht mehr ändern, nachdem er einmal erstellt wurde. Ein Dateisystem kann maximal 1 000 Zugangspunkte haben. Weitere Hinweise zu EFS-Zugangspunkten finden Sie unter [Arbeiten mit Amazon EFS Access Points](#).

In den folgenden Verfahren wird beschrieben, wie ein Zugangspunkt mithilfe der Konsole und der AWS CLI erstellt wird.

Erstellen eines Zugangspunkts (Konsole)

Sie können Amazon-EFS-Zugriffspunkte mithilfe der AWS Management Console, der AWS Command Line Interface (AWS CLI) und der Amazon-EFS-API und SDKs erstellen und löschen. Sie können einen Zugangspunkt nicht mehr ändern, nachdem er einmal erstellt wurde. Ein Dateisystem kann maximal 1 000 Zugangspunkte haben.

Note

Wenn mehrere Anfragen zum Erstellen von Zugangspunkten auf demselben Dateisystem schnell hintereinander gesendet werden und sich das Dateisystem dem Grenzwert von 1 000 Zugangspunkten 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.

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.

Create access point

An access point is an application-specific entry point into an EFS file system that makes it easier to manage application access to shared datasets. [Learn more](#) 

Details


File system

Choose the file system to which your access point is associated.

Name - optional


Maximum of 256 Unicode letters, whitespace, and numbers, plus + - = . _ : /

Root directory path - optional

Connections use the specified path as the file system's virtual root directory [Learn more](#) 

Example: "/foo/bar"

POSIX user - optional

The full POSIX identity on the access point that is used for all file operations by NFS clients. [Learn more](#) 

User ID

POSIX user ID used for all file system operations using this access point.

Accepts values from 0 to 4294967295

Group ID

POSIX group ID used for all file system operations using this access point.


Accepts values from 0 to 4294967295

Secondary group IDs

Secondary POSIX group IDs used for all file system operations using this access point.

A comma-separated list of valid POSIX group IDs

Root directory creation permissions - optional

EFS will automatically create the specified root directory with these permissions if the directory does not already exist. [Learn more](#) 

Owner user ID

Owner user ID for the access point's root directory, if the directory does not already exist.

Accepts values from 0 to 4294967295

Owner group ID

Owner group ID for the access point's root directory, if the directory does not already exist.

- 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 [Ein Stammverzeichnis mit einem Access Point erzwingen](#).
- 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 [Durchsetzung einer Benutzeridentität mithilfe eines Access Points](#).
- 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 Gruppen-IDs – Geben Sie eine optionale, durch Kommas getrennte Liste von sekundären Gruppen-IDs ein.
- 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 [Ein Stammverzeichnis mit einem Access Point erzwingen](#).

 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 Benutzergruppe 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.

Erstellen eines Zugangspunkts (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,
  "OwnerUid": 11,
  "Permissions": "775"
},
"Path": "/efs/mobileapp/east",
},
"Tags": []
}
```

Note

Wenn mehrere Anfragen zum Erstellen von Zugangspunkten auf demselben Dateisystem schnell hintereinander gesendet werden und sich das Dateisystem dem Grenzwert von 1 000 Zugangspunkten 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.

Löschen eines Zugriffspunkts

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

Löschen eines Zugangspunkts (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.

Löschen eines Zugangspunkts (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
```

Markieren der Amazon-EFS-Ressourcen

Um Sie bei der Verwaltung Ihrer Amazon-EFS-Ressourcen zu unterstützen, können Sie jeder Ressource eigene Metadaten in Form von Tags zuweisen. Mit 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.

Mit 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.

Markieren Ihrer -Ressourcen mit Tags (Markierungen)

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

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.

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

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

CLI-Befehl	Beschreibung	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

Tag (Markierung)-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 zur AWS 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. Keine anderen AWS-Konto haben Zugriff auf diese Tags. Für die Tag-basierte Zugriffskontrolle auf gemeinsam genutzte Ressourcen AWS-Konto muss jedes seinen eigenen Satz von Tags zuweisen, um den Zugriff auf die Ressource zu steuern.

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

Verwenden von Dateisystemen in Amazon EFS

Amazon Elastic File System bietet eine standardmäßige Dateisystemschnittstelle, die die vollständige Semantik des Dateisystemzugriffs unterstützt. Mit Network File System (NFS) Version 4.1 (NFSv4.1) können Sie Ihr Amazon EFS-Dateisystem auf jeder Linux-basierten Amazon Elastic Compute Cloud (Amazon EC2) -Instance mounten. Nach dem Mounten Ihres Systems können Sie mit den Dateien und Verzeichnissen wie in einem lokalen Dateisystem arbeiten. Weitere Informationen zum Mounten finden Sie unter [Mounting von EFS-Dateisystemen](#).

Nachdem Sie ein Dateisystem erstellt und auf Ihre EC2-Instance aufgespielt haben, benötigen Sie Informationen zum Verwalten von Berechtigungen für Benutzer, Gruppen und zugehörige Ressourcen auf NFS-Level, um das Dateisystem effektiv zu verwenden. Beim ersten Erstellen Ihres Dateisystems ist nur ein Stammverzeichnis unter / vorhanden. Standardmäßig hat nur der Root-Benutzer (UID 0) Berechtigungen. read-write-execute Damit auch andere Benutzer das Dateisystem ändern können, muss Ihnen der Root-Benutzer ausdrücklich Zugriff gewähren. Mithilfe von EFS-Zugriffspunkten können Sie Verzeichnisse bereitstellen, die von einer bestimmten Anwendung aus schreibbar sind. Weitere Informationen finden Sie unter [Mit Benutzern, Gruppen und Berechtigungen auf Network File System-\(NFS-\)Level arbeiten](#) und [Arbeiten mit Amazon EFS Access Points](#).

Verwandte Themen

[Amazon EFS – Funktionsweise](#)

[Erste Schritte](#)

[Anleitungen](#)

Verwenden der amazon-efs-utils Tools

Das `amazon-efs-utils`-Paket ist eine Open-Source-Sammlung von Amazon EFS-Tools, die auch als Amazon EFS-Client bezeichnet wird. Im Folgenden finden Sie eine Beschreibung des Amazon EFS-Clients. Der Amazon EFS-Client enthält die Amazon EFS-Mountinghilfe, die das Mounten von EFS-Dateisystemen erleichtert. Die Verwendung des EFS-Clients ermöglicht die Verwendung von Amazon CloudWatch zur Überwachung des Mountingstatus eines EFS-Dateisystems. Sie müssen den Amazon EFS-Client auf einer Amazon EC2-Instance installieren, bevor Sie ein EFS-Dateisystem mounten können.

Themen

- [Übersicht](#)
- [Verwendung von AWS Systems Manager zur automatischen Installation oder Aktualisierung von Amazon EFS-Clients](#)
- [Manuelles Installieren des Amazon EFS-Clients](#)
- [Installation von botocore](#)
- [Upgraden von stunnel](#)

Übersicht

Der Amazon EFS-Client (`amazon-efs-utils`) ist eine Open-Source-Sammlung von Amazon EFS-Tools. Für die Nutzung des Amazon-EFS-Clients fallen keine zusätzlichen Kosten an, die Sie unter GitHub <https://github.com/aws/efs-utils> herunterladen können. Das Paket `amazon-efs-utils` ist in den Amazon Linux-Paket-Repositorys verfügbar. Sie können das Paket jedoch auch für andere Linux-Distributionen erstellen und installieren. Sie können AWS Systems Manager auch verwenden, um das Paket automatisch zu installieren oder zu aktualisieren. Weitere Informationen finden Sie unter [Verwendung von AWS Systems Manager zur automatischen Installation oder Aktualisierung von Amazon EFS-Clients](#).

Note

Das `amazon-efs-utils` Paket ist auf Amazon Linux und Amazon Linux 2 Amazon Machine Images (AMIs) vorinstalliert.

Der Amazon EFS-Client umfasst eine Mountinghilfe und Tools, die die Verschlüsselung von Daten während der Übertragung für Amazon EFS-Dateisysteme erleichtern. Eine Mountinghilfe ist ein Programm, das zum Mounten eines bestimmten Dateisystems eingesetzt wird. Wir empfehlen Ihnen, die Mountinghilfe des Amazon EFS-Clients zu verwenden, um Ihre Amazon EFS-Dateisysteme zu mounten. Die Verwendung des Amazon EFS-Clients vereinfacht das Mounten von EFS-Dateisystemen und kann die Dateisystemleistung verbessern. Weitere Informationen zur Verwendung von EFS-Client und der Mountinghilfe finden Sie unter [Mounting von EFS-Dateisystemen](#).

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 Sie die Amazon EFS-Mountinghilfe mit Transport Layer Security (TLS) verwenden, wird die Mountinghilfe standardmäßig eine Prüfung des Hostnamens des Zertifikats durchgeführt. Die Amazon EFS-Mountinghilfe verwendet das `stunnel`-Programm für die 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 Mounting eines Amazon EFS-Dateisystems mit TLS fehl.

Informationen dazu, wie Sie die Stunnel-Version Ihres Systems nach der Installation von `amazon-efs-utils` aktualisieren, finden Sie unter [Upgraden von stunnel](#).

Sie können verwenden AWS Systems Manager, um Amazon-EFS-Clients zu verwalten und die Aufgaben zu automatisieren, die zum Installieren oder Aktualisieren des `amazon-efs-utils` Pakets auf Ihren EC2-Instances erforderlich sind. Weitere Informationen finden Sie unter [Verwendung von AWS Systems Manager zur automatischen Installation oder Aktualisierung von Amazon EFS-Clients](#).

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
Amazon Linux 2017.09	RPM	upstart
Amazon Linux 2	RPM	systemd
CentOS 7, 8	RPM	systemd
Debian 9, 10	deb	systemd
Fedora 28–32	RPM	systemd
macOS Big Sur		launchd
macOS Monterey		launchd
macOS Ventura		launchd
OpenSUSE Leap, Tumbleweed	RPM	systemd
Oracle 8	rpm	systemd
Red Hat Enterprise Linux (RHEL) 7, 8, 9	rpm	systemd
SUSE Linux Enterprise Server (SLES) 12, 15	RPM	systemd
Ubuntu 16.04 LTS, 18.04 LTS, 20.04 LTS	deb	systemd

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

In den folgenden Abschnitten erfahren Sie, wie Sie den Amazon EFS-Client auf Ihren EC2-Linux- und macOS-Instances installieren.

Verwendung von AWS Systems Manager zur automatischen Installation oder Aktualisierung von Amazon EFS-Clients

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 auf AWS anzeigen und steuern können. Mit AWS Systems Manager können Sie die Aufgaben automatisieren, die zur Installation oder Aktualisierung des `amazon-efs-utils`-Pakets auf Ihren 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 Vorgang, um Ihre Amazon EC2-Instances in einem definierten Status 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 Amazon- CloudWatch Protokollen für den Mounting-Status des Dateisystems zu automatisieren und auf die neueste Version für ausgewählte Linux-Distributionen `stunnel` zu aktualisieren. 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 von botocore](#) beschrieben. Der Amazon EFS-Client verwendet `botocore`, um den Mounting-Status des EFS-Dateisystems zu überwachen.
- Ermöglicht die Überwachung des EFS-Dateisystem-Mounting-Status in - CloudWatch Protokollen durch Aktualisierung von `efs-utils.conf`. Weitere Informationen finden Sie unter [Überwachung des Erfolgs- oder Fehlerstatus des Mount-Versuchs](#).

- 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 Distributor unterstützte Betriebssysteme

Auf Ihren EC2-Instances muss eines der folgenden Betriebssysteme laufen, damit Sie mit AWS Systems Manager den Amazon EFS-Client automatisch aktualisieren oder installieren können.

Plattform	Plattformversion	Architektur
Amazon Linux	2017.09, 2018.03	x86_64
Amazon Linux 2	2.0	x86_64, arm64 (Amazon Linux 2, A1 Instance-Typen)
CentOS	7, 8	x86_64
Red Hat Enterprise Linux (RHEL)	7, 8	x86_64, arm64 (RHEL 7.6 und neuer, A1-Instance-Typen)
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)

So verwenden Sie , AWS Systems Manager um automatisch zu installieren oder zu aktualisieren `amazon-efs-utils`

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)-Instance-Profil mit den erforderlichen Berechtigungen.
2. Konfigurieren Sie eine Zuordnung (einschließlich des Zeitplans), die für die Installation oder Aktualisierung durch den State Manager verwendet wird.

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

Standardmäßig AWS Systems Manager ist nicht berechtigt, 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 Informationen zur IAM-Rolle beim Start an eine EC2-Instance übergibt.

Verwenden Sie die AmazonElasticFileSystemsUtils AWS verwaltete Berechtigungsrichtlinie, um den Rollen die entsprechenden Berechtigungen zuzuweisen. Sie können eine neue Rolle für Ihr Instance-Profil erstellen oder die AmazonElasticFileSystemsUtils Berechtigungsrichtlinie zu einer vorhandenen Rolle hinzufügen. Sie müssen dann dieses Instance-Profil verwenden, um Ihre Amazon EC2-Instances zu starten. Weitere Informationen finden Sie unter [Schritt 4: Erstellen eines IAM-Instance-Profils für Systems Manager](#).

Schritt 2: Konfigurieren Sie eine Zuordnung, die von State Manager für die Installation oder Aktualisierung des Amazon EFS-Clients verwendet wird

Das amazon-efs-utils Paket ist im Lieferumfang von Distributor enthalten und kann sofort auf verwalteten EC2-Instances bereitgestellt werden. Um die neueste Version von amazon-efs-utils zu sehen, die zur Installation verfügbar ist, können Sie die AWS Systems Manager-Konsole oder Ihr bevorzugtes AWS-Befehlszeilentool verwenden. Um auf Distributor zuzugreifen, öffnen Sie <https://console.aws.amazon.com/systems-manager/> und wählen Sie im linken Navigationsbereich Distributor aus. Suchen Sie AmazonEFSUtils im Abschnitt Eigentum von Amazon. Wählen Sie AmazonEFSUtils, 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 zum automatischen Installieren oder Aktualisieren des amazon-efs-utils Pakets auf Instances mithilfe der Systems Manager-Konsole finden Sie unter [Planen einer Paketinstallation oder -aktualisierung \(Konsole\)](#). 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 ist die empfohlene Einstellung Alle Instances auswählen, um alle neuen und vorhandenen EC2-Instances als Ziele für die automatische Installation oder Aktualisierung von AmazonEFSUtils 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 ist die empfohlene Einstellung für AmazonEFSUtils alle 30 Tage. Sie können die Steuerelemente verwenden, um einen Cron- oder Ratenplan für die Vereinigung zu erstellen.

Wie Sie AWS Systems Manager verwenden, um mehrere Amazon EFS-Dateisysteme in mehrere EC2-Instances einzuhängen, erfahren Sie unter [Mounten von EFS auf mehreren EC2-Instances mit AWS Systems Manager](#).

Manuelles Installieren des Amazon EFS-Clients

Sie können den Amazon EFS-Client manuell auf Ihren Amazon EC2-Linux-Instances mit Amazon Linux und Amazon Linux 2 und anderen unterstützten Linux-Distributionen sowie auf EC2-Mac-Instances mit macOS Big Sur, macOS Monterey und macOS Ventura installieren. Die amazon-efs-utils Installationsverfahren für diese Betriebssysteme werden in den folgenden Abschnitten beschrieben.

Themen

- [Installation des Amazon EFS-Clients auf Amazon Linux und Amazon Linux 2](#)
- [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 Linux und Amazon Linux 2

Das `amazon-efs-utils` Paket für die Installation auf Amazon Linux und Amazon Linux 2 ist an den folgenden Orten verfügbar:

- Die Amazon Linux und Amazon Linux 2 Amazon Machine Image (AMI) Paket-Repositories.
- Das AWS [efs-utils](#) GitHub -Repository.

Im Folgenden wird beschrieben, wie Sie `amazon-efs-utils` aus den Amazon Linux- und Amazon Linux 2 AMI-Paket-Repositories installieren.

Sie können auch `amazon-efs-utils` aus dem AWS [efs-utils](#) GitHub -Repository installieren oder aktualisieren. Anweisungen zur Installation und Aktualisierung des Amazon-EFS-Clients mit GitHub finden Sie unter [So erstellen und installieren Sie amazon-efs-utils als RPM-Paket für Amazon Linux, Amazon Linux 2](#).

Informationen zur Installation des Amazon-EFS-Clients auf anderen Linux-Distributionen finden Sie unter [Installation des Amazon EFS-Clients auf anderen Linux-Distributionen](#).

Note

Wenn Sie AWS Direct Connect verwenden, finden Sie eine Installationsanleitung unter [Exemplarische Vorgehensweise: Erstellen und Bereitstellen eines lokalen Dateisystems mit VPN AWS Direct Connect](#).

So installieren Sie das **`amazon-efs-utils`** Paket unter Amazon Linux 2 und Amazon Linux AMI

1. Stellen Sie sicher, dass Sie eine Amazon Linux- oder Amazon Linux 2-EC2-Instance erstellt haben. Weitere Informationen finden Sie unter [Schritt 1: Starten einer Instance](#) im Benutzerhandbuch zu Amazon EC2 für Linux-Instances.
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 hierzu finden Sie unter [Herstellen einer Verbindung mit Ihrer Linux-Instance mit SSH](#) im Amazon-EC2-Benutzerhandbuch für Linux-Instances.
3. Zur Installation des `amazon-efs-utils` Pakets führen Sie den folgenden Befehl aus:

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

Nächste Schritte

Fahren Sie nach der Installation von `amazon-efs-utils` auf Ihrer EC2-Instance mit den nächsten Schritten zum Mounten Ihres Dateisystems fort:

- [Installieren boto3](#) Sie , damit Sie Amazon verwenden können CloudWatch , um den Mounting-Status Ihres Dateisystems zu überwachen.
- [Führen Sie ein Upgrade auf die neueste Version von durch stunnel](#), um die Verschlüsselung von Daten während der Übertragung zu aktivieren.
- [Mounten Sie Ihr Dateisystem](#) mit der EFS-Mountinghilfe ein.

Installation des Amazon EFS-Clients auf anderen Linux-Distributionen

Wenn Sie das `amazon-efs-utils` Paket nicht aus den Amazon Linux- oder Amazon Linux 2 AMI-Paket-Repositorys abrufen möchten, ist es auch auf verfügbar GitHub.

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

So erstellen und installieren Sie **amazon-efs-utils** als RPM-Paket für Amazon Linux, Amazon Linux 2 und andere Linux-Distributionen als OpenSUSE oder SLES

So klonen Sie **amazon-efs-utils** von GitHub

1. Stellen Sie eine Verbindung mit der EC2-Instance mithilfe von Secure Shell (SSH) her und melden Sie sich an. Weitere Informationen finden Sie unter [Herstellen einer Verbindung mit Ihrer Linux-Instance mit SSH](#) im Amazon-EC2-Benutzerhandbuch für Linux-Instances.
2. Installieren Sie `git` mit dem folgenden Befehl:

```
sudo yum -y install git
```

3. Klonen GitHub Sie mit dem folgenden Befehl `amazon-efs-utils` aus .

```
git clone https://github.com/aws/efs-utils
```

So erstellen und installieren Sie das **amazon-efs-utils**-RPM-Paket

1. Öffnen Sie ein Terminal auf Ihrem Client und navigieren Sie zu dem Verzeichnis, das das `amazon-efs-utils` Paket enthält.

```
cd /path/efs-utils
```

2. Installieren Sie den Bash-Befehl `make`, wenn Ihr Betriebssystem ihn nicht bereits enthält, wie folgt.

```
sudo yum -y install make
```

3. Installieren Sie das `rpm-build` Paket, falls es nicht bereits installiert ist, mit dem folgenden Befehl:

```
sudo yum -y install rpm-build
```

4. Aktualisieren Sie das `amazon-efs-utils`-Paket mit dem folgenden Befehl:

```
sudo make rpm
```

5. Installieren Sie das `amazon-efs-utils` Paket mit dem folgenden Befehl:

```
sudo yum -y install ./build/amazon-efs-utils*.rpm
```

Nächste Schritte

Fahren Sie nach der Installation von `amazon-efs-utils` auf Ihrer EC2-Instance mit den nächsten Schritten zum Mounten Ihres Dateisystems fort:

- [Installieren `botocore`](#) Sie , damit Sie Amazon verwenden können CloudWatch , um den Mounting-Status Ihres Dateisystems zu überwachen.

- [Führen Sie ein Upgrade auf die neueste Version von durch stunnel](#), um die Verschlüsselung von Daten während der Übertragung zu aktivieren.
- [Mounten Sie Ihr Dateisystem](#) mit der EFS-Mountinghilfe ein.

So erstellen und installieren Sie **amazon-efs-utils** als RPM-Paket für OpenSUSE und SLES

So klonen Sie **amazon-efs-utils** von GitHub

1. Stellen Sie eine Verbindung mit der EC2-Instance mithilfe von Secure Shell (SSH) her und melden Sie sich an. Weitere Informationen finden Sie unter [Herstellen einer Verbindung mit Ihrer Linux-Instance mit SSH](#) im Amazon-EC2-Benutzerhandbuch für Linux-Instances.
2. Installieren Sie zypper mit dem folgenden Befehl:

```
sudo zypper refresh
```

3. Installieren Sie das rpm-build-Paket und den Bash-Befehl make, falls beides noch nicht installiert ist, mit dem folgenden Befehl:

```
sudo zypper install -y git rpm-build make
```

- a. Wenn Sie unter OpenSUSE eine Fehlermeldung ähnlich der folgenden erhalten:

```
File './suse/noarch/bash-completion-2.11-2.1.noarch.rpm' not found on medium  
'http://download.opensuse.org/tumbleweed/repo/oss/'
```

Führen Sie den folgenden Befehl aus, um das Repo OSS und NON-OSS erneut hinzuzufügen.

```
sudo zypper ar -f -n OSS http://download.opensuse.org/tumbleweed/repo/oss/ OSS  
sudo zypper ar -f -n NON-OSS http://download.opensuse.org/tumbleweed/repo/non-  
oss/ NON-OSS  
sudo zypper refresh
```

- b. Führen Sie das Git-Install-Skript erneut aus:

```
sudo zypper install -y git rpm-build make
```

4. Klonen GitHub Sie mit dem folgenden Befehl **amazon-efs-utils** aus .

```
git clone https://github.com/aws/efs-utils
```

So erstellen und installieren Sie das **amazon-efs-utils**-RPM-Paket

1. Öffnen Sie ein Terminal auf Ihrem Client und navigieren Sie zu dem Verzeichnis, das das amazon-efs-utils Paket enthält.

```
cd /path/efs-utils
```

2. Aktualisieren Sie das amazon-efs-utils-Paket mit dem folgenden Befehl:

```
make rpm
```

3. Installieren Sie das amazon-efs-utils Paket mit dem folgenden Befehl:

```
sudo zypper --no-gpg-checks install -y build/amazon-efs-utils*.rpm
```

Nächste Schritte

Fahren Sie nach der Installation von amazon-efs-utils auf Ihrer EC2-Instance mit den nächsten Schritten zum Mounten Ihres Dateisystems fort:

- [Installieren boto3](#) Sie , damit Sie Amazon verwenden können CloudWatch , um den Mounting-Status Ihres Dateisystems zu überwachen.
- [Führen Sie ein Upgrade auf die neueste Version von durch stunnel](#), um die Verschlüsselung von Daten während der Übertragung zu aktivieren.
- [Mounten Sie Ihr Dateisystem](#) mit der EFS-Mountinghilfe ein.

So erstellen und installieren Sie amazon-efs-utils als Debian-Paket für Ubuntu und Debian

So klonen Sie **amazon-efs-utils** von GitHub

1. Stellen Sie eine Verbindung mit der EC2-Instance mithilfe von Secure Shell (SSH) her und melden Sie sich an. Weitere Informationen finden Sie unter [Herstellen einer Verbindung mit Ihrer Linux-Instance mit SSH](#) im Amazon-EC2-Benutzerhandbuch für Linux-Instances.
2. (Optional) Wenden Sie Updates an, bevor Sie das Paket mit dem folgenden Befehl installieren:

```
sudo apt-get update
```

Installieren Sie Updates nach Bedarf.

3. Installieren Sie `git` und `binutils`, indem Sie den folgenden Befehl verwenden. `binutils` ist für die Erstellung von DEB-Paketen erforderlich,

```
sudo apt-get -y install git binutils
```

4. Klonen GitHub Sie mit dem folgenden Befehl `amazon-efs-utils` aus .

```
git clone https://github.com/aws/efs-utils
```

So erstellen und installieren Sie das **amazon-efs-utils**-DEB-Paket

1. Navigieren Sie zu dem Verzeichnis, das das `amazon-efs-utils`-Paket enthält.

```
cd /path/efs-utils
```

2. Erstellen Sie `amazon-efs-utils` mit dem folgenden Befehl:

```
./build-deb.sh
```

3. Installieren Sie das Paket mit dem folgenden Befehl:

```
sudo apt-get -y install ./build/amazon-efs-utils*.deb
```

Nächste Schritte

Fahren Sie nach der Installation von `amazon-efs-utils` auf Ihrer EC2-Instance mit den nächsten Schritten zum Mounten Ihres Dateisystems fort:

- [Installieren boto3](#) Sie , damit Sie Amazon verwenden können CloudWatch , um den Mounting-Status Ihres Dateisystems zu überwachen.
- [Führen Sie ein Upgrade auf die neueste Version von durch stunnel](#), um die Verschlüsselung von Daten während der Übertragung zu aktivieren.
- [Mounten Sie Ihr Dateisystem](#) mit der EFS-Mountinghilfe ein.

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.

So installieren Sie das **amazon-efs-utils**-Paket:

1. Stellen Sie sicher, dass Sie eine EC2-Mac-Instance erstellt haben, auf der eines der unterstützten Mac-Betriebssysteme ausgeführt wird:

- macOS Big Sur
- macOS Monterey
- macOS Ventura

Informationen dazu finden Sie unter [Schritt 1: Starten einer Instance](#) im Amazon EC2-Benutzerhandbuch für Mac-Instances.

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 hierzu finden Sie unter [Verbindung zu Ihrer Instance mit SSH](#) im Amazon EC2-Benutzerhandbuch für Mac-Instances.
3. Führen Sie zum Installieren des `amazon-efs-utils` den folgenden Befehl aus.

```
brew install amazon-efs-utils
```

Note

Das System antwortet mit Anweisungen zur Konfiguration der Mountinghilfe und zur Aktivierung des Watchdog-Vorgangs, die in den nächsten beiden Schritten enthalten sind. Wenn Sie die Anweisungen später anzeigen möchten, können Sie den folgenden Befehl ausführen.

```
brew info amazon-efs-utils
```

4. Stellen Sie sicher, dass die EFS-Mountinghilfe in `amazon-efs-utils` für den Mounting-Befehl zugänglich ist. Der Befehl, den Sie ausführen müssen, hängt von der EC2-Mac-Instance ab, auf der Sie das Paket installieren.

- Wenn Sie das Paket auf einem EC2-x86-Mac (mac1.metal) installieren, führen Sie den folgenden Befehl aus:

```
sudo mkdir -p /Library/Filesystems/efs.fs/Contents/Resources
sudo ln -s /usr/local/bin/mount.efs /Library/Filesystems/efs.fs/Contents/
Resources/mount_efs
```

- Wenn Sie das Paket auf EC2 M1 Mac (mac2.metal) installieren, führen Sie den folgenden Befehl aus:

```
sudo mkdir -p /Library/Filesystems/efs.fs/Contents/Resources
sudo ln -s /opt/homebrew/bin/mount.efs /Library/Filesystems/efs.fs/Contents/
Resources/mount_efs
```

5. Aktivieren Sie den Watchdog-Vorgang (amazon-efs-mount-watchdog), der den Zustand von TLS-Mounts in Ihrem EFS-Dateisystem überwacht. Der Befehl, den Sie ausführen müssen, hängt von der EC2-Mac-Instance ab, auf der Sie das Paket installieren.

- Wenn Sie das Paket auf einem EC2-x86-Mac (mac1.metal) installieren, führen Sie den folgenden Befehl aus:

```
sudo cp /usr/local/Cellar/amazon-efs-utils/<version>/libexec/amazon-efs-mount-
watchdog.plist /Library/LaunchAgents
sudo launchctl load /Library/LaunchAgents/amazon-efs-mount-watchdog.plist
```

- Wenn Sie das Paket auf EC2 M1 Mac (mac2.metal) installieren, führen Sie den folgenden Befehl aus:

```
sudo cp /opt/homebrew/Cellar/amazon-efs-utils/<version>/libexec/amazon-efs-mount-
watchdog.plist /Library/LaunchAgents
sudo launchctl load /Library/LaunchAgents/amazon-efs-mount-watchdog.plist
```

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 botocore](#) Sie , damit Sie Amazon verwenden können CloudWatch , um den Mounting-Status Ihres Dateisystems zu überwachen.

- [Führen Sie ein Upgrade auf die neueste Version von durch stunnel](#), um die Verschlüsselung von Daten während der Übertragung zu aktivieren.
- [Mounten Sie Ihr Dateisystem](#) mit der EFS-Mountinghilfe ein.

Installation von **botocore**

Der Amazon EFS-Client verwendet botocore, um mit anderen AWS-Services zu interagieren. Sie ist erforderlich, wenn Sie den Erfolg oder Misserfolg des Mountingversuchs für Ihre Amazon-EFS-Dateisysteme in - CloudWatch Protokollen überwachen möchten. Weitere Informationen finden Sie unter [Überwachung des Erfolgs- oder Fehlerstatus des Mount-Versuchs](#). Dieser Abschnitt beschreibt, wie Sie botocore auf einer Amazon EC2-Instance installieren und aktualisieren.

Zur Installation von **botocore** als RPM-Paket

1. Führen Sie zum Installieren des wget den folgenden Befehl aus.

```
sudo yum -y install wget
```

2. Verwenden Sie das folgende Skript, um die entsprechende Version des pip-Paketmanagers zu installieren.

```
if [[ "$(python3 -V 2>&1)" =~ ^(Python 3.6.*) ]]; then
    sudo wget https://bootstrap.pypa.io/pip/3.6/get-pip.py -O /tmp/get-pip.py
elif [[ "$(python3 -V 2>&1)" =~ ^(Python 3.5.*) ]]; then
    sudo wget https://bootstrap.pypa.io/pip/3.5/get-pip.py -O /tmp/get-pip.py
elif [[ "$(python3 -V 2>&1)" =~ ^(Python 3.4.*) ]]; then
    sudo wget https://bootstrap.pypa.io/pip/3.4/get-pip.py -O /tmp/get-pip.py
else
    sudo wget https://bootstrap.pypa.io/get-pip.py -O /tmp/get-pip.py
fi
```

3. Führen Sie die folgenden Befehle aus, um botocore zu installieren.

```
sudo python3 /tmp/get-pip.py
sudo pip3 install botocore
```

Oder

```
sudo /usr/local/bin/pip3 install botocore
```

So installieren Sie botocore als DEB-Paket

1. Führen Sie die folgenden Befehle aus, um wget zu installieren.

```
sudo apt-get update
sudo apt-get -y install wget
```

2. Verwenden Sie das folgende Skript, um die entsprechende Version des pip-Paketmanagers zu installieren.

```
if echo $(python3 -V 2>&1) | grep -e "Python 3.6"; then
    sudo wget https://bootstrap.pypa.io/pip/3.6/get-pip.py -O /tmp/get-pip.py
elif echo $(python3 -V 2>&1) | grep -e "Python 3.5"; then
    sudo wget https://bootstrap.pypa.io/pip/3.5/get-pip.py -O /tmp/get-pip.py
elif echo $(python3 -V 2>&1) | grep -e "Python 3.4"; then
    sudo wget https://bootstrap.pypa.io/pip/3.4/get-pip.py -O /tmp/get-pip.py
else
    sudo apt-get -y install python3-distutils
    sudo wget https://bootstrap.pypa.io/get-pip.py -O /tmp/get-pip.py
fi
```

3. Führen Sie die folgenden Befehle aus, um botocore zu installieren.

```
sudo python3 /tmp/get-pip.py
sudo pip3 install botocore
```

Oder

```
sudo /usr/local/bin/pip3 install botocore
```

Wenn Sie botocore auf Debian10 oder Ubuntu20 installieren, verwenden Sie die folgenden Befehle, um die Installation von botocore im angegebenen Zielordner durchzuführen.

- Für Debian10:

```
sudo python3 /tmp/get-pip.py
sudo pip3 install --target /usr/lib/python3/dist-packages botocore
```

- Für Ubuntu20:

```
sudo /usr/local/bin/pip3 install --target /usr/lib/python3/dist-packages boto3
```

So installieren Sie **boto3** auf einer Mac-Instance

- Führen Sie den folgenden Befehl aus, um boto3 auf Ihrer Mac-Instance zu installieren.

```
sudo pip3 install boto3
```

Upgraden von **boto3**

Um auf die neueste kompatible Version von boto3 zu aktualisieren, verwenden Sie die Option `--upgrade`. Beispielsweise:

```
sudo pip3 install boto3 --upgrade
```

Upgraden von **stunnel**

Die Verschlüsselung von Daten während der Übertragung mit der Amazon EFS-Mountinghilfe erfordert OpenSSL Version 1.0.2 oder neuer und eine Version von `stunnel`, die sowohl das Online Certificate Status Protocol (OCSP) als auch die Überprüfung von Zertifikathostnamen unterstützt. Die Amazon EFS-Mountinghilfe verwendet das `stunnel`-Programm für die 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 Amazon EFS-Dateisystems mit TLS fehl.

Nach der Installation der Amazon EFS-Mountinghilfe können Sie die Version von `stunnel` auf Ihrem System 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](#))

- Rufen Sie in einem Webbrowser die `stunnel` Download-Seite <https://stunnel.org/downloads.html> auf.
- 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. Ersetzen Sie durch *latest-stunnel-version* 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 Compute-Instance mit SLES12 zu aktualisieren.

```
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.

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:

- Öffnen Sie mit einem Texteditor Ihrer Wahl die Datei `/etc/amazon/efs/efs-utils.conf`.
- Legen Sie für den `stunnel_check_cert_hostname`-Wert "false" fest.
- 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 widerrufen 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 EFS den Zertifikatsstatus kontinuierlich und erstellt neue Zertifikate, um widerrufen Zertifikate zu ersetzen.

Für höchste Sicherheit können Sie OCSP so aktivieren, dass Ihre Linux-Clients eine Prüfung auf widerrufen Zertifikate ausführen können. OCSP schützt vor der böartigen Verwendung widerrufen Zertifikate. Es ist jedoch unwahrscheinlich, dass dies innerhalb Ihrer VPC auftritt. Für den Fall, dass ein EFS-TLS-Zertifikat widerrufen wird, veröffentlicht Amazon ein Security Bulletin und es wird eine neue Version der EFS-Mountinghilfe freigegeben, die das widerrufen Zertifikat ablehnt.

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
```

Mounting von EFS-Dateisystemen

Im folgenden Abschnitt erfahren Sie, wie Sie ein Amazon EFS-Dateisystem auf einer Linux-Instance mithilfe der Amazon EFS-Mountinghilfe mounten. Dazu erfahren Sie, wie Sie das Dateisystem mit der Datei `fstab` nach Systemneustarts automatisch erneut mounten. Mit der EFS-Mountinghilfe haben Sie die folgenden Optionen zum Mounting des Amazon EFS-Dateisystems:

- Mounting auf unterstützten EC2-Instances
- Mounting mit IAM-Autorisierung
- Mounting mit Amazon EFS-Zugangspunkten
- Mounting mit einem On-Premises-Linux-Client
- Automatisches Mounting von EFS-Dateisystemen beim Neustart einer EC2-Instance
- Mounting eines Dateisystems beim Erstellen einer neuen EC2-Instance

Note

Amazon EFS unterstützt das Mounting von Amazon EC2-Windows-Instances nicht.

Die EFS-Mountinghilfe ist im Paket "amazon-efs-utils" enthalten. Das Paket "amazon-efs-utils" ist eine Open-Source-Sammlung von Amazon EFS-Tools. Weitere Informationen finden Sie unter [Manuelles Installieren des Amazon EFS-Clients](#).

Bevor die Amazon EFS-Mountinghilfe verfügbar war, empfohlen wir das Mounting des Amazon EFS-Dateisystems mithilfe des standardmäßigen Linux-NFS-Clients. Weitere Informationen finden Sie unter [Mounten von Dateisystemen ohne die EFS-Mountinghilfe](#).

Themen

- [Verwenden der EFS-Mountinghilfe zum Mounten von EFS-Dateisystemen](#)
- [Zusätzliche Überlegungen zum Mounting](#)
- [Fehlerbehebung für AMI- und Kernel-Versionen](#)

Verwenden der EFS-Mountinghilfe zum Mounten von EFS-Dateisystemen

Die EFS-Mountinghilfe hilft Ihnen beim Mounting der EFS-Dateisysteme auf den EC2-Linux- und Mac-Instances, auf denen die unter [Übersicht](#) aufgeführten unterstützten Distributionen ausgeführt werden.

Mit der Amazon EFS-Mountinghilfe lassen sich Dateisysteme einfacher mounten. Sie enthält standardmäßig die empfohlenen Amazon EFS-Mountingoptionen. Darüber hinaus enthält die Mountinghilfe integrierte Protokolle für Fehlerbehebungszwecke. Wenn Sie auf ein Problem mit Ihrem Amazon-EFS-Dateisystem stoßen, können Sie diese Protokolle mit dem - AWS Support teilen. Weitere Informationen über das Mounting des Dateisystems finden Sie unter [Mounting von EFS-Dateisystemen](#).

Note

Amazon EFS unterstützt das Mounting von Amazon EC2-Windows-Instances nicht.

Themen

- [Funktionsweise](#)
- [Abrufen von Support-Protokollen](#)
- [Voraussetzungen für die Verwendung der EFS-Mountinghilfe](#)
- [Mounting auf Amazon EC2-Linux-Instances mithilfe der EFS-Mountinghilfe](#)
- [Mounting auf Amazon EC2-Mac-Instances mithilfe der EFS-Mountinghilfe](#)
- [Mounten von Amazon-EFS-Dateisystemen aus einem anderen AWS-Region](#)
- [Mounting von One-Zone-Dateisystemen](#)
- [Mounting mit IAM-Autorisierung](#)
- [Mounting mit EFS-Zugangspunkten](#)
- [Mounting mit On-Premises-Linux-Clients mithilfe der EFS-Mountinghilfe AWS Direct Connect und VPN](#)
- [Automatisches Mounting des Amazon EFS-Dateisystems](#)
- [Mounten von EFS auf mehreren EC2-Instances mit AWS Systems Manager](#)

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

Funktionsweise

Die Mountinghilfe definiert einen Netzwerk-Dateisystemtyp namens `efs`. Dieser ist vollständig kompatibel mit dem Standardbefehl `mount` unter Linux. Außerdem unterstützt die Mountinghilfe das automatische Mounting von Amazon EFS-Dateisystemen während des Bootens einer Instance über Einträge 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 Mounting-Ziels zurückgegriffen. Weitere Informationen finden Sie unter [Mounten von EFS-Dateisystemen von einem anderen AWS-Konto oder einer anderen VPC](#).
- Dateisystem-ID – Wenn Sie die Dateisystem-ID verwenden, löst die Mountinghilfe sie in die lokale IP-Adresse der Elastic-Network-Schnittstelle (ENI) des Mounting-Ziels auf, ohne externe Ressourcen aufzurufen.
- IP-Adresse des Mounting-Ziels – Sie können die IP-Adresse eines der Mounting-Ziele 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 für das Amazon EFS-Dateisystem die Verschlüsselung von Daten während der Übertragung als Mountingoption festgelegt wurde, initialisiert die Mountinghilfe einen `Client-stunnel`-Prozess

sowie einen Kontrollprozess namens `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](#).

Vom Amazon EFS-Client verwendete Mountingoptionen

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

Die Mountinghilfe verfügt über integrierte Protokolle für das Amazon EFS-Dateisystem. Sie können diese Protokolle zur Fehlerbehebung für den - AWS Support freigeben. 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 die einzelnen Stunnel-Prozesse ausgeführt werden, und beendet den Stunnel-Prozess, wenn das Amazon EFS-Dateisystem ausgebonden 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 Protokolländerungen wirksam werden, müssen Sie das Mounting und erneute Mounting des Dateisystems mithilfe der EFS-Mountinghilfe aufheben. 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

Sie können ein Amazon EFS-Dateisystem auf einer Amazon EC2-Instance mithilfe der Amazon EFS-Mountinghilfe 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 Amazon EFS-Mounting-Ziel – Sie erstellen Mountingziele in der Virtual Private Cloud (VPC). Wenn Sie Ihr Dateisystem in der Konsole mit den vom Service empfohlenen Einstellungen erstellen, wird in jeder Availability Zone in der ein Mountingziel erstellt AWS-Region , in der sich das Dateisystem befindet. Anweisungen zur Erstellung von Mountingzielen finden Sie unter [Erstellen und Verwalten von Mount-Zielen und Sicherheitsgruppen](#).

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 die DNS-Datensätze vollständig in der weitergegeben werden AWS-Region , in der sich das Dateisystem befindet.

Wenn Sie ein Mounting-Ziel 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 Amazon EC2-Instance, auf der eine unterstützte Linux- oder macOS-Distribution ausgeführt wird – Es folgen die unterstützten Linux-Distributionen zum Mounting des Dateisystems mit der Mountinghilfe:
 - Amazon Linux 2
 - 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.

- Die Amazon EFS-Mountinghilfe ist auf der EC2-Instance installiert – Die Mountinghilfe ist ein Tool im Dienstprogrammpaket `amazon-efs-utils`. Weitere Informationen zum Installieren von `amazon-efs-utils` finden Sie unter [So installieren Sie amazon-efs-utils mit AWS Systems Manager](#) sowie unter [Manuelles Installieren von amazon-efs-utils](#).
- 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. Sie muss auch so konfiguriert sein, dass sie den von bereitgestellten DNS-Server verwendet AWS. Informationen zum Amazon DNS-Server finden Sie unter [DHCP-Optionsgruppen](#) 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 [Anzeige von DNS-Hostnamen für die EC2-Instance](#) im Amazon-VPC-Benutzerhandbuch.
- Für EC2-Instances und Dateisysteme in verschiedenen AWS-Regionen – Wenn sich die EC2-Instance und das Dateisystem, das Sie mounten, in verschiedenen befinden AWS-Regionen, müssen Sie die `-region`Eigenschaft in der `-efs-utils.conf` Datei bearbeiten. Weitere Informationen finden Sie unter [Mounten von Amazon-EFS-Dateisystemen aus einem anderen AWS-Region](#).

Mounting auf Amazon EC2-Linux-Instances mithilfe der EFS-Mountinghilfe

Dieser Vorgang erfordert die folgenden Voraussetzungen:

- Sie müssen das `amazon-efs-utils`-Paket für die EC2-Instance installieren. 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 [Erstellen und Verwalten von Mount-Zielen und Sicherheitsgruppen](#).

So mounten Sie ein Amazon EFS-Dateisystem mithilfe der EFS-Mountinghilfe auf einer EC2-Linux-Instance

1. Öffnen Sie über Secure Shell (SSH) ein Terminal für die EC2-Instance und melden Sie sich mit dem entsprechenden Benutzernamen an. Weitere Informationen zu Linux-Instances finden Sie unter [Herstellen einer Verbindung mit Ihrer Linux-Instance mithilfe von SSH](#).
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 Amazon-EFS-Dateisystemen aus 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, das Sie an Ort *file-system-id* und `efs` Stelle von 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.

Mounting auf Amazon EC2-Mac-Instances mithilfe der EFS-Mountinghilfe

Dieser Vorgang erfordert die folgenden Voraussetzungen:

- Sie müssen das `amazon-efs-utils`-Paket für die EC2-Mac-Instance installieren. 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 [Erstellen und Verwalten von Mount-Zielen und Sicherheitsgruppen](#).

- 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 Amazon EFS-Dateisysteme mit der EFS-Mountinghilfe auf EC2 Mac-Instances, auf denen macOS Big Sur oder Monterey 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 [Herstellen einer Verbindung mit Ihrer Instance mit SSH](#) für Mac-Instances im Amazon-EC2-Benutzerhandbuch für Linux-Instances.
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 Amazon-EFS-Dateisystemen aus einem anderen AWS-Region

Wenn Sie Ihr EFS-Dateisystem von einer Amazon EC2-Instance mounten, die sich in einer anderen AWS-Region als das Dateisystem befindet, müssen Sie den `region` Eigenschaftswert in der `efs-utils.conf` Datei bearbeiten.

So bearbeiten Sie die Eigenschaft "region" 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 hierzu finden Sie unter [Herstellen einer Verbindung mit Ihrer Linux-Instance mit SSH](#) im Amazon-EC2-Benutzerhandbuch für Linux-Instances.

2. Suchen Sie die Datei `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 Region `us-east-1` befindet, ersetzen Sie `us-east-1` durch die ID der entsprechenden Region.
 - c. Speichern Sie die Änderungen.
4. Fügen Sie einen Hosteintrag für das regionsübergreifende Mounting hinzu. Weitere Information dazu finden Sie unter [Schritt 3: Hinzufügen eines Hosteintrags für das Mounting-Ziel](#).
 5. Mounting Sie das Dateisystem mit der EFS-Mountinghilfe für [Linux](#)- oder [Mac](#)-Instances.

Mounting von One-Zone-Dateisystemen

Amazon EFS One-Zone-Dateisysteme unterstützen nur ein einziges Mountingziel, 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 [Manuelles Installieren des Amazon EFS-Clients](#).
- Das Dateisystem, das Sie gerade erstellt haben, verfügt über ein Mountingziel. Weitere Informationen finden Sie unter [Erstellen und Verwalten von Mount-Zielen und Sicherheitsgruppen](#).

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

Wenn Sie ein One-Zone-Dateisystem auf einer EC2-Instance mounten, die sich in einer anderen Availability Zone befindet, müssen Sie im Befehl für die Mountinghilfe den Availability-Zone-Namen des Dateisystems oder den DNS-Namen des Mountingziels des Dateisystems 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 verwenden `/etc/fstab`, um ein EFS-Dateisystem mit One Zone-Speicher auf einer EC2-Instance zu 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.

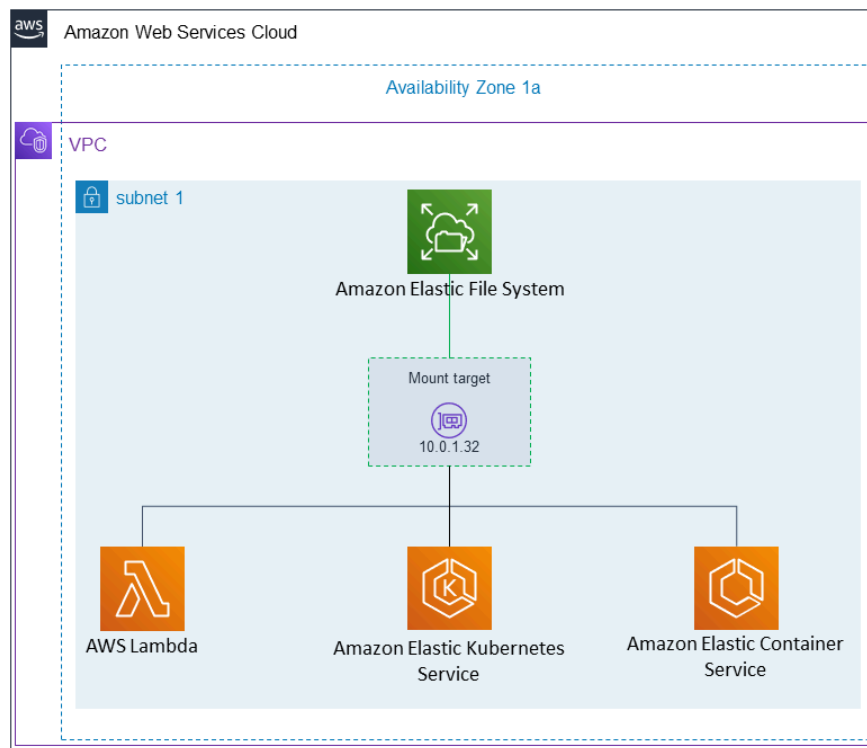
```
availability-zone-name.file-system-id.efs.aws-region.amazonaws.com:/ efs-mount-point  
nfs4  
nfsvers=4.1,rsize=1048576,wsiz=1048576,hard,timeo=600,retrans=2,noresvport,_netdev 0  
0
```

```
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 [Verwenden von NFS zum automatischen Mounting von EFS-Dateisystemen](#).

Mounting von Dateisystemen mit One Zone-Dateisystem auf anderen AWS Datenverarbeitungs-Instances

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



Herstellen einer von Amazon Elastic Container Service ausgehenden Verbindung

Sie können Amazon EFS-Dateisysteme mit Amazon ECS verwenden, um Daten des Dateisystems in Ihrer gesamten Flotte der Container-Instances gemeinsam zu nutzen, sodass die Aufgaben unabhängig von der Instance, auf der sie landen, Zugriff auf denselben persistenten Speicher haben. Für die Verwendung von Amazon EFS One-Zone-Dateisystemen mit Amazon ECS sollten Sie beim Starten der Aufgabe nur Subnetze auswählen, die sich in derselben Availability Zone wie das 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 mit dem Amazon EFS [Container Storage Interface](#) (CSI)-Treiber, der Amazon EFS-Zugangspunkte unterstützt, ein Dateisystem für mehrere Pods in einem Amazon EKS- oder selbstverwalteten Kubernetes-Cluster gemeinsam nutzen. Der Amazon EFS CSI-Treiber ist im Fargate-Stack installiert. Wenn Sie den Amazon EFS CSI-Treiber mit Amazon EFS One-Zone-Dateisystemen verwenden, können Sie die Option `nodeSelector` beim Starten des Pods verwenden, um sicherzustellen, dass er in derselben Availability Zone wie das Dateisystem geplant wird.

Herstellen einer Verbindung von AWS Lambda

Sie können Amazon EFS mit verwenden, AWS Lambda um Daten über Funktionsaufrufe hinweg freizugeben, große Referenzdatendateien zu lesen und Funktionsausgaben in einen persistenten und freigegebenen Speicher zu schreiben. Lambda verbindet die Funktionsinstanzen sicher mit den Amazon EFS-Mountingzielen, 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 die EFS-Mountinghilfe, um Ihr Amazon 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 Dateisystemdatenzugriffs](#).

In den folgenden Abschnitten müssen Sie ein Verzeichnis erstellen, das als Mountingpunkt 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 zu einer Amazon 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 mit IAM-Autorisierung unter Verwendung der IAM-Anmeldeinformationen mounten ~/.aws/credentials, die sich in der Datei mit den AWS CLI Anmeldeinformationen oder in 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
```

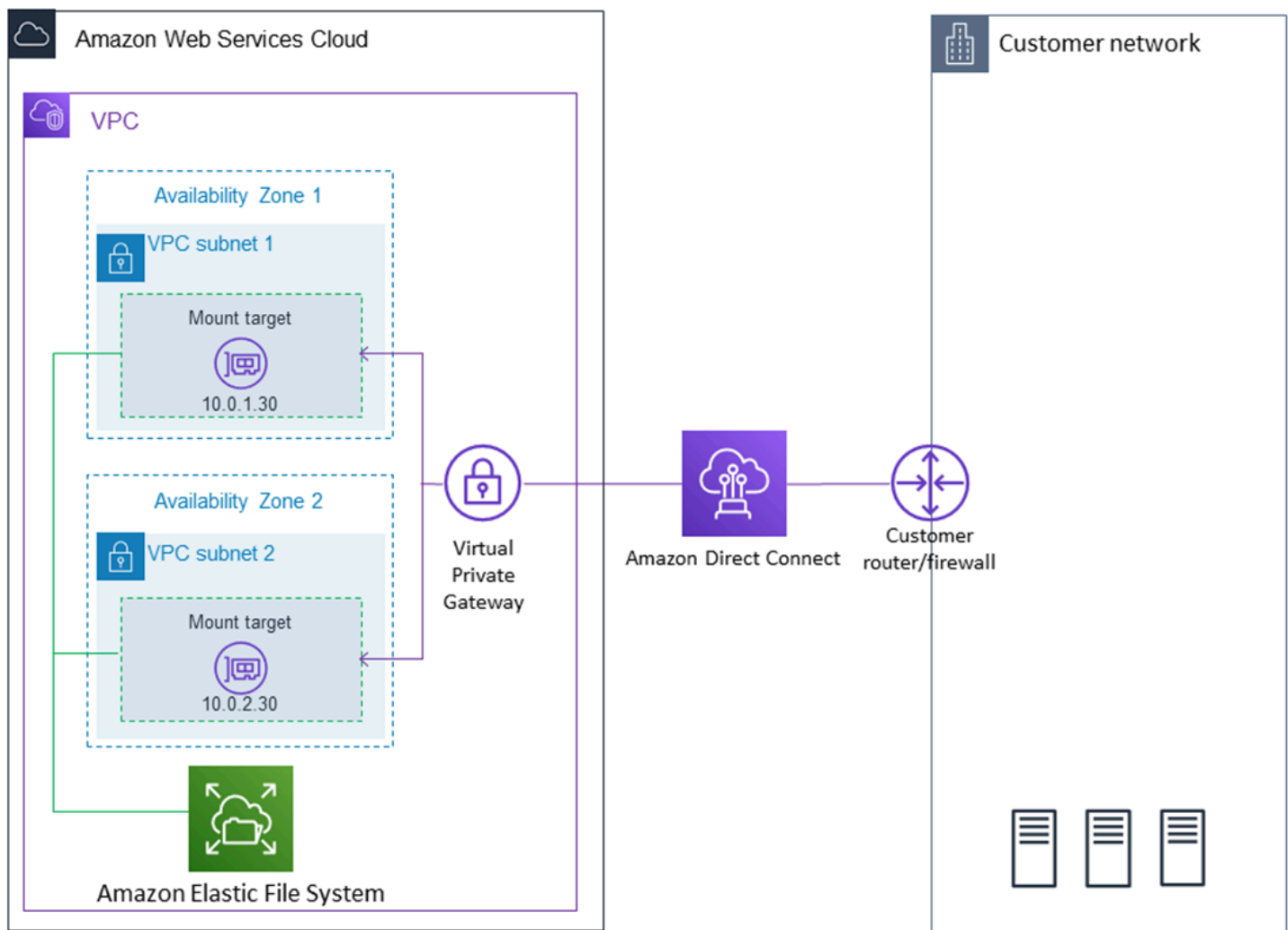
Wenn Sie ein Dateisystem mithilfe eines Zugangspunkts automatisch mounten möchten, fügen Sie der `/etc/fstab`-Datei auf der EC2-Instance die folgende Zeile hinzu.

```
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 Amazon EFS Access Points](#).

Mounting mit On-Premises-Linux-Clients mithilfe der EFS-Mountinghilfe AWS Direct Connect und VPN

Sie können Ihre Amazon-EFS-Dateisysteme auf Ihren On-Premises-Rechenzentrum-Servern mounten, wenn Sie mit Ihrer Amazon VPC über AWS Direct Connect oder VPN verbunden sind. Die folgende Grafik zeigt ein allgemeines Schemadiagramm der , die zum Mounten von Amazon-EFS-Dateisystemen von On-Premises AWS-Services erforderlich sind.



Weitere Informationen zur Verwendung von `amazon-efs-utils` mit AWS Direct Connect und VPN zum Mounten von Amazon-EFS-Dateisystemen auf On-Premises-Linux-Clients finden Sie unter

[Exemplarische Vorgehensweise: Erstellen und Bereitstellen eines lokalen Dateisystems mit VPN AWS Direct Connect.](#)

Automatisches Mounting des Amazon EFS-Dateisystems

Sie können eine Amazon EC2-Instance so konfigurieren, dass ein EFS-Dateisystem beim Neustart mit der EFS-Mountinghilfe oder mit NFS automatisch bereitgestellt wird.

- Verwenden der EFS-Mountinghilfe:
 - Hängen Sie bei der Erstellung einer neuen EC2-Linux-Instance mit dem Assistenten zum Starten von EC2-Instances ein EFS-Dateisystem an.
 - Aktualisieren Sie die `/etc/fstab`-EC2-Datei mit einem Eintrag für das EFS-Dateisystem.
- Verwenden Sie [NFS ohne die EFS-Mountinghilfe](#) zur Aktualisierung der EC2-Datei `/etc/fstab` zur Unterstützung von EC2-Linux- und -Mac-Instances.

Note

Die EFS-Mountinghilfe unterstützt kein automatisches Mounten auf Amazon 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

- [Verwenden der EFS-Mountinghilfe, um EFS-Dateisysteme automatisch erneut zu mounten](#)
- [Verwenden von NFS zum automatischen Mounting von EFS-Dateisystemen](#)

Verwenden der EFS-Mountinghilfe, um EFS-Dateisysteme automatisch erneut zu mounten

Verwenden Sie die EFS-Mountinghilfe, um die `/etc/fstab`-Datei auf EC2-Linux-Instances so zu konfigurieren, dass die EFS-Dateisysteme beim Neustart der Instance automatisch erneut gemountet werden.

Themen

- [Anhängen eines EFS-Dateisystems beim Erstellen einer EC2-Instance, um das automatische Mounting beim Neustart zu ermöglichen](#)
- [Verwenden von /etc/fstab mit der EFS-Mountinghilfe, um EFS-Dateisysteme automatisch erneut zu mounten](#)

Anhängen eines EFS-Dateisystems beim Erstellen einer EC2-Instance, um das automatische Mounting beim Neustart zu ermöglichen

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.

Wenn Sie eine neue Amazon EC2 Linux-Instance mit dem Assistenten für EC2-Start-Instances erstellen, können Sie diese so konfigurieren, dass das Amazon EFS-Dateisystem automatisch gemountet wird. Die EC2-Instance mountet das Dateisystem automatisch bei der zuerst gestarteten Instance und auch bei jedem Neustart.

 Note

Amazon EFS-Dateisysteme unterstützen kein Mounting auf Amazon EC2 Mac-Instances, auf denen beim Start der Instance macOS Big Sur oder Monterey ausgeführt wird.

Sie müssen zuvor das Amazon EFS-Dateisystem erstellen. Weitere Informationen finden Sie unter [Schritt 1: Erstellen eines Amazon-EFS-Dateisystems](#) in der Übung "Amazon EFS – Erste Schritte".

 Note

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

Bevor Sie eine Amazon EC2-Instance starten und sich mit ihr verbinden können, müssen Sie ein Schlüsselpaar erstellen, es sei denn, es ist bereits eines vorhanden. Folgen Sie den Schritten unter [Einrichtung mit Amazon EC2](#) im Amazon EC2-Benutzerhandbuch für Linux-Instances, um ein Schlüsselpaar zu erstellen. Wenn Sie bereits über ein Schlüsselpaar verfügen, können Sie es für diese Übung verwenden.

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.

Verwenden von **/etc/fstab** mit der EFS-Mountinghilfe, um EFS-Dateisysteme automatisch erneut zu mounten

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**-Datei auf einer EC2-Linux-Instance manuell, sodass die Instance die EFS-Mountinghilfe verwendet, um ein EFS-Dateisystem beim Neustart der Instance automatisch erneut zu mounten.

Note

Amazon EFS-Dateisysteme unterstützen kein automatisches Mounting mithilfe von **/etc/fstab** mit der EFS-Mountinghilfe auf Amazon 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 stehen für die Installation auf Amazon Linux- und Amazon Linux 2-Amazon-Systemabbildern Machine Images (AMIs) zur Verfügung. Mehr über `amazon-efs-utils` erfahren Sie unter [Verwenden der amazon-efs-utils Tools](#). Wenn Sie eine andere Linux-Verteilung

wie Red Hat Enterprise Linux (RHEL) verwenden, erstellen und installieren Sie die `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 Amazon EFS-Dateisystem erstellt, das automatisch erneut gemountet werden soll. Weitere Informationen finden Sie unter [Schritt 1: Erstellen eines Amazon-EFS-Dateisystems](#).
- 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 [Verwenden der amazon-efs-utils Tools](#).

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

1. Stellen Sie eine Verbindung mit der EC2-Instance her:

- Wenn Sie von einem Computer unter macOS oder Linux eine Verbindung mit Ihrer Instance herzustellen möchten, geben Sie die PEM-Datei für den SSH-Befehl an. Verwenden Sie dazu die `-i`-Option und den Pfad zu Ihrem privaten Schlüssel.
- Um von einem Computer, auf dem Windows ausgeführt wird, eine Verbindung zu Ihrer Instance herzustellen, können Sie entweder MindTerm oder PuTTY verwenden. Zur Verwendung von PuTTY installieren Sie es und konvertieren Sie die PEM-Datei in eine PPK-Datei.

Weitere Informationen finden Sie in den folgenden Themen im Amazon-EC2-Benutzerhandbuch für Linux-Instances:

- [Herstellen einer Verbindung zu Ihrer Linux-Instance von Windows über PuTTY](#)
- [Herstellen einer Verbindung zu Ihrer Linux-Instance über SSH](#)

2. Öffnen Sie die Datei `/etc/fstab` in einem Editor.

3. Automatisches Mounting des EFS-Dateisystems mithilfe der IAM-Autorisierung oder eines EFS-Zugangspunkts:

- Für ein automatisches Mounting eines Dateisystems mit IAM-Autorisierung zu einer Amazon EC2-Instance mit einem Instance-Profil fügen Sie der `/etc/fstab`-Datei die folgende Zeile hinzu.

```
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,iam,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 finden Sie unter [Mounting mit IAM-Autorisierung](#) und [Mounting mit EFS-Zugangspunkten](#).

4. Speichern Sie die Änderungen an der Datei.
5. Testen Sie den `fstab`-Eintrag, indem Sie den `mount`-Befehl mit der `'fake'`-Option zusammen mit den Optionen `„all“` und `„verbose“` verwenden.

```
$ sudo mount -fav  
home/ec2-user/efs      : successfully mounted
```

Ihre EC2-Instance ist jetzt so konfiguriert, dass das EFS-Dateisystem gemountet wird, wenn sie neu gestartet wird.

 Note

In einigen Fällen muss die Amazon EC2-Instance möglicherweise unabhängig vom Status des gemounteten Amazon 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	Beschreibung
<i>file-system-id</i> :/	Die ID des Amazon EFS-Dateisystems. Sie können diese ID über die Konsole oder programmgesteuert über die CLI oder ein AWS SDK abrufen.
<i>efs-mount-point</i>	Der Mountingpunkt für das EFS-Dateisystem auf Ihrer EC2-Instance.
<code>efs</code>	Der Typ des Dateisystems. Wenn Sie die Mountinghilfe verwenden, ist dieser Typ immer <code>efs</code> .
<code>mount options</code>	<p>Mountingoptionen für das Dateisystem. Dies ist eine durch Kommata getrennte Liste der folgenden Optionen:</p> <ul style="list-style-type: none">• <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.

Feld	Beschreibung
	<ul style="list-style-type: none"> • <code>iam</code> – Verwenden Sie diese Option, um mit IAM-Autorisierung zu einer Amazon EC2-Instance zu mounten, die über ein Instance-Profil 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 Dateisystemdatenzugriffs. • <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 Dateisystemdatenzugriffs. • <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 Amazon EFS Access Points.
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 <code>fsck</code> die Dateisysteme beim Systemstart prüft. Bei EFS-Dateisystemen sollte dieser Wert 0 lauten; dieser gibt an, dass <code>fsck</code> beim Start nicht ausgeführt werden soll.

Verwenden von NFS zum automatischen Mounting von EFS-Dateisystemen

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

1. Stellen Sie eine Verbindung mit der EC2-Instance her:

- Wenn Sie von einem Computer unter macOS oder Linux eine Verbindung mit Ihrer Instance herzustellen möchten, geben Sie die PEM-Datei für den SSH-Befehl an. Verwenden Sie dazu die `-i`-Option und den Pfad zu Ihrem privaten Schlüssel.
- Um von einem Computer, auf dem Windows ausgeführt wird, eine Verbindung zu Ihrer Instance herzustellen, können Sie entweder MindTerm oder PuTTY verwenden. Zur

Verwendung von PuTTY installieren Sie es und konvertieren Sie die PEM-Datei in eine PPK-Datei.

Weitere Informationen finden Sie in den folgenden Themen im Amazon-EC2-Benutzerhandbuch für Linux-Instances:

- [Herstellen einer Verbindung zu Ihrer Linux-Instance von Windows über PuTTY](#)
- [Herstellen einer Verbindung zu Ihrer Linux-Instance über SSH](#)

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.
 - Ersetzen Sie *file_system_id* durch die ID des Dateisystems, das Sie mounten.
 - Ersetzen Sie *aws-region* durch die AWS-Region, in der sich das Dateisystem befindet, z. B. `us-east-1`.
 - 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	Beschreibung
<i>file-system-id</i> :/	Die ID des Amazon EFS-Dateisystems. Sie können diese ID über die Konsole oder programmgesteuert über die CLI oder ein AWS SDK abrufen.
<i>efs-mount-point</i>	Der Mountingpunkt für das EFS-Dateisystem auf Ihrer EC2-Instance.
nfs4	Gibt den Dateisystemtyp an.
mount options	Die kommasetrennte Liste der Mountingoptionen für das Dateisystem: <ul style="list-style-type: none"> • <code>nfsvers=4.1</code> – Gibt die Verwendung von NFS v4.1 an.

Feld	Beschreibung
	<ul style="list-style-type: none"> • <code>rsize=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>wsize=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 Soft-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 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 Netzwerkwiederherstellungsereignis sichergestellt.

Feld	Beschreibung
	<ul style="list-style-type: none"> • <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.

Mounten von EFS auf mehreren EC2-Instances mit AWS Systems Manager

Sie können EFS-Dateisysteme remote und sicher auf mehreren Amazon EC2-Instances mounten, ohne sich mit dem AWS Systems Manager Run Befehl 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 Instances zu installieren. Weitere Informationen finden Sie unter [Schritt 2: Konfigurieren Sie eine Zuordnung, die von State Manager für die Installation oder Aktualisierung des Amazon EFS-Clients verwendet wird.](#)

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 aus.

6. Geben Sie unter Befehlsparameter den Mountingbefehl für jedes EFS-Dateisystem ein, das Sie mounten möchten. Beispielsweise:

```
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 [Mounting auf Amazon EC2-Linux-Instances mithilfe der EFS-Mountinghilfe](#) oder [Mounting auf Amazon EC2-Mac-Instances mithilfe der EFS-Mountinghilfe](#).

7. Wählen Sie die AWS Systems Manager verwalteten EC2-Ziel-Instances 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 einem anderen AWS-Konto oder einer anderen VPC

Sie können das Amazon EFS-Dateisystem mithilfe der IAM-Autorisierung für NFS-Clients und EFS-Zugangspunkte mithilfe der EFS-Mountinghilfe mounten. Standardmäßig verwendet die EFS-Mountinghilfe DNS (Domain Name Service), um die IP-Adresse Ihres EFS-Mounting-Ziels aufzulösen. Wenn Sie das Dateisystem von einem anderen Konto oder einer anderen Virtual Private Cloud (VPC) mounten, müssen Sie das EFS-Mounting-Ziel manuell auflösen.

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

Mounting mithilfe von IAM oder Zugangspunkten von einer anderen VPC

Wenn Sie eine VPC-Peering-Verbindung oder ein Transit-Gateway zur Verbindung von VPC verwenden, können Amazon EC2-Instances in einer VPC auf EFS-Dateisysteme in einer anderen VPC zugreifen, selbst wenn die VPCs zu verschiedenen Konten gehören.

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 [Verwenden der amazon-efs-utils Tools](#).
- Lassen Sie die Aktion `ec2:DescribeAvailabilityZones` in der IAM-Richtlinie für die IAM-Rolle zu, die Sie der Instance zugewiesen haben. Wir empfehlen Ihnen, die AWS verwaltete Richtlinie an eine IAM-Entität `AmazonElasticFileSystemsUtils` anzuhängen, um die erforderlichen Berechtigungen für die Entität bereitzustellen.
- Aktualisieren Sie beim Mounten von einem anderen aus die Dateisystemressourcenrichtlinie AWS-Konto, um die `elasticfilesystem:DescribeMountTarget` Aktion für den Prinzipal-ARN anderer zuzulassen AWS-Konto. Beispielsweise:

```
{
  "Id": "access-point-example03",
  "Statement": [
    {
      "Sid": "access-point-statement-example03",
      "Effect": "Allow",
      "Principal": {"AWS": "arn:aws:iam::555555555555"},
      "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 innerhalb von 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 zur Verbindung von VPC verwenden, können Amazon EC2-Instances in

einer VPC auf EFS-Dateisysteme in einer anderen VPC zugreifen, selbst wenn die VPCs zu verschiedenen Konten gehören.

Ein Transit-Gateway ist ein Netzwerk-Transit-Hub, mit dem Sie Ihre VPCs und On-Premises-Netzwerke miteinander verbinden können. 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. Mit diesem Verbindungstyp können Sie Datenverkehr dazwischen über private IPv4 (Internet Protocol Version 4) oder IPv6-Adressen (Internet Protocol Version 6) weiterleiten. Sie können VPC-Peering verwenden, um VPCs innerhalb derselben AWS-Region oder zwischen AWS-Region zu verbinden. 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-Mounting-Ziel 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 mit der EC2-Instance her:

- Wenn Sie von einem Computer unter macOS oder Linux eine Verbindung mit Ihrer Instance herzustellen möchten, geben Sie die PEM-Datei für den SSH-Befehl an. Verwenden Sie dazu die `-i`-Option und den Pfad zu Ihrem privaten Schlüssel.
- Um von einem Computer, auf dem Windows ausgeführt wird, eine Verbindung zu Ihrer Instance herzustellen, können Sie entweder MindTerm oder PuTTY verwenden. Zur Verwendung von PuTTY installieren Sie es und konvertieren Sie die PEM-Datei in eine PPK-Datei.

Weitere Informationen finden Sie in den folgenden Themen im Amazon-EC2-Benutzerhandbuch für Linux-Instances:

- [Herstellung einer Verbindung zu Ihrer Linux-Instance von Windows mit PuTTY](#)
 - [Herstellen einer Verbindung mit Ihrer Linux-Instance per SSH](#)
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 Dateisystemdatenzugriffs](#).

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 Amazon EFS Access Points](#).

Mounting von Amazon EFS-Dateisystemen aus einer anderen AWS-Region

Wenn Sie Ihr EFS-Dateisystem von einer anderen VPC aus mounten, die sich in einer anderen AWS-Region als das 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.

Mounting von einem anderen AWS-Konto in derselben VPC

Mit gemeinsam genutzten VPCs können Sie ein Amazon-EFS-Dateisystem mounten, das einem AWS-Konto von Amazon EC2-Instances gehört, die einem anderen gehören AWS-Konto. Weitere

Informationen zum Einrichten einer gemeinsam genutzten VPC finden Sie unter [Arbeiten mit freigegebenen VPCs](#) 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.

Zusätzliche Überlegungen zum Mounting

Wir empfehlen die folgenden Werte für die Mountingoptionen unter Linux:

- `rsz=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).
- `wsz=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 Amazon EFS-Mountingziels in derselben Availability Zone wie die Amazon-EC2-Instance auf, ohne externe Ressourcen aufzurufen. 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 Informationen und ausführliche Erklärungen der Standardwerte finden Sie auf den Seiten [man fstab](#) und [man nfs](#) 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.

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 das Mounting eines Amazon-EFS-Dateisystems nicht über die AWS Management Console, AWS CLI oder über eines der - AWS SDKs aufheben. Um das Mounting eines Amazon EFS-Dateisystems aufzuheben, das mit einer Amazon EC2-Instance unter Linux verbunden ist, verwenden Sie den Befehl `umount` 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 das Amazon EFS-Dateisystem gemountet wurde, indem Sie den Befehl `df` ausführen. Mit diesem Befehl werden die Datenträgnutzungstatistiken für die Dateisysteme angezeigt, die derzeit auf der Linux-basierten Amazon EC2-Instance gemountet werden. Wenn das Amazon EFS-Dateisystem, dessen Mounting Sie aufheben möchten, in der Ausgabe des Befehls `df` nicht aufgeführt wird, bedeutet dies, dass das Mounting des Dateisystems aufgehoben wurde.

Example – Identifizieren des Mountingstatus eines Amazon EFS-Dateisystems und Aufheben des Mountings

```
$ 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% /
```

Fehlerbehebung für AMI- und Kernel-Versionen

Informationen zum Beheben von Problemen im Zusammenhang mit bestimmten Amazon Machine Image (AMI)- oder Kernel-Versionen bei der Verwendung von Amazon EFS auf einer Amazon EC2-Instance finden Sie unter [Beheben von AMI- und Kernel-Problemen](#).

 **Note**

Amazon EFS unterstützt das Mounting von Amazon EC2-Windows-Instances nicht.

Übertragung von Daten in und aus Amazon EFS

Sie können Daten in AWS Transfer Family und AWS DataSync aus Ihren Amazon EFS-Dateisystemen verwenden und übertragen. AWS DataSync ist ein Online-Datenübertragungsdienst, 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 [AWS DataSync Zur Übertragung von Daten in Amazon EFS verwenden](#).

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 über das 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 gespeichert sind, und zwar für Anwendungsfälle wie Datenverteilung, Lieferkette, Inhaltsmanagement und Web-Servering-Anwendungen. Weitere Informationen zur Verwendung von Transfer Family mit Amazon EFS finden Sie unter [Für AWS Transfer Family den Zugriff auf Dateien in Ihrem Amazon EFS-Dateisystem verwenden](#).

Themen

- [AWS DataSync Zur Übertragung von Daten in Amazon EFS verwenden](#)
- [Für AWS Transfer Family den Zugriff auf Dateien in Ihrem Amazon EFS-Dateisystem verwenden](#)

AWS DataSync Zur Übertragung von Daten in Amazon EFS verwenden

AWS DataSync ist ein Online-Datenübertragungsdienst, der das Verschieben und Replizieren von Daten zwischen lokalen Speichersystemen sowie zwischen Speicherdiensten vereinfacht, automatisiert und beschleunigt. AWS DataSync kann Daten zwischen Network File System (NFS), Server Message Block (SMB) -Dateiservern, selbstverwaltetem Objektspeicher, Amazon S3-Buckets, AWS Snowcone, Amazon EFS-Dateisystemen und FSx für Windows File Server-Dateisystemen kopieren.

Sie können es auch verwenden DataSync, um Dateien zwischen zwei EFS-Dateisystemen zu übertragen, einschließlich Dateisystemen in verschiedenen AWS-Regionen und Dateisystemen, die verschiedenen AWS-Konten gehören. Bei Verwendung von DataSync zum Kopieren von Daten zwischen EFS-Dateisystemen können Sie einmalige Datenmigrationen und regelmäßige

Dateneingaben für verteilte Workloads durchführen und die Replikation für Datenschutz und Wiederherstellung automatisieren.

Weitere Informationen finden Sie im [Erste Schritte mit Amazon Elastic File System](#) und dem [AWS DataSync-Benutzerhandbuch](#).

Für AWS Transfer Family den Zugriff auf Dateien in Ihrem Amazon EFS-Dateisystem 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](#))
- Sicheres Dateiübertragungsprotokoll (FTPS) ([AWS Transfer for FTPS](#))
- Dateiübertragungsprotokoll (FTP) ([AWS Transfer for FTP](#))

Mit Transfer Family können Sie Dritten wie Ihren Lieferanten, Partnern oder Kunden den sicheren Zugriff auf Ihre Dateien über die unterstützten Protokolle weltweit ermöglichen, ohne eine Infrastruktur verwalten zu müssen. Darüber hinaus können Sie jetzt mithilfe von SFTP-, FTPS- und FTP-Clients problemlos von Windows-, MacOS- und Linux-Umgebungen aus auf Ihre EFS-Dateisysteme zugreifen. Dies trägt dazu bei, den Zugriff auf Ihre Daten über NFS-Clients und Zugriffspunkte hinaus für Benutzer in mehreren Umgebungen zu erweitern.

Die Verwendung von Transfer Family zur Übertragung von Daten in Amazon EFS-Dateisystemen wird genauso berücksichtigt wie die Nutzung anderer Clients. Weitere Informationen finden Sie unter [Durchsatzmodi](#) und [Amazon-EFS-Kontingente und -Limits](#).

Weitere Informationen zu AWS Transfer Family 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 AWS Support.

Themen

- [Voraussetzungen für die Verwendung AWS Transfer Family mit Amazon EFS](#)
- [Konfiguration Ihres Amazon EFS-Dateisystems für die Verwendung mit 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 Family-Server und Ihr Amazon EFS-Dateisystem befinden sich im selben VerzeichnisAWS-Region.
- IAM-Richtlinien sind so konfiguriert, dass sie den Zugriff auf die von Transfer Family verwendete IAM-Rolle ermöglichen. Weitere Informationen finden Sie im Benutzerhandbuch unter [Erstellen einer IAM-Rolle und -Richtlinie](#). AWS Transfer Family
- (Optional) Wenn der Transfer Family 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 [Blockieren des öffentlichen Zugriffs](#).
 - Ändern Sie die Dateisystemrichtlinie, um den kontoübergreifenden Zugriff zu ermöglichen. Weitere Informationen finden Sie unter [Konfiguration des kontoübergreifenden Zugriffs für Transfer Family](#).

Konfiguration Ihres Amazon EFS-Dateisystems für die Verwendung mit 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-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 die POSIX-IDs verwenden, die den Transfer Family Family-Benutzern zugewiesen sind.

- Schritt 3. Konfigurieren Sie IAM, um den Zugriff auf die von Transfer Family verwendete IAM-Rolle zu ermöglichen.

Datei- und Verzeichnisberechtigungen für Transfer Family Family-Benutzer einrichten

Stellen Sie sicher, dass die Transfer Family Family-Benutzer Zugriff auf die erforderlichen Dateien und Verzeichnisse in Ihrem EFS-Dateisystem haben. Weisen Sie dem Verzeichnis mithilfe der Liste der POSIX-IDs, die den Benutzern der Transfer Family zugewiesen sind, Zugriffsberechtigungen zu. In diesem Beispiel erstellt ein Benutzer ein Verzeichnis, das `transferFam` unter dem EFS-Bereitstellungspunkt benannt ist. Das Erstellen eines Verzeichnisses ist je nach Anwendungsfall optional. Bei Bedarf können Sie den Namen und den Speicherort im EFS-Dateisystem auswählen.

Um POSIX-Benutzern Datei- und Verzeichnisberechtigungen für Transfer Family zuzuweisen

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 bereitgestellt ist. Weitere Informationen finden Sie unter [Mounting von EFS-Dateisystemen](#).
3. Im folgenden Beispiel wird das Verzeichnis im EFS-Dateisystem erstellt und seine Gruppe in die POSIX-Gruppen-ID für die Transfer Family Family-Benutzer geändert, die in diesem Beispiel 1101 ist.
 - a. Erstellen Sie das Verzeichnis `efs/transferFam` mit den folgenden Befehlen. 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 von in die POSIX-GID `efs/transferFam` zu ändern, die den Transfer Family Family-Benutzern zugewiesen ist.

```
[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
```

Zugriff auf die von Transfer Family verwendete IAM-Rolle aktivieren

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 im Benutzerhandbuch unter [Erstellen einer IAM-Rolle und -Richtlinie](#). AWS Transfer Family Sie müssen dieser Transfer Family IAM-Rolle Zugriff auf Ihr EFS-Dateisystem gewähren, indem Sie entweder eine IAM-Identitätsrichtlinie oder eine Dateisystemrichtlinie verwenden.

Im Folgenden finden Sie ein Beispiel für eine Dateisystemrichtlinie, die ClientMount (Lese-) ClientWrite Zugriff auf die IAM-Rolle gewährt. EFS-role-for-transfer

```
{
  "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"
      ]
    }
  ]
}
```

Weitere Informationen zum Erstellen einer Dateisystemrichtlinie finden Sie unter [Erstellen von Dateisystemrichtlinien](#). Weitere Informationen zur 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, zu einem anderen AWS-Konto gehört, müssen Sie diesem Konto Zugriff auf Ihr Dateisystem gewähren. Außerdem muss Ihre Dateisystemrichtlinie nicht öffentlich sein. Weitere Informationen zum Blockieren des öffentlichen Zugriffs auf Ihr Dateisystem finden Sie unter [Blockieren des öffentlichen Zugriffs](#).

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** 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 anhand der Konto-ID oder des Kontos Amazon Resource Name (ARN) angeben. Weitere Informationen zu ARNs finden Sie unter [IAM-ARNs](#) im IAM-Benutzerhandbuch.

Das folgende Beispiel ist eine Richtlinie für ein nicht öffentliches Dateisystem, die kontoübergreifenden Zugriff auf das Dateisystem gewährt. Sie enthält die folgenden zwei Anweisungen:

1. Die erste Anweisung, `NFS-client-read-write-via-fsmt`, gewährt NFS-Clients, die über ein Dateisystem-Mount-Ziel auf das Dateisystem zugreifen, Lese-, Schreib- und Root-Rechte.
2. Die zweite Anweisung, `Grant-cross-account-access`, gewährt dem AWS-Konto `11122223333`, 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",
      "Principal": {
        "AWS": "*"
      },
      "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",
      "Principal": {
        "AWS": "*"
      },
      "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"
  ]
}
]
```

Verwalten von Amazon-EFS-Dateisystemen

Die Aufgaben zur Dateisystemverwaltung beziehen sich auf das Erstellen und Löschen von Dateisystemen, die Verwaltung von Tags und die Verwaltung der Netzwerkzugänglichkeit mit Mountingzielen von vorhandenen Dateisystemen.

Sie können diese Dateisystemverwaltungsaufgaben mit der AWS Management Console oder programmgesteuert mit der AWS Command Line Interface (AWS CLI) oder API ausführen, wie in den folgenden Abschnitten beschrieben.

Themen

- [Verwalten der Netzwerkzugänglichkeit des Dateisystems](#)
- [Verwalten des Dateisystemdurchsatzes](#)
- [Verwaltung des Dateisystemspeichers](#)
- [Zugriffsverwaltung auf verschlüsselte Dateisysteme](#)
- [Messung: Wie Amazon EFS die Größe von Dateisystemen und Objekten meldet](#)
- [Verwalten der Amazon EFS-Dateisystemkosten mitAWSBudgets](#)
- [Status des Dateisystems](#)

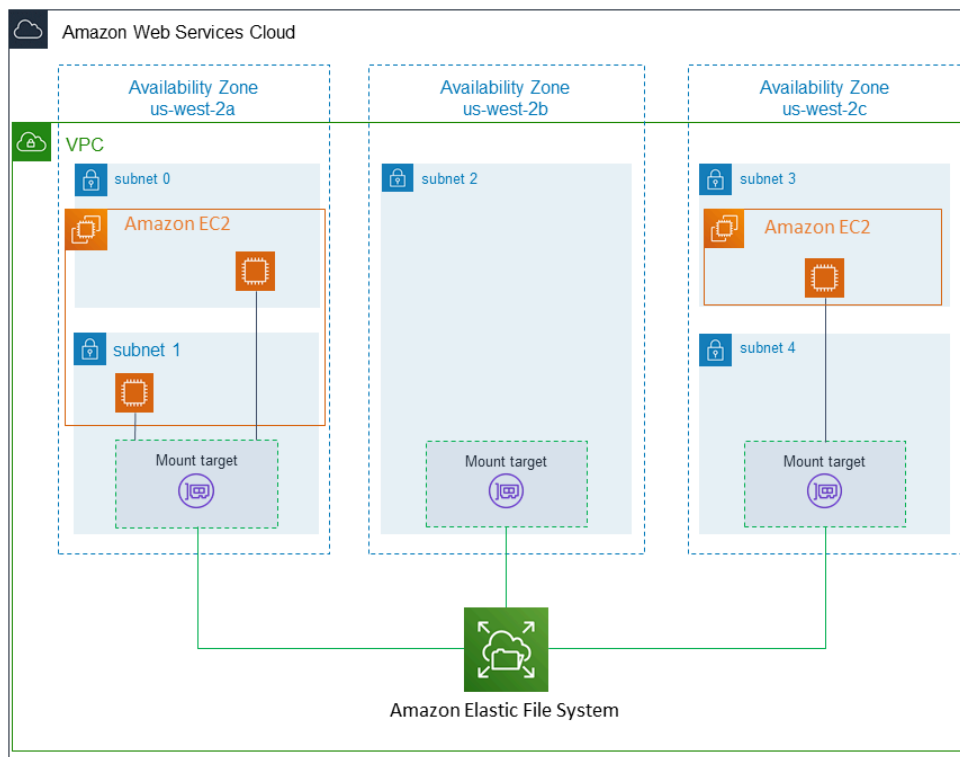
Wenn Sie noch nicht mit Amazon EFS vertraut sind, empfehlen wir Ihnen, die folgenden Übungen auszuprobieren, die Ihnen end-to-end Erfahrung mit der Verwendung eines Amazon-EFS-Dateisystems bieten:

- [Erste Schritte](#) – Diese Übung bietet eine konsolenbasierte end-to-end Einrichtung, in der Sie ein Dateisystem erstellen, es auf einer Amazon EC2 mounten und die Einrichtung testen. Die Konsole kümmert sich um viele Dinge für Sie und hilft Ihnen so, das end-to-end Erlebnis schnell einzurichten.
- [Exemplarische Anleitung: Erstellen eines Amazon EFS-Dateisystems und das Mounten auf einer Amazon EC2 EC2-Instance mithilfe derAWS CLI](#) – Diese Anleitung ähnelt der Übung „Erste Schritte“, verwendet jedoch die , AWS CLI um die meisten Aufgaben auszuführen. Da die CLI-Befehle weitgehend der Amazon-EFS-API zugeordnet werden können, kann diese Anleitung hilfreich sein, um sich mit der Amazon-EFS-API vertraut zu machen.

Verwalten der Netzwerkzugänglichkeit des Dateisystems

Sie mounten Ihr Dateisystem auf Amazon EC2 oder einer anderen AWS Computing-Instance in Ihrer Virtual Private Cloud (VPC) mithilfe eines Mountingziels, das Sie für das Dateisystem erstellen. Die Verwaltung der Netzwerkzugänglichkeit eines Dateisystems bezieht sich auf die Verwaltung der Mountingziele.

Die folgende Abbildung zeigt, wie EC2-Instances in einer VPC unter Verwendung eines Mountingziels auf ein Amazon-EFS-Dateisystem zugreifen.



In der Abbildung sind drei EC2-Instances zu sehen, die in verschiedenen VPC-Subnetzen gestartet wurden und die auf ein Amazon-EFS-Dateisystem zugreifen. Darüber hinaus ist in der Abbildung in jeder Availability Zone (unabhängig von der Anzahl der Subnetze in jeder Availability Zone) ein Mounting-Ziel zu sehen.

Sie können nur ein Mounting-Ziel 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 Mounting-Ziel. Solange es in einer Availability Zone ein Mounting-Ziel gibt, können die EC2-Instances, die in einem der Subnetze der Zone gestartet wurden, dasselbe Mounting-Ziel gemeinsam verwenden.

Die Verwaltung von Mounting-Zielen bezieht sich auf folgende Aktivitäten:

- Erstellen und Löschen von Mountingzielen in einer VPC – Sie sollten mindestens in jeder Availability Zone, von der aus Sie auf das Dateisystem zugreifen möchten, ein Mountingziel erstellen.

 Note

Wir empfehlen, dass Sie Mounting-Ziele in allen Availability Zones erstellen. Wenn Sie dies tun, können Sie das Dateisystem problemlos auf EC2-Instances mounten, die Sie möglicherweise in einer der Availability Zones starten werden.

Wenn Sie ein Mounting-Ziel löschen, werden bei diesem Vorgang zwangsweise alle Dateisystem-Mounts aufgehoben. Dies könnte zu einer Störung der Instances oder Anwendungen führen, die diese Mounts verwenden. Um eine Anwendungsunterbrechung zu vermeiden, stoppen Sie die Anwendungen und heben Sie den Dateisystem-Mount auf, bevor Sie das Mounting-Ziel löschen.

Sie können ein Dateisystem immer nur in jeweils einer VPC verwenden. Sie können also immer nur in jeweils einer VPC Mounting-Ziele für das Dateisystem erstellen. Wenn Sie von einer anderen VPC auf das Dateisystem zugreifen möchten, müssen Sie zunächst die Mounting-Ziele aus der aktuellen VPC löschen. Anschließend können Sie neue Mounting-Ziele in einer anderen VPC erstellen.

- Aktualisieren der Konfiguration des Mountingziels – Wenn Sie ein Mountingziel erstellen, ordnen Sie diesem Sicherheitsgruppen zu. Eine Sicherheitsgruppe fungiert als virtuelle Firewall zur Steuerung des Datenverkehrs zu und von dem Mounting-Ziel. Sie können Regeln für eingehenden Datenverkehr hinzufügen, um den Zugriff auf das Mounting-Ziel und damit das Dateisystem zu kontrollieren. Möglicherweise möchten Sie nach dem Erstellen eines Mounting-Ziels die dem Ziel zugewiesenen Sicherheitsgruppen ändern.

Jedes Mounting-Ziel verfügt auch über eine IP-Adresse. Wenn Sie ein Mounting-Ziel erstellen, können Sie eine IP-Adresse aus dem Subnetz auswählen, in das Sie das Mounting-Ziel stellen. Wenn Sie keinen Wert angeben, wählt Amazon EFS eine nicht verwendete IP-Adresse aus dem betreffenden Subnetz aus.

Es gibt keine Amazon-EFS-Operation zum Ändern der IP-Adresse nach Erstellen eines Mountingziels. Daher können Sie die IP-Adresse nicht programmgesteuert oder mithilfe der AWS

CLLändern. Eine Änderung der IP-Adresse über die Konsole ist jedoch möglich. Im Hintergrund löscht die Konsole das Mounting-Ziel und erstellt es erneut.

 **Warning**

Wenn Sie die IP-Adresse eines Mountingziels ändern, werden dabei alle vorhandenen Dateisystem-Mounts aufgehoben. Sie müssen das Dateisystem in diesem Fall erneut mounten.

Das Dateisystem selbst ist von den Konfigurationsänderungen bezüglich der Netzwerkzugänglichkeit des Dateisystems nicht betroffen. Das Dateisystem und Ihre Daten bleiben unverändert.

In den folgenden Abschnitten finden Sie Informationen zur Verwaltung der Netzwerkzugänglichkeit des Dateisystems.

Themen

- [Erstellen oder Löschen von Mountingzielen in einer VPC](#)
- [Ändern der VPC für Ihr Mounting-Ziel](#)
- [Aktualisieren der Konfiguration von Mountingzielen](#)

Erstellen oder Löschen von Mountingzielen in einer VPC

Für den Zugriff auf ein Amazon-EFS-Dateisystem in einer VPC benötigen Sie Mountingziele. Für ein Amazon-EFS-Dateisystem gilt Folgendes:

- Sie können in jeder Availability Zone ein Mounting-Ziel erstellen.
- Falls die VPC über mehrere Subnetze in einer Availability Zone verfügt, können Sie nur in einem dieser Subnetze ein Mounting-Ziel erstellen. Alle EC2-Instances in der Availability Zone können dieses einzelne Mounting-Ziel gemeinsam verwenden.

 **Note**

Wir empfehlen Ihnen, in jeder Availability Zone ein Mounting-Ziel zu erstellen. Es gibt Kostenüberlegungen für das Mounten eines Dateisystems auf einer EC2-Instance in einer Availability Zone über ein in einer anderen Availability Zone erstelltes Mounting-Ziel. Weitere Informationen finden Sie unter [Amazon EFS](#). Wenn Sie immer ein für die

Availability Zone der Instance lokales Mountingziel verwenden, vermeiden Sie darüber hinaus ein Teilausfallszenario. Wenn die Zone des Mounting-Ziels ausfällt, können Sie über das betreffende Mounting-Ziel nicht mehr auf Ihr Dateisystem zugreifen.

Mounting-Ziele können gelöscht werden. Beim Löschen eines Mountingziels werden alle Dateisystem-Mounts über das betreffende Mountingziel zwangsweise aufgehoben. Dies könnte zu einer Störung der Instances oder Anwendungen führen, die diese Mounts verwenden. Weitere Informationen finden Sie unter [Erstellen und Verwalten von Mount-Zielen und Sicherheitsgruppen](#).

Note

Bevor Sie ein Mountingziel löschen, müssen Sie das Mounting des Dateisystems aufheben. Weitere Informationen finden Sie unter [Aufheben des Mountings von Dateisystemen](#).

Mit der AWS Management Console, der AWS CLI und der -API können Sie Mount-Ziele auf Dateisystemen erstellen und verwalten. Für bestehende Mountingziele können Sie Sicherheitsgruppen hinzufügen und entfernen oder das Mountingziel löschen. Weitere Informationen finden Sie unter [Erstellen und Verwalten von Mount-Zielen und Sicherheitsgruppen](#).

Ändern der VPC für Ihr Mounting-Ziel

Sie können ein Amazon-EFS-Dateisystem basierend auf dem Amazon-VPC-Service immer in jeweils einer VPC verwenden. Sie erstellen also Mounting-Ziele in einer VPC für Ihr Dateisystem und verwenden diese Mounting-Ziele, um Zugriff auf das Dateisystem zu ermöglichen.

Sie können das Amazon-EFS-Dateisystem von diesen Zielen mounten:

- Amazon-EC2-Instances in derselben VPC
- EC2-Instances in einer VPC, die durch VPC-Peering verbunden sind
- On-Premises-Server mithilfe von AWS Direct Connect
- On-Premises-Server über ein AWS Virtual Private Network (VPN) mithilfe von Amazon VPC

Eine VPC-Peering-Verbindung ist eine Netzwerkverbindung zwischen zwei VPCs, die den Datenverkehr zwischen diesen beiden VPCs ermöglicht. Die Verbindung kann private IPv4- oder IPv6-Adressen nutzen. Weitere Informationen dazu, wie Amazon EFS mit VPC-Peering funktioniert,

finden Sie unter [Mounten von EFS-Dateisystemen von einem anderen AWS-Konto oder einer anderen VPC](#).

Um von EC2-Instances in einer anderen VPC auf das Dateisystem zuzugreifen, müssen Sie:

- die aktuellen Mountingziele löschen
- die VPC ändern
- neue Mountingziele erstellen

Weitere Informationen zum Ausführen dieser Schritte in der finden Sie AWS Management Console unter [Gehen Sie wie folgt vor, um die VPC für ein Amazon-EFS-Dateisystem \(Konsole\) zu ändern](#).

Verwenden der -CLI

Wenn Sie ein Dateisystem in einer anderen VPC verwenden möchten, müssen Sie zunächst alle Mounting-Ziele löschen, die Sie zuvor in einer VPC erstellt haben. Anschließend können Sie neue Mounting-Ziele in einer anderen VPC erstellen. Beispiele für AWS CLI -Befehle finden Sie unter [Verwaltung der Mount-Zielen mithilfe der AWS CLI](#).

Aktualisieren der Konfiguration von Mountingzielen

Nach dem Erstellen eines Mountingziels für Ihr Dateisystem möchten Sie möglicherweise die geltenden Sicherheitsgruppen aktualisieren. Die IP-Adresse eines vorhandenen Mounting-Ziels kann nicht geändert werden. Zum Ändern der IP-Adresse müssen Sie das Mounting-Ziel löschen und ein neues mit der neuen Adresse erstellen. Durch das Löschen eines Mounting-Ziels werden alle bestehenden Dateisystem-Mounts aufgehoben.

Note

Bevor Sie ein Mountingziel löschen, müssen Sie das Mounting des Dateisystems aufheben.

Ändern einer Sicherheitsgruppe

Sicherheitsgruppen definieren den ein- und ausgehenden Zugriff. Wenn Sie einem Mounting-Ziel 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 in [Amazon-EC2-Sicherheitsgruppen](#) im Amazon-EC2-Benutzerhandbuch für Linux-Instances.

Informationen zum Ändern der Sicherheitsgruppe eines Mounting-Ziels mithilfe der finden Sie AWS Management Console unter [Verwalten von Mount-Zielen mit der Amazon-EFS-Konsole](#).

Informationen zum Ändern der Sicherheitsgruppe eines Mounting-Ziels mithilfe der finden Sie AWS CLI unter [Verwaltung der Mount-Zielen mithilfe der AWS CLI](#).

Verwalten des Dateisystemdurchsatzes

Elastic ist der Standarddurchsatzmodus und wird für die meisten Anwendungsfälle empfohlen. Mit Elastic Throughput 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.

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 Amazon-EFS-Durchsatzmodi finden Sie unter [Durchsatzmodi](#).

Note

Sie können den Durchsatzmodus und die bereitgestellte Durchsatzmenge ändern, sobald das Dateisystem verfügbar ist. 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 Dateisystemdurchsatzmodus mithilfe der Amazon-EFS-Konsole, der AWS Command Line Interface (AWS CLI) und der Amazon-EFS-API verwalten.

So verwalten Sie den Durchsatz des Dateisystems (Konsole)

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 Throughput“ oder „Bereitgestellter Durchsatz“ zu verwenden, wählen Sie Erweitert und dann Elastic oder Bereitgestellt.

Wenn Sie Bereitgestellt wählen, geben Sie unter Bereitgestellter Durchsatz (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 in Bereitgestellt 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.

- 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.

So verwalten Sie den Durchsatz des Dateisystems (CLI)

- Verwenden Sie den [update-file-system](#) CLI-Befehl oder die [UpdateFileSystem](#) API-Aktion , um den Durchsatzmodus eines Dateisystems zu ändern.

Verwaltung des Dateisystemspeichers

Um Ihre Dateisysteme so zu verwalten, dass sie während ihres gesamten Lebenszyklus kostengünstig gespeichert werden, führt die Lebenszyklusverwaltung automatisch Daten zwischen Speicherklassen um, entsprechend der Lebenszykluskonfiguration, die für das Dateisystem definiert ist. Die Lebenszykluskonfiguration besteht aus einer Reihe von Lebenszyklusrichtlinien, die definieren, wann die Daten des Dateisystems in eine andere Speicherklasse übertragen werden sollen.

Lebenszyklus-Richtlinien

Lebenszyklusrichtlinien weisen an, wann Dateien in die Speicherklassen EFS Infrequent Access (IA) und EFS Archive übergehen sollen. Dies basiert darauf, wann in der Speicherklasse „Standard“ zuletzt auf die Dateien zugegriffen wurde. Lebenszyklusrichtlinien gelten für das gesamte EFS-Dateisystem.

Die EFS-Lebenszyklusrichtlinien lauten:

- Übergang zu IA – Weist das Lebenszyklusmanagement an, wann Dateien in den Speicher für seltenen Zugriff verschoben werden sollen, der für Daten, auf die nur ein paar Mal pro Quartal zugegriffen wird, kostenoptimiert ist. Standardmäßig werden Dateien, auf die 30 Tage lang nicht im Standardspeicher zugegriffen wurde, in IA übertragen.
- Übergang ins Archiv – Weist das Lebenszyklusmanagement an, wann Dateien in die Speicherklasse Archiv verschoben werden sollen. Diese Klasse ist kostenoptimiert für Daten, auf die nur wenige Male pro Jahr zugegriffen wird. Standardmäßig werden Dateien, auf die 90 Tage lang nicht im Standardspeicher zugegriffen wurde, in „Archive“ übertragen.
- Übergang in Standard – Weist das Lebenszyklusmanagement an, ob Dateien aus IA oder Archive und zurück in den Standardspeicher überführt werden sollen, der Leselatenzen unter einer Millisekunde für Daten bereitstellt, auf die häufig zugegriffen wird. Standardmäßig werden Dateien nicht zurück in den Standardspeicher verschoben und verbleiben in der Speicherklasse IA oder Archive. 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 [Verwaltung von Lebenszyklusrichtlinien für ein Dateisystem](#).

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). Wenn auf eine Datei in Standard zugegriffen wird, wird der Lebenszyklusverwaltungs-Timer zurückgesetzt. Nachdem die Lebenszyklusverwaltung eine Datei in die Speicherklassen IA oder Archive verschoben hat, bleibt die Datei dort auf unbestimmte Zeit, es sei denn, die Richtlinie Übergang in den Standard ist festgelegt, die das Lebenszyklusmanagement anweist, Dateien beim Zugriff wieder in den Standard zu verschieben.

Metadaten-Operationen, wie die Auflistung der Inhalte 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.

Dateisystemoperationen für die Lebenszyklusverwaltung

Dateisystemoperationen für die Lebenszyklusverwaltung haben eine niedrigere Priorität als Operationen für EFS-Dateisystem-Workloads. Der Zeitaufwand für die Übergabe von Dateien an die IA- und Archive-Speicherklasse variiert und ist abhängig von Dateigröße und Dateisystem-Workload.

Datei-Metadaten, einschließlich Dateinamen, Eigentümerinformationen und Dateisystem-Verzeichnisstruktur, werden immer im Standardspeicher gespeichert, um eine konsistente Metadaten-Leistung sicherzustellen. 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.

Verwaltung von Lebenszyklusrichtlinien für ein Dateisystem

Wenn Sie ein Amazon EFS-Dateisystem erstellen, das die vom Service empfohlenen Einstellungen mithilfe der verwendeten AWS Management Console, verwenden die Lebenszyklusrichtlinien des Dateisystems die folgenden Standardeinstellungen:

- Übergang in IA ist auf 30 Tage seit dem letzten Zugriff festgelegt.

- Übergang ins Archiv ist auf 90 Tage seit dem letzten Zugriff festgelegt.
- Übergang zum Standard ist auf Keine gesetzt.

Weitere Informationen zum Erstellen eines Dateisystems mit den vom Service empfohlenen Einstellungen finden Sie unter [Schritt 1: Erstellen eines Amazon-EFS-Dateisystems](#).

Sie können die Lebenszyklusrichtlinien konfigurieren, nachdem das Dateisystem erstellt wurde oder wenn Sie ein Dateisystem mit benutzerdefinierten Einstellungen erstellen.

Zu den möglichen Werten für die Lebenszyklusrichtlinien Übergang in IA und Übergang ins Archiv gehören:

- Keine
- 1 Tag seit dem letzten Zugriff
- 7 Tage seit dem letzten Zugriff
- 14 Tage seit dem letzten Zugriff
- 30 Tage seit dem letzten Zugriff
- 60 Tage seit dem letzten Zugriff
- 90 Tage seit dem letzten Zugriff
- 180 Tage seit dem letzten Zugriff
- 270 Tage seit dem letzten Zugriff
- 365 Tage seit dem letzten Zugriff

Zu den möglichen Werten für die Lebenszyklusrichtlinie Übergang zum Standard gehören:

- Keine
- Beim ersten Zugriff

Sie können Lebenszyklusrichtlinien mithilfe der AWS Management Console und der konfigurieren AWS CLI, wie in den folgenden Verfahren beschrieben.

Verwalten von Lebenszyklusrichtlinien auf einem vorhandenen Dateisystem (Konsole)

Sie können die verwenden AWS Management Console , um die Lebenszyklusrichtlinien für ein vorhandenes Dateisystem festzulegen.

1. Melden Sie sich bei der an AWS Management Console 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. In der Lebenszyklusverwaltung können Sie die folgenden Lebenszyklusrichtlinien ändern:
 - Stellen Sie Übergang in IA auf eine der verfügbaren Einstellungen. Um das Verschieben von Dateien in den IA-Speicher zu beenden, wählen Sie Keine.
 - Stellen Sie Übergang ins Archiv auf eine der verfügbaren Einstellungen. Um das Verschieben von Dateien in den Archive-Speicher zu beenden, wählen Sie Keine.
 - 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, setzen Sie die Einstellung auf Keine.

6. Wählen Sie Änderungen speichern aus, um Ihre Änderungen zu speichern.

Verwalten von Lebenszyklusrichtlinien auf einem vorhandenen Dateisystem (CLI)

Sie können die 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 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": []
}
```

Zugriffsverwaltung auf verschlüsselte Dateisysteme

Mit Amazon EFS können Sie verschlüsselte Dateisysteme erstellen. Amazon EFS unterstützt zwei Formen der Verschlüsselung für Dateisysteme: Verschlüsselung bei der Übertragung und Verschlüsselung im Ruhezustand. Die Schlüsselverwaltung, die Sie durchführen müssen, bezieht sich nur auf die Verschlüsselung im Ruhezustand. Amazon EFS verwaltet die Schlüssel für die Verschlüsselung während der Übertragung automatisch.

Wenn Sie ein Dateisystem mit Verschlüsselung im Ruhezustand erstellen, werden Daten und Metadaten im Ruhezustand verschlüsselt. Amazon EFS verwendet AWS Key Management Service (AWS KMS) für die Schlüsselverwaltung. Wenn Sie ein Dateisystem mit Verschlüsselung im Ruhezustand erstellen, geben Sie einen AWS KMS key an. Der KMS-Schlüssel kann `aws/elasticfilesystem` (für Von AWS verwalteter Schlüssel Amazon EFS) oder ein vom Kunden verwalteter Schlüssel sein, den Sie verwalten.

Dateidaten – der Inhalt Ihrer Dateien – werden im Ruhezustand mit dem KMS-Schlüssel verschlüsselt, den Sie beim Erstellen des Dateisystems angegeben haben. Metadaten – Datei- und Verzeichnisnamen sowie Verzeichnisinhalte – werden mit dem von Amazon EFS verwalteten Schlüssel verschlüsselt.

Das EFS Von AWS verwalteter Schlüssel für Ihr Dateisystem wird als KMS-Schlüssel für die Verschlüsselung der Metadaten in Ihrem Dateisystem verwendet, z. B. Dateinamen, Verzeichnisnamen und Verzeichnisinhalte. Sie sind Eigentümer des kundenseitig verwalteten Schlüssels für die Verschlüsselung von Dateidaten (dem Inhalt Ihrer Dateien) im Ruhezustand.

Sie verwalten den Zugriff auf Ihre KMS-Schlüssel und den Inhalt Ihrer verschlüsselten Dateisysteme. Dieser Zugriff wird sowohl durch AWS Identity and Access Management (IAM)-Richtlinien als auch durch gesteuert AWS KMS. IAM-Richtlinien steuern den Zugriff eines Benutzers auf die Amazon-EFS-API actions. AWS KMS key-Richtlinien steuern den Zugriff eines Benutzers auf den KMS-Schlüssel, den Sie bei der Erstellung des Dateisystems angegeben haben. Weitere Informationen finden Sie hier:

- [IAM-Benutzer](#) im IAM-Benutzerhandbuch
- [Verwenden von Schlüsselrichtlinien in AWS KMS](#) im AWS Key Management Service - Entwicklerhandbuch
- [Verwenden von Erteilungen](#) im AWS Key Management Service -Entwicklerhandbuch

Als Schlüsseladministrator können Sie externe Schlüssel importieren. Sie können auch Schlüssel aktivieren, deaktivieren oder löschen. Der Zustand des beim Erstellen des Dateisystems mit Verschlüsselung im Ruhezustand angegebenen KMS-Schlüssels wirkt sich auf den Zugriff auf dessen Inhalt aus. Der KMS-Schlüssel muss sich im `enabled` Status befinden, damit Benutzer Zugriff auf den Inhalt eines `encrypted-at-rest` Dateisystems haben, das mit diesem Schlüssel verschlüsselt ist.

Ausführen von administrativen Aktionen für Amazon-EFS-KMS-Schlüssel

Nachfolgend erfahren Sie, wie Sie KMS-Schlüssel, die Ihrem Amazon-EFS-Dateisystem zugeordnet sind, aktivieren, deaktivieren oder löschen. Außerdem wird erläutert, wie das Dateisystem beim Ausführen dieser Aktionen reagiert.

Deaktivieren, Löschen oder Widerrufen des Zugriffs für den KMS-Schlüssel eines Dateisystems

Sie können Ihre kundenverwalteten KMS-Schlüssel deaktivieren oder löschen oder den Amazon-EFS-Zugriff für Ihre KMS-Schlüssel widerrufen. Die Deaktivierung oder das Widerrufen des Zugriffs auf Amazon EFS können rückgängig gemacht werden. KMS-Schlüssel sollten nur nach sorgfältiger Prüfung gelöscht werden, da dieser Vorgang nicht rückgängig gemacht werden kann.

Wenn Sie den KMS-Schlüssel für Ihr gemountetes Dateisystem deaktivieren oder löschen, gilt Folgendes:

- Dieser KMS-Schlüssel kann nicht als Schlüssel für neue `encrypted-at-rest` Dateisysteme verwendet werden.
- Bestehende `encrypted-at-rest` Dateisysteme, die diesen KMS-Schlüssel verwenden, funktionieren nach einer bestimmten Zeit nicht mehr.

Wenn Sie den Amazon-EFS-Zugriff für eine Erteilung bei einem vorhandenen gemounteten Dateisystem widerrufen, sind die Folgen dieselben wie beim Deaktivieren oder Löschen des zugehörigen KMS-Schlüssels. Mit anderen Worten, das `encrypted-at-rest` Dateisystem funktioniert weiterhin, funktioniert aber nach einer bestimmten Zeit nicht mehr.

Sie können den Zugriff auf ein aufgespieltes `encrypted-at-rest` Dateisystem verhindern, das über einen KMS-Schlüssel verfügt, auf den Sie den Amazon-EFS-Zugriff deaktiviert, gelöscht oder widerrufen haben. Heben Sie dazu das Mounting des Dateisystems auf und löschen Sie Ihre Amazon-EFS-Mountingziele.

Sie können ein nicht sofort löschen AWS KMS key, aber Sie können es innerhalb von 7–30 Tagen zum Löschen planen. Wenn ein KMS-Schlüssel zum Löschen vorgesehen ist, können Sie ihn nicht für kryptografische Operationen verwenden. Sie können eine geplante KMS-Schlüssellöschung auch abbrechen.

Informationen zum Deaktivieren und Reaktivieren von kundenverwalteten KMS-Schlüsseln finden Sie unter [Aktivieren und Deaktivieren von Schlüsseln](#) im AWS Key Management Service -Entwicklerhandbuch. Informationen zum Planen des Löschens von kundenverwalteten KMS-Schlüsseln finden Sie unter [Löschen von KMS-Schlüsseln](#) im AWS Key Management Service -Entwicklerhandbuch.

Verwandte Themen

- Weitere Informationen zu im Ruhezustand verschlüsselten Daten und Metadaten in Amazon EFS finden Sie unter [Datenverschlüsselung in Amazon EFS](#).
- Beispiele für Schlüsselrichtlinien finden Sie unter [Amazon-EFS-Schlüsselrichtlinien für AWS KMS](#).
- Eine Liste der AWS CloudTrail Protokolleinträge, die einem verschlüsselten Dateisystem zugeordnet sind, finden Sie unter [Amazon EFS-Protokolleinträge für encrypted-at-rest Dateisysteme](#).
- Weitere Informationen dazu, wie Sie festlegen können, welche Konten und Services auf Ihre KMS-Schlüssel zugreifen dürfen, finden Sie unter [Bestimmen des Zugriffs auf einen AWS KMS key](#) im AWS Key Management Service -Entwicklerhandbuch.

Messung: Wie Amazon EFS die Größe von Dateisystemen und Objekten meldet

In den folgenden Abschnitten wird beschrieben, wie Amazon EFS Dateigrößen und Objektgrößen innerhalb eines Dateisystems meldet.

Messung von Amazon-EFS-Dateisystemobjekten

Bei den Objekten, die Sie in einem Amazon-EFS-System anzeigen können, handelt es sich um normale Dateien, Verzeichnisse, symbolische Verknüpfungen 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 über das NFSv4.1-Attribut `space_used` den für ein Objekt verwendeten Speicherplatz meldet, ist dabei auch die aktuelle gemessene Datengröße des Objekts, nicht jedoch seine Metadatengröße einbezogen. 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`-Hilfsprogramms für eine leere Datei, die die `-k` Option enthält, die Ausgabe in Kilobyte zurückzugeben.

```
$ du -k file
4      file
```

Das folgende Beispiel zeigt, wie Sie das `stat` Dienstprogramm für eine leere Datei verwenden, um die Festplattennutzung 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 Amazon-EFS-Dateisystems

Die gemessene Größe eines Amazon-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 Amazon-EFS-Dateisystem wie folgt anzeigen:

- Mit dem [-describe-file-systems](#) AWS CLI Befehl und der [DescribeFileSystem](#) -API-Operation enthält die Antwort Folgendes:

```
"SizeInBytes":{
    "Timestamp": 1403301078,
    "Value": 29313744866,
    "ValueInIA": 675432,
    "ValueInStandard": 29312741784
    "ValueInArchive": 327650
}
```

Wobei die gemessene Größe von auch verwendet `ValueInStandard` wird, um Ihre E/A-Durchsatz-Baseline und Burst-Raten für Dateisysteme zu bestimmen, die den [Bursting-Durchsatzmodus](#) verwenden.

- Zeigen Sie die `-StorageBytes` CloudWatch Metrik an, die die gemessene Gesamtgröße der Daten in jeder Speicherklasse anzeigt. Für weitere Informationen über die `StorageBytes`-Metrik siehe [Amazon- CloudWatch Metriken für Amazon EFS](#).
- Führen Sie in Linux bei der Terminal-Eingabeaufforderung einer EC2-Instance den Befehl `df` aus.

Verwenden Sie nicht den `du` Befehl im Stammverzeichnis des Dateisystems für Speichermessungen, da die Antwort nicht die vollständigen Satzdaten widerspiegelt, die für die Messung Ihres Dateisystems verwendet werden.

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 der Speicherklassen Infrequent Access und Archive

Die Speicherklassen EFS Infrequent Access (IA) und Archive werden in Schritten von 4 KiB gemessen und haben eine Mindestabrechnungsgebühr pro Datei von 128 KiB . IA- und Archivdateimetadaten (2 KiB pro Datei) werden immer in der Standardspeicherklasse gespeichert und gemessen. Die Unterstützung für Dateien kleiner als 128 KiB ist nur für Lebenszyklusrichtlinien verfügbar, die am oder nach dem 26. November 2023 um 12:00 Uhr PT aktualisiert wurden. Der Datenzugriff für IA- und Archive-Speicher wird in Schritten von 128 KiB gemessen.

Sie können die `-StorageBytes` CloudWatch Metrik verwenden, um die gemessene Datengröße in jeder der Speicherklassen anzuzeigen. Die Metrik zeigt auch die Gesamtzahl der Bytes an, die von 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 weitere Informationen über die `StorageBytes`-Metrik siehe [Amazon- CloudWatch Metriken für Amazon EFS](#).

Mess-Durchsatz

Amazon EFS misst den Durchsatz für Leseanforderungen mit einem Drittel der Rate der anderen E/A-Operationen des Dateisystems. Wenn Sie beispielsweise 30 Mebibyte pro Sekunde (MiBps) sowohl Lese- als auch Schreibdurchsatz erreichen, zählt der Leseanteil als 10 MiBps des effektiven Durchsatzes, der Schreibanteil als 30 MiBps und der gemessene Gesamtdurchsatz ist 40 MiBps. Dieser kombinierte Durchsatz, der an die Verbrauchsraten angepasst ist, spiegelt sich in der `-MeteredIOBytes` CloudWatch Metrik wider.

Messung des Elastic-Durchsatzes

Wenn der elastische Durchsatzmodus für ein Dateisystem aktiviert ist, zahlen Sie nur für die Menge der Metadaten und Daten, die aus dem Dateisystem gelesen oder in das Dateisystem geschrieben werden. Amazon-EFS-Dateisysteme, die den elastischen Durchsatzmodus verwenden, messen Metadatenlesevorgänge als Lesevorgänge und Metadatenschreibvorgänge als Schreibvorgänge. Metadatenoperationen werden in Schritten von 4 KiB und Datenoperationen in Schritten von 32 KiB gemessen.

Messen des bereitgestellten Durchsatzes

Für Dateisysteme, die den Modus des bereitgestellten Durchsatzes verwenden, zahlen Sie nur für die Zeit, für die der Durchsatz aktiviert ist. Amazon EFS misst Dateisysteme mit aktiviertem Modus des bereitgestellten Durchsatzes einmal pro Stunde. Für die Messung, wenn der Modus des bereitgestellten Durchsatzes weniger als eine Stunde festgelegt ist, berechnet Amazon EFS den Zeitdurchschnitt mit einer Genauigkeit von Millisekunden.

Verwalten der Amazon EFS-Dateisystemkosten mitAWSBudgets

Sie können Ihre Amazon EFS-Dateisystemkosten mitAWSBudgets.

Sie können von der AWS Billing and Cost Management-Konsole aus mit AWS-Budgets arbeiten. Wenn Sie AWS-Budgets verwenden möchten, erstellen Sie ein monatliches Kostenbudget für Ihre EFS-Dateisysteme. Sie können Ihr Budget so einrichten, dass Sie benachrichtigt werden, wenn Ihre Kosten den budgetierten Betrag überschreiten. Dann können Sie Anpassungen vornehmen, um Ihr Budget bei Bedarf einzuhalten.

Mit der Verwendung sind Kosten verbundenAWSBudgets. Für normaleAWS-Kontensind Ihre ersten zwei Budgets kostenlos. Weitere Informationen zuAWSBudgets, einschließlich Kosten, siehe[Verwalten der Kosten mit Budgets](#)imAWS Billing-Benutzerhandbuchaus.

Sie können benutzerdefinierte Budgets für Ihre Amazon EFS-Kosten und -Nutzung im Konto festlegen, AWS-Region, Service- oder Tag-Level unter Verwendung von Budgetparametern. Im folgenden Abschnitt finden Sie eine allgemeine Beschreibung zum Einrichten eines Kostenbudgets auf einem EFS-Dateisystem mit AWS-Budgets. Dazu verwenden Sie Kostenzuordnungs-Tags.

Voraussetzungen

Zum Ausführen der in den folgenden Abschnitten referenzierten Prozeduren stellen Sie sicher, dass Sie über Folgendes verfügen:

- Ein EFS-Dateisystem
- Eine AWS Identity and Access Management (IAM)-Richtlinie mit den folgenden Berechtigungen:
 - Sorgen Sie für Zugriff auf die AWS Billing and Cost Management-Konsole.
 - Die Fähigkeit, die Aktionen „elasticfilesystem:CreateTags“ und „elasticfilesystem:DescribeTags“ auszuführen.

Erstellen eines monatlichen Kostenbudgets für ein EFS-Dateisystem

Die Erstellung eines monatlichen Kostenbudgets für Ihr Amazon EFS-Dateisystem mithilfe von Tags erfolgt in drei Schritten.

So erstellen Sie ein monatliches Kostenbudget für Ihr EFS-Dateisystem mithilfe von Tags

1. Erstellen Sie ein Tag, mit dem das Dateisystem, für das Sie die Kosten verfolgen möchten, identifiziert werden soll. Um zu erfahren wie, siehe [Markieren der Amazon-EFS-Ressourcen](#).
2. Aktivieren Sie in der Konsole Billing and Cost Management das Tag als Kostenzuordnungs-Tag. Eine ausführliche Vorgehensweise finden Sie unter [Aktivieren von benutzerdefinierten Kostenzuordnungs-Tags](#) im AWS Billing-Benutzerhandbuch.
3. In der Konsole Billing and Cost Management unter Budgetserstellen Sie ein monatliches Kostenbudget in AWS Budgets. Eine ausführliche Vorgehensweise finden Sie unter [Erstellen eines Kostenbudgets](#) im AWS Billing-Benutzerhandbuch.

Nachdem Sie Ihr monatliches EFS-Kostenbudget erstellt haben, können Sie es im Budgets-Dashboard anzeigen, in dem die folgenden Budgetdaten angezeigt werden:

- Ihre aktuellen Kosten und Ihre Nutzung während des Budgetzeitraums für ein Budget.
- Ihre budgetierten Kosten für den Budgetzeitraum.

- Ihre Prognosekosten für den Budgetzeitraum.
- Einen Prozentwert, der die tatsächlichen Kosten im Vergleich zur veranschlagten Menge zeigt.
- Ein Prozentwert, der die tatsächlichen Prognosekosten im Vergleich zur veranschlagten Menge zeigt

Weitere Informationen zum Anzeigen des EFS-Kostenbudgets finden Sie unter [Anzeigen Ihrer Budgets](#) im AWS Billing-Benutzerhandbuch.

Status des Dateisystems

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	Beschreibung
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 eines Amazon-EFS-Dateisystems .
GELÖSCHT	Amazon EFS hat das Dateisystem als Antwort auf eine vom Benutzer initiierte Löschanforderung gelöscht. Weitere Informationen finden Sie unter Löschen eines Amazon-EFS-Dateisystems .
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.

Status des Dateisystems	Beschreibung
	<p>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 hier:</p> <ul style="list-style-type: none">• Wiederherstellen eines Wiederherstellungspunkts.• EFS-Speicherklassen• Replizieren von Dateisystemen

Überwachen von Amazon EFS

Die Überwachung ist wichtig, um die Zuverlässigkeit, Verfügbarkeit und Leistung von Amazon EFS und Ihrer - AWS Lösungen aufrechtzuerhalten. Wir empfehlen Ihnen, Überwachungsdaten von allen Teilen Ihrer - AWS Lösung zu sammeln, damit Sie Ausfälle an mehreren Punkten 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.

Sie können beispielsweise mit Amazon EFS den Netzwerkdurchsatz, die E/A-Leistung für Lese-, Schreib- und/oder Metadaten-Operationen, Client-Verbindungen und Burst-Gutschriften 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.

Überwachungstools

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 -Alarmer – 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 eine Reihe von Zeiträumen basieren. Die Aktion ist eine Benachrichtigung, die an ein Amazon Simple Notification Service (Amazon SNS)-Thema oder eine Amazon EC2 Auto Scaling-Richtlinie gesendet wird. - CloudWatch Alarmer rufen keine Aktionen auf, nur weil sie sich in einem bestimmten Status befinden. Der Status muss geändert und für eine bestimmte Anzahl von Zeiträumen beibehalten worden sein. Weitere Informationen finden Sie unter [Überwachen von Amazon EFS mit Amazon CloudWatch](#).
- Amazon CloudWatch Logs – Überwachen, Speichern und Zugriff auf Ihre Protokolldateien von AWS CloudTrail oder anderen Quellen. Weitere Informationen finden Sie unter [Überwachen von Protokolldateien](#) im Amazon- CloudWatch 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 Amazon- CloudWatch Benutzerhandbuch.
- AWS CloudTrail Protokollüberwachung – Teilen Sie Protokolldateien zwischen Konten, überwachen Sie CloudTrail Protokolldateien in Echtzeit, indem Sie sie an - CloudWatch Protokolle senden, schreiben Sie Anwendungen zur Protokollverarbeitung in Java und überprüfen Sie, ob sich Ihre Protokolldateien nach der Bereitstellung durch nicht geändert haben CloudTrail. Weitere Informationen finden Sie unter [Arbeiten mit CloudTrail Protokolldateien](#) im AWS CloudTrail - Benutzerhandbuch.

Manuelle Überwachungstools

Ein weiterer wichtiger Bestandteil der Überwachung von Amazon EFS ist die manuelle Überwachung derjenigen Elemente, die die Amazon- CloudWatch Alarmer nicht abdecken. Die Amazon-EFS CloudWatch- und andere AWS Management Console Dashboards bieten einen at-a-glance Überblick über den Zustand Ihrer AWS Umgebung. Zudem empfehlen wir die Überprüfung der Protokolldateien auf dem Dateisystem.

- In der Amazon-EFS-Konsole finden Sie die folgenden Elemente für Ihre Dateisysteme:
 - Die aktuelle gemessene Größe
 - Die Anzahl der Mounting-Ziele
 - Lebenszyklusstatus
- CloudWatch Die -Startseite zeigt:
 - Aktuelle Alarmer und Status
 - Diagramme mit Alarmen und Ressourcen
 - Servicestatus

Darüber hinaus können Sie mit Folgendes CloudWatch tun:

- 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.

Überwachen von Amazon EFS mit Amazon CloudWatch

Sie können Dateisysteme mit Amazon überwachen CloudWatch, das Rohdaten von Amazon EFS sammelt und in lesbare Metriken verarbeitet, die nahezu in Echtzeit vorliegen. 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-Metrikdaten automatisch CloudWatch in Abständen von 1 Minute an gesendet, sofern für einige einzelne Metriken nicht anders angegeben. Die Amazon-EFS-Konsole zeigt eine Reihe von Diagrammen an, die auf den Rohdaten von Amazon basieren CloudWatch. Abhängig von Ihren Anforderungen ziehen Sie es möglicherweise vor, Daten für Ihre Dateisysteme von CloudWatch anstelle der Diagramme in der Konsole abzurufen.

Weitere Informationen zu Amazon CloudWatch finden Sie im [Amazon- CloudWatch Benutzerhandbuch](#).

Themen

- [Amazon- CloudWatch Metriken für Amazon EFS](#)
- [Wie verwende ich die Amazon-EFS-Metriken?](#)
- [Verwenden von Metrikberechnungen mit Amazon EFS](#)
- [Überwachung des Erfolgs- oder Fehlerstatus des Mount-Versuchs](#)
- [Zugreifen auf CloudWatch Metriken](#)
- [Erstellen von CloudWatch Alarmen zur Überwachung von Amazon EFS](#)

Amazon- CloudWatch Metriken für Amazon EFS

Amazon-EFS-Metriken verwenden den EFS-Namespace und stellen Metriken für eine einzelne Dimension (FileSystemId) bereit. Die Dateisystem-ID kann der Amazon-EFS-Konsole entnommen werden und hat das Format fs-abcdef0123456789a.

Der AWS/EFS-Namespace enthält die folgenden Metriken.

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.

Einheiten: Sekunden

Gültige Statistiken: Minimum, Maximum, Average

PercentIOLimit

Zeigt, wie nah sich ein Dateisystem am E/A-Limit des Allzweck-Leistungsmodus befindet.

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 den elastischen Durchsatz verwenden, spiegelt dieser Wert den maximalen Schreibdurchsatz des Dateisystems wider.
- Wenn bei Dateisystemen, die den bereitgestellten Durchsatz verwenden, die Datenmenge, die in der EFS-Archive-Speicherklasse gespeichert ist, Ihrem Dateisystem ermöglicht, einen höheren Durchsatz zu erzielen, als Sie bereitgestellt haben, spiegelt diese Metrik den höheren Durchsatz statt die bereitgestellte Menge wider.
- Für Dateisysteme im Bursting-Durchsatzmodus 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 Metadatenvorgänge, wobei Lesevorgänge mit einem Drittel der Rate anderer Vorgänge gemessen werden.

Sie können einen [CloudWatch metrischen mathematischen Ausdruck](#) erstellen, der MeteredIOBytes mit vergleichtPermittedThroughput. 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

TotalIOBytes

Die tatsächliche Bytezahl für jede Dateisystemoperation, einschließlich Datenlese-, Datenschreib- und Metadatenoperationen. Dabei handelt es sich um die tatsächliche Menge, die Ihre Anwendung generiert, und nicht um den Durchsatz, mit dem das Dateisystem gemessen wird. Sie ist möglicherweise höher als die unter PermittedThroughput angegebenen Zahlen.

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 Anzahl von Byte für jede Dateisystemleseoperation.

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 Anzahl von Byte für jede Dateisystem-Schreiboperation.

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 Anzahl von Byte für jede Metadatenoperation.

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`

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 an 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 und EFS Archive werden Dateien kleiner als 128KiB auf 128KiBgerundet.
- **Standard** ist die gemessene Größe (in Byte) der Daten, die in der EFS-Standard-Speicherklasse gespeichert sind.
- **IA** ist die gemessene Größe (in Byte) der Daten, die in der Speicherklasse EFS Infrequent Access gespeichert sind.
- **IASizeOverhead** ist die Differenz (in Byte) zwischen der tatsächlichen Datengröße in der Speicherklasse EFS Infrequent Access (in der -IADimension angegeben) und der Menge, um die sie auf 128KiBgerundet wird, wenn das Dateisystem kleiner als 128KiB ist.
- **Archive** ist die gemessene Datengröße (in Byte) in der EFS-Archive-Speicherklasse. Diese Zahl gibt die tatsächliche Größe der im Archivspeicher gespeicherten Daten an, bevor kleine Dateien auf 128KiB.
- **ArchiveSizeOverhead** ist die Differenz (in Byte) zwischen der tatsächlichen Datengröße in der EFS-Archive-Speicherklasse (in der -ArchiveDimension angegeben) und der Menge, um die sie auf 128KiBgerundet wird, wenn das Dateisystem kleiner als 128KiB ist.

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).

In gemeldete Byte CloudWatch

Amazon-EFS CloudWatch -Metriken werden als Rohbytes gemeldet. Bytes werden nicht auf eine Dezimalzahl oder ein binäres Vielfaches der Einheit gerundet. Beachten Sie dies beim Berechnen

Ihrer Burst-Rate mithilfe der Daten, die Sie von den Metriken erhalten. Weitere Informationen zum Bursting finden Sie unter [Bursting-Durchsatz](#).

Wie verwende ich die Amazon-EFS-Metriken?

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 Summe Statistik der <code>TotalIOBytes</code> -Metrik überwachen, um Ihren Durchsatz zu sehen.
Wie kann ich die Anzahl der Amazon-EC2-Instances, die mit einem Dateisystem verbunden sind, nachverfolgen?	Sie können die Summe-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 .

Verwenden von CloudWatch Metriken zur Überwachung der Durchsatzleistung

Die CloudWatch Metriken für die Durchsatzüberwachung – `TotalIOBytes`, `ReadIOBytesWriteIOBytes`, und `MetadataIOBytes` – stellen den tatsächlichen Durchsatz dar, den Sie auf Ihrem Dateisystem erreichen. 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` mit `vergleichtPermittedThroughput`.

PermittedThroughput misst die Menge des zulässigen Durchsatzes für das Dateisystem. Dieser Wert basiert auf einer der folgenden Methoden:

- Bei Dateisystemen mit Elastic-Durchsatz spiegelt dieser Wert den maximalen Schreibdurchsatz des Dateisystems wider.
- Wenn bei Dateisystemen, die den bereitgestellten Durchsatz verwenden, die Datenmenge, die in der EFS-Archive-Speicherklasse gespeichert ist, Ihrem Dateisystem ermöglicht, einen höheren Durchsatz zu erzielen, als Sie bereitgestellt haben, spiegelt diese Metrik den höheren Durchsatz statt 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 zu Elastic-Durchsatz oder Bereitgestellter Durchsatz wechseln, um zusätzlichen Durchsatz zu erhalten.

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 Metrikberechnungen mit Amazon EFS

Mithilfe von Metrikberechnungen können Sie mehrere CloudWatch Metriken abfragen und mathematische Ausdrücke verwenden, um neue Zeitreihen basierend auf diesen Metriken zu erstellen. Sie können die resultierenden Zeitreihen in der CloudWatch Konsole visualisieren und 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 zu Metrikberechnungen finden Sie unter [Verwenden von Metrikberechnungen](#) im Amazon CloudWatch -Benutzerhandbuch.

Im Folgenden finden Sie einige nützliche Ausdrücke für Metrikberechnungen mit Amazon EFS.

Themen

- [Metrikberechnung: Durchsatz in MiBps](#)
- [Metrikberechnung: Durchsatz in Prozent](#)
- [Metrikberechnung: Prozentsatz der zulässigen Auslastung des Durchsatzes](#)

- [Metrikberechnung: Durchsatz in IOPS](#)
- [Metrikberechnung: Prozentsatz der IOPS](#)
- [Metrikberechnung: Durchschnittliche E/A-Größe in KiB](#)
- [Verwenden von Metrikberechnungen über eine AWS CloudFormation -Vorlage für Amazon EFS](#)

Metrikberechnung: 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	Verwendbare Metriken	Statistik	Intervall
m1	<ul style="list-style-type: none"> • <code>DataReadIOBytes</code> • <code>DataWriteIOBytes</code> • <code>MetadataIOBytes</code> • <code>TotalIOBytes</code> 	sum	1 Minute

ID und Ausdruck Ihrer Metrikberechnung lauten wie folgt.

ID	Expression
e1	<code>(m1/1048576)/PERIOD(m1)</code>

Metrikberechnung: Durchsatz in Prozent

Dieser Metrikberechnungsausdruck berechnet den Prozentsatz am Gesamtdurchsatz für die verschiedenen E/A-Typen – zum Beispiel den Prozentsatz des Gesamtdurchsatzes, der durch Leseanforderungen entsteht. 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	Verwendbare Metrik oder Metriken	Statistik	Intervall
m1	• TotalIOBytes	sum	1 Minute
m2	• DataReadIOBytes	sum	1 Minute

ID und Ausdruck Ihrer Metrikberechnung lauten wie folgt.

ID	Expression
e1	$(m2 * 100) / m1$

Metrikberechnung: Prozentsatz der zulässigen Auslastung des Durchsatzes

Um den Prozentsatz der zulässigen Durchsatzauslastung (MeteredIOBytes) für einen Zeitraum zu berechnen, multiplizieren Sie zunächst den Durchsatz in MiBps mit 100. Dann teilen Sie das Ergebnis durch die durchschnittliche Statistik von , die für denselben Zeitraum in MiB PermittedThroughput konvertiert wurde.

Angenommen, Ihre Beispiellogik ist die folgende: (metrischer mathematischer Ausdruck für den Durchsatz in MiBps x 100 (um in Prozentsatz zu konvertieren)) (Summe von PermittedThroughput 1 048 576 (um Bytes in MiB zu konvertieren))

Dann lauten Ihre CloudWatch Metrikinformationen wie folgt.

ID	Verwendbare Metrik oder Metriken	Statistik	Intervall
m1	MeteredIOBytes	sum	1 Minute
m2	Permitted Throughput	Durchschnitt	1 Minute

ID und Ausdruck Ihrer Metrikberechnung lauten wie folgt.

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 Metrikinformationen wie folgt.

ID	Verwendbare Metriken	Statistik	Intervall
m1	• DataReadIOBytes	Beispiellanzahl	1 Minute

ID	Verwendbare Metriken	Statistik	Intervall
	<ul style="list-style-type: none"> • DataWriteIOBytes • MetadataIOBytes • TotalIOBytes 		

ID und Ausdruck Ihrer Metrikberechnung lauten wie folgt.

ID	Expression
e1	m1/PERIOD(m1)

Metrikberechnung: Prozentsatz der IOPS

Zum Berechnen der IOPS in Prozent pro Sekunde der verschiedenen E/A-Typen (DataReadIOBytes, DataWriteIOBytes oder MetadataIOBytes) für ein Intervall multiplizieren Sie zunächst die entsprechende Beispiellanzahlstatistik mit 100. Teilen Sie dann diesen Wert durch die Beispiellanzahlstatistik von TotalIOBytes für das gleiche Intervall.

Angenommen, Ihre Beispiellogik sieht wie folgt aus: (Beispiellanzahl von MetadataIOBytes x 100 (zu konvertieren in Prozent)) ÷ Beispiellanzahl von TotalIOBytes

Dann lauten Ihre CloudWatch Metrikinformationen wie folgt.

ID	Verwendbare Metriken	Statistik	Intervall
m1	• TotalIOBytes	Beispiellanzahl	1 Minute
m2	<ul style="list-style-type: none"> • DataReadIOBytes • DataWriteIOBytes 	Beispiellanzahl	1 Minute

ID	Verwendbare Metriken	Statistik	Intervall
	<ul style="list-style-type: none"> MetadataIOBytes 		

ID und Ausdruck Ihrer Metrikberechnung lauten wie folgt.

ID	Expression
e1	$(m2 \times 100) / m1$

Metrikberechnung: Durchschnittliche E/A-Größe in KiB

Zum Berechnen der durchschnittlichen E/A-Größe (in KiB) für ein Intervall dividieren Sie die entsprechende Summenstatistik für die Metrik DataReadIOBytes, DataWriteIOBytes oder MetadataIOBytes durch die gleiche Beispiellanzahlstatistik dieser Metrik.

Angenommen, Ihre Beispiellogik sieht wie folgt aus: $(\text{Summe von DataReadIOBytes} \div 1.024 \text{ (zu konvertieren in KiB)}) \div \text{Beispiellanzahl von DataReadIOBytes}$

Dann lauten Ihre CloudWatch Metrikinformationen wie folgt.

ID	Verwendbare Metriken	Statistik	Intervall
m1	<ul style="list-style-type: none"> DataReadIOBytes DataWriteIOBytes MetadataIOBytes 	sum	1 Minute
m2	<ul style="list-style-type: none"> DataReadIOBytes DataWriteIOBytes 	Beispiellanzahl	1 Minute

ID	Verwendbare Metriken	Statistik	Intervall
	<ul style="list-style-type: none"> • MetadataI 0Bytes 		

ID und Ausdruck Ihrer Metrikberechnung lauten wie folgt.

ID	Expression
e1	$(m1/1024)/m2$

Verwenden von Metrikberechnungen über eine AWS CloudFormation -Vorlage für Amazon EFS

Sie können metrische mathematische Ausdrücke auch über AWS CloudFormation Vorlagen erstellen. Eine solche Vorlage können Sie in den [AmazonEFSEFS-Tutorials](#) auf herunterladen und zur Verwendung anpassen GitHub. Weitere Informationen zur Verwendung von AWS CloudFormation Vorlagen finden Sie unter [Arbeiten mit AWS CloudFormation Vorlagen](#) im AWS CloudFormation - Benutzerhandbuch.

Überwachung des Erfolgs- oder Fehlerstatus des Mount-Versuchs

Sie können Amazon CloudWatch Logs verwenden, um den Erfolg oder Misserfolg von Mountingversuchen 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 Protokolle verwendet, um den Erfolg oder Misserfolg der Mounting-Versuche des Dateisystems zu überwachen.

So aktivieren Sie die Benachrichtigung über erfolgreiche oder fehlgeschlagene Mountingversuche in - CloudWatch Protokollen

1. Installieren Sie `amazon-efs-utils` auf der EC2-Instance, die das Dateisystem mountet. Weitere Informationen finden Sie unter [Verwendung von AWS Systems Manager zur automatischen Installation oder Aktualisierung von Amazon EFS-Clients](#) 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 von botocore](#).
3. Aktivieren Sie die Funktion CloudWatch Protokolle in `amazon-efs-utils`. Wenn Sie verwenden, AWS Systems Manager um zu installieren und zu konfigurieren `amazon-efs-utils`, erfolgt die CloudWatch Protokollierung automatisch für Sie. 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 Protokolle 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
```

4. Optional können Sie die Namen von CloudWatch Protokollgruppen konfigurieren und die Protokollaufbewahrungstage in der `efs-utils.conf` Datei festlegen. Wenn Sie CloudWatch für jedes bereitgestellte Dateisystem separate Protokollgruppen in haben möchten, fügen Sie `efs-utils.conf` wie folgt `{fs_id}` am Ende des `log_group_name` Feldes in Datei hinzu:

```
[cloudwatch-log]
log_group_name = /aws/efs/utis/{fs_id}
```

5. Hängen Sie die `AmazonElasticFileSystemsUtils` AWS verwaltete Richtlinie an die IAM-Rolle an, die Sie an die EC2-Instance angehängt 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"
```

So zeigen Sie den Mount-Status in - CloudWatch Protokollen an

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 Protokollgruppe /aws/efs/utils 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.

Zugreifen auf CloudWatch Metriken

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

Die folgenden Verfahren zeigen, wie Sie mithilfe dieser verschiedenen Tools auf die Metriken zugreifen können.

So zeigen Sie CloudWatch Metriken und Alarmer in der Amazon-EFS-Konsole an

1. Melden Sie sich bei der an AWS Management Console 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.

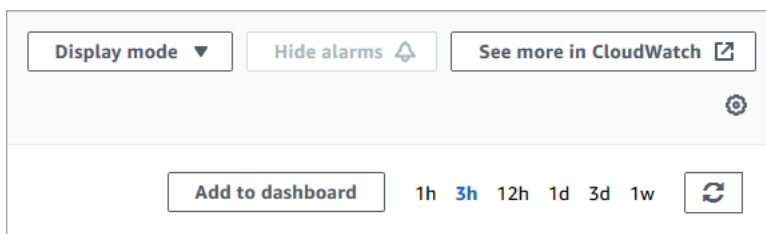
4. Wählen Sie Überwachung aus, um die Seite mit den Dateisystem-Metriken anzuzeigen.

Auf der Seite Dateisystemmetriken wird ein Standardsatz von CloudWatch Metriken für das Dateisystem angezeigt. Alle CloudWatch Alarme, die Sie konfiguriert haben, werden auch mit diesen Metriken angezeigt. Bei Dateisystemen, die den „Max. E/A“-Leistungsmodus verwenden, beinhaltet der Standardsatz von Metriken den Burst-Guthabensaldo anstelle von „Prozent E/A-Limit“. Sie können die Standardeinstellungen überschreiben, indem Sie das Dialogfeld Metrikeinstellungen verwenden, auf das Sie zugreifen, indem Sie die Einstellungen öffnen.

Note

Die Metrik Durchsatzauslastung (%) ist keine CloudWatch Metrik und wird mithilfe von CloudWatch Metrikberechnungen 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.
- Ein- oder Ausblenden aller für das Dateisystem konfigurierten CloudWatch Alarme.
- Wählen Sie Weitere Informationen in CloudWatch anzeigen, um die Metriken in 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.

So zeigen Sie Metriken mit der CloudWatch Konsole an

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.

So greifen Sie über die auf Metriken zu AWS CLI

- Verwenden Sie den Befehl [list-metrics](#) mit dem --namespace "AWS/EFS"-Namespace. Weitere Informationen finden Sie in der [AWS CLI -Befehlsreferenz](#).

So greifen Sie über die CloudWatch API auf Metriken zu

- Rufen Sie die folgende Seite auf [GetMetricStatistics](#). Weitere Informationen finden Sie in der [Amazon CloudWatch -API-Referenz](#).

Erstellen von CloudWatch Alarmen zur Überwachung von Amazon EFS

Sie können einen CloudWatch Alarm erstellen, der eine Amazon 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.

Alarme rufen nur Aktionen für anhaltende Statusänderungen auf. CloudWatch Alarme rufen keine Aktionen nur auf, weil sie sich in einem bestimmten Status befinden. Der Status muss geändert 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 durchzusetzen. 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 und verwenden, AWS CloudTrail um die Erstellung eines Dateisystems zu erkennen und zu überprüfen, ob die Verschlüsselung im Ruhezustand aktiviert ist. Weitere Informationen finden Sie unter [Exemplarische Anleitung: Erzwingen der Verschlüsselung auf einem Amazon EFS-Dateisystem im Ruhezustand](#).

 Note

Derzeit können Sie keine Verschlüsselung während der Übertragung erzwingen.

Im folgenden Verfahren wird beschrieben, wie Sie Alarme für Amazon EFS erstellen.

So richten Sie Alarme mithilfe der CloudWatch Konsole ein

1. Melden Sie sich bei der an AWS Management Console 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 eine E-Mail CloudWatch senden möchten, wenn der Alarmstatus erreicht ist, wählen Sie im Feld Wann immer dieser Alarm: den Status ALARM aus. 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.

So richten Sie einen Alarm mithilfe der ein AWS CLI

- Rufen Sie die folgende Seite auf [put-metric-alarm](#). Weitere Informationen finden Sie in der [AWS CLI -Befehlsreferenz](#).

So richten Sie einen Alarm mithilfe der CloudWatch API ein

- Rufen Sie die folgende Seite auf [PutMetricAlarm](#). 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 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 die neusten Ereignisse in Ihr(em) AWS-Konto anzeigen, suchen und herunterladen. Weitere Informationen finden Sie unter [Ereignisse mit CloudTrail Ereignisverlauf anzeigen](#).

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 dieser für alle AWS-Regionen. Der Trail protokolliert Ereignisse aus allen AWS-Regionen in der

AWS-Partition und stellt die Protokolldateien in dem Amazon-S3-Bucket bereit, den Sie angeben. 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:

- [Übersicht zum Erstellen eines Trails](#)
- [CloudTrail Unterstützte Dienste und Integrationen](#)
- [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-API-Aufrufe](#) 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 verbundenen Benutzer gesendet wurde.
- Gibt an, ob die Anforderung aus einem anderen AWS-Service gesendet wurde

Weitere Informationen finden Sie im [CloudTrail UserIdentity-Element](#) im AWS CloudTrail Benutzerhandbuch.

Grundlegendes zu Amazon EFS-Protokolldateieinträgen

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 dienstverknüpfte 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 serviceverknüpften Rollen für Amazon EFS](#).

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der eine `CreateServiceLinkedRole` Aktion demonstriert, wenn Amazon EFS die `AWSServiceRoleForAmazonElasticFileSystem` 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":
"111122223333-10-111122223333Statement111122223333Action111122223333AssumeRole111122223333Effect%22%3A%20%22Allow%22%2C%20%22Principal%22%3A%20%7B%22Service%22%3A%20%5B%22elasticfilesystem.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 die EFS-Authentifizierung

Amazon EFS-Autorisierung für NFS-Clients, Emittiert 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. Ein 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 nichts enthältClientMount. Weitere Informationen zur EFS-Autorisierung finden Sie unter [Verwendung von IAM zur Steuerung des Dateisystemdatenzugriffs](#).

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag, der ein NewClientConnection Ereignis veranschaulicht.

```
{
  "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

Amazon EFS bietet Ihnen die Möglichkeit, Verschlüsselung im Ruhezustand, Verschlüsselung bei der Übertragung oder beides für Ihre Dateisysteme zu 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"
```

Amazon-EFS-Leistung

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](#)

Zusammenfassung der Leistung

Die Leistung des Dateisystems wird in der Regel anhand der Dimensionen Latenz, Durchsatz und Eingabe-/Ausgabevorgänge 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 maximale E/A-Leistungsmodus hat höhere Latenzen pro Vorgang als der Allzweck-Leistungsmodus. Für eine schnellere Leistung empfehlen wir, immer den Allzweck-Leistungsmodus zu verwenden. Weitere Informationen finden Sie unter [Leistungsmodi](#).

- Durchsatzmodus – Elastisch, Bereitgestellt oder Bursting

In der folgenden Tabelle werden die Leistungsspezifikationen für Dateisysteme im Allzweck-Leistungsmodus und die möglichen verschiedenen Kombinationen von Dateisystemtyp und Durchsatzmodus beschrieben.

Leistungsspezifikationen für Dateisysteme, die den Allzweck-Leistungsmodus verwenden

Konfiguration von Speicher und Durchsatz		Latency		Maximale IOPS		Maximaler Durchsatz		
Dateisystemtyp	Durchsatzmodus	Lesevorgänge	Schreibvorgänge	Lesevorgänge	Schreibvorgänge	Per-file-system lesen ¹	Per-file-system Schreiben ¹	Lesen/Schreiben pro Client
Regional	Elastic	Nur 250 µs	As low as 2.7 ms	90,000–250,000 ²	50,000	3–20 GiBps	1–5 GiBps	500 MiBps
Regional	Provisioned	Nur 250 µs	As low as 2.7 ms	55,000	25,000	3–10 GiBps	1–3,33 GiBps	500 MiBps
Regional	Bursting	Nur 250 µs	As low as 2.7 ms	35,000	7,000	3–5 GiBps	1–3 GiBps	500 MiBps
One Zone	Elastic, Provisioned, or Bursting	Nur 250 Mikrosekunden (µs)	Nur 1,6 Millisekunden (ms)	35,000	7,000	3–5 Gibibyte pro Sekunde (GiBps)	1–3 GiBps	500 mebibytes per second (MiBps)

 Note

Fußnoten:

1. Der maximale Lese- und Schreibdurchsatz hängt von der AWS-Region ab. Ein Durchsatz, der den maximalen Durchsatz einer AWS-Region überschreitet, erfordert eine Erhöhung des Durchsatzkontingents. Jede Anforderung eines zusätzlichen Durchsatzes wird vom AmazonEFSEFS-Serviceteam auf der case-by-case Basis berücksichtigt. Die

Genehmigung kann von Ihrer Art des Workloads abhängen. Weitere Informationen zum Anfordern einer Kontingenterhöhung finden Sie unter [Amazon-EFS-Kontingente und -Limits](#).

2. Dateisysteme, die den elastischen Durchsatz verwenden, können maximal 90.000 Lesevorgänge für Daten mit seltenem Zugriff und 250.000 Lese-IOPS für Daten mit häufigem Zugriff durchführen. 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”](#).

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 250 µs für Lesevorgänge und 2,7 ms für Schreibvorgänge.
- Die Speicherklassen EFS Infrequent Access (IA) und EFS Archive speichern weniger häufig aufgerufene Daten, die nicht die Latenzleistung erfordern, die häufig aufgerufene Daten erfordern. 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 Operation 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.
- Der Modus Max. E/A ist ein Leistungstyp der vorherigen Generation, der für stark parallelisierte Workloads konzipiert wurde, die höhere Latenzen tolerieren können als der Allzweckmodus. Der Modus „Max. E/A“ wird von One-Zone-Dateisystemen oder Dateisystemen, die den elastischen Durchsatzmodus 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 des für Dateisysteme verfügbaren IOPS-Limits im Allzweck-Leistungsmodus bleibt, können Sie die `-PercentIOLimit` CloudWatch Metrik überwachen. Weitere Informationen finden Sie unter [Amazon- 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: Elastisch, 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 und -Limits](#).

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 [Verwenden von CloudWatch Metriken zur 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 elastischen Durchsatz, wenn Sie hohe oder unvorhersehbare Workloads und Leistungsanforderungen haben, die schwer zu prognostizieren 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 in 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 wird.

Wenn Sie nach der Verwendung des Bursting-Durchsatzes feststellen, dass Ihre Anwendung durch den Durchsatz eingeschränkt ist (z. B. mehr als 80 % des zulässigen Durchsatzes oder Sie haben alle Ihre Burst-Guthaben verwendet), sollten Sie entweder den elastischen oder den bereitgestellten Durchsatz verwenden. Weitere Informationen finden Sie unter [Bursting-Durchsatz](#).

Sie können Amazon verwenden CloudWatch , um das average-to-peak Verhältnis Ihres Workloads zu bestimmen, indem Sie die -MeteredIOBytesMetrik mit der -PermittedThroughputMetrik vergleichen. Weitere Informationen zu Amazon-EFS-Metriken finden Sie unter [Amazon- CloudWatch Metriken für Amazon EFS](#).

Elastischer Durchsatz

Für Dateisysteme, die den elastischen Durchsatz verwenden, skaliert Amazon EFS die Durchsatzleistung automatisch nach oben oder unten, um die Anforderungen Ihrer Workload-Aktivität zu erfüllen. Der elastische Durchsatz ist der beste Durchsatzmodus für hohe oder unvorhersehbare Workloads mit Leistungsanforderungen, die schwer zu prognostizieren sind, oder für Anwendungen, die den Durchsatz im Durchschnitt bei 5 % oder weniger des Spitzendurchsatzes (das average-to-peak Verhältnis) erhöhen.

Da die Durchsatzleistung für Dateisysteme mit Elastic-Durchsatz automatisch skaliert wird, müssen Sie die Durchsatzkapazität nicht angeben 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 keine Burst-Guthaben, während Sie den Elastic-Durchsatz verwenden.

 Note

Der elastische Durchsatz ist nur für Dateisysteme verfügbar, die den Allzweck-Leistungsmodus verwenden.

Informationen zu den Elastic-Durchsatzlimits pro Region finden Sie unter [Amazon-EFS-Kontingente, die Sie erhöhen können](#).

Bereitgestellter Durchsatz

Mit Bereitgestellter Durchsatz geben Sie einen Durchsatz an, den das Dateisystem unabhängig von der Größe oder dem Burst-Guthabensaldo des Dateisystems erreichen kann. Verwenden Sie den bereitgestellten Durchsatz, wenn Sie die Leistungsanforderungen Ihres Workloads kennen oder wenn Ihre Anwendung den Durchsatz bei mindestens 5 % des average-to-peak Verhältnisses steigert.

Bei Dateisystemen, die den bereitgestellten Durchsatz verwenden, wird Ihnen die für das Dateisystem aktivierte Durchsatzmenge 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 die Menge des bereitgestellten Durchsatzes überschreitet, verwendet es automatisch den für das Dateisystem zulässigen Bursting-Durchsatz (bis zu den entsprechenden \Bursting-Basisdurchsatzlimits in dieser AWS-Region).

Informationen zu RegionProvisioned den Grenzwerten pro Durchsatz finden Sie unter [Amazon-EFS-Kontingente, die Sie erhöhen können](#).

Bursting-Durchsatz

Der Bursting-Durchsatz wird für Workloads empfohlen, die einen Durchsatz erfordern, der mit der Speichermenge in Ihrem Dateisystem skaliert wird. Beim Bursting-Durchsatz ist der Basisdurchsatz proportional zur Größe des Dateisystems in der Standard-Speicherklasse 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-Guthaben verfügbar sind, kann ein Dateisystem den Durchsatz bis zu 100 MiBps pro TiB Speicher bis zum AWS-Region Limit mit mindestens 100 steigern MiBps. Wenn keine Burst-

Guthaben verfügbar sind, kann ein Dateisystem bis zu 50 MiBps pro TiB Speicher mit mindestens 1 übertragen MiBps.

Informationen zum Bursting-Durchsatz pro Region finden Sie unter [General resource quotas that cannot be changed](#).

Wissenswertes zu Amazon-EFS-Burst-Guthaben

Mit dem Bursting-Durchsatz verdient jedes Dateisystem im Laufe der Zeit Burst-Guthaben mit einer Basisrate, die durch die Größe des Dateisystems bestimmt wird, das in der EFS-Standard-speicherklasse gespeichert ist. Die Basisrate beträgt 50 MiBps pro Tebibyte [TiB] Speicher (entsprechend 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 GiB Lesedurchsatz oder 50 KiBps 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 gemessener Daten in der Speicherklasse Standard einen Basisdurchsatz von 5 MiBps. Über einen Inaktivitätszeitraum von 24 Stunden verdient das Dateisystem ein Guthaben 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 steigen ($432.000 \text{ MiB} / 100 \text{ MiBps} = 72 \text{ Minuten}$).

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	<ul style="list-style-type: none">Burst auf 300 (MiBps) schreibgeschützt für bis zu 72 Minuten pro Tag oderBurst auf 100 MiBps write-only für bis zu 72 Minuten pro Tag	<ul style="list-style-type: none">Kontinuierliches Fahren bis zu 15 MiBps schreibgeschütztKontinuierliches Hochfahren bis zu 5 MiBps Write-Only

Größe des Dateisystems	Bursting-Durchsatz	Basisdurchsatz
1 TiB gemessene r Daten im Standards peicher	<ul style="list-style-type: none"> Burst auf 300 MiBps schreibgeschützt für 12 Stunden pro Tag oder Burst auf 100 MiBps Schreibvorgänge für 12 Stunden pro Tag 	<ul style="list-style-type: none"> Kontinuierliches Fahren mit 150 MiBps schreibgeschützten Fortlaufendes 50 MiBps Write-Only-Laufwerk
10 TiB gemessene r Daten im Standards peicher	<ul style="list-style-type: none"> Burst auf 3 GiBps schreibgeschützt für 12 Stunden pro Tag oder Burst auf 1 GiBps Write-only für 12 Stunden pro Tag 	<ul style="list-style-type: none"> Fortlaufendes GiBps Laufwerk 1.5 Fortlaufendes Laufwerk 500 MiBps write-only
Im Allgemeinen größere Dateisysteme	<ul style="list-style-type: none"> Burst auf 300 MiBps schreibgeschützt pro TiB Speicher für 12 Stunden pro Tag oder Burst auf 100 MiBps Schreibvorgänge pro TiB Speicher für 12 Stunden pro Tag 	<ul style="list-style-type: none"> Fahren Sie kontinuierlich 150 MiBps schreibgeschützt pro TiB Speicher Ständiges Antreiben von 50 MiBps Schreibvorgängen pro TiB Speicher

Note

Amazon EFS bietet einen gemessenen Durchsatz von 1 MiBps zu allen Dateisystemen, 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 Guthachten bis zu einer Höhe von maximal 2,1 TiB erwerben. Dateisysteme über 1 TiB können Guthachten 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 Ändern der bereitgestellten Menge

Sie können den Durchsatzmodus eines vorhandenen Dateisystems wechseln und die Durchsatzmenge ändern. Nachdem Sie jedoch den Durchsatzmodus auf Bereitgestellter Durchsatz umgeschaltet oder die Menge des bereitgestellten Durchsatzes geändert haben, sind die folgenden Aktionen für einen Zeitraum von 24 Stunden eingeschränkt:

- Wechsel vom Modus des bereitgestellten Durchsatzes zum Modus des elastischen Durchsatzes oder des Bursting-Durchsatzes.
- Verringerung des bereitgestellten Durchsatzes.

Tipps zur Amazon EFS-Leistung

Berücksichtigen Sie bei der Verwendung von Amazon EFS die folgenden Tipps zur Leistung:

Durchschnittliche E/A-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 vorgangsbasierten Latenz wird der Gesamtdurchsatz im Allgemeinen erhöht, wenn die durchschnittliche E/A-Größe steigt, da der Overhead über eine größere Menge von Daten amortisiert wird.

Optimierung von Workloads, die einen hohen Durchsatz und IOPS erfordern

Verwenden Sie für Workloads, die einen hohen Durchsatz und IOPS erfordern, regionale Dateisysteme, die mit dem Allzweck-Leistungsmodus und dem elastischen Durchsatz konfiguriert sind.

Note

Um die maximalen 250.000 Lese-IOPS für Daten zu erreichen, auf die häufig zugegriffen wird, muss das Dateisystem den elastischen Durchsatz 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 den Workload auf 10 oder mehr NFS-Clients mit mindestens 64 Threads pro Client in einem einzigen Mountingziel.

Gleichzeitige Verbindungen

Sie können Amazon-EFS-Dateisysteme auf bis zu Tausenden von Amazon EC2- und anderen AWS Datenverarbeitungs-Instances 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 Schreibanforderungstransaktion 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 [Zusätzliche Überlegungen zum Mounting](#) 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-Mac-Instances, auf denen macOS Big Sur ausgeführt wird, wird nur NFS v4.0 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"
```

Sichern Ihrer Amazon-EFS-Dateisysteme

AWS Backup ist eine einfache und kostengünstige Möglichkeit, Ihre Daten zu schützen, indem Sie Ihre Amazon-EFS-Dateisysteme sichern. AWS Backup ist ein einheitlicher Backup-Service, der die Erstellung vereinfachen soll. -Migration, Wiederherstellung, und Löschen von Backups, bietet gleichzeitig verbesserte Berichte und Prüfungen. AWS Backup erleichtert die Entwicklung einer zentralen Backup-Strategie für die Rechtliche, regulatorische, und Professional Compliance. AWS Backup macht auch den Schutz Ihrer AWS Speicher-Volumes. -Datenbanken, - und -Dateisysteme vereinfachen die Bereitstellung eines zentralen Orts, an dem Sie Folgendes tun können:

- Konfigurieren und prüfen Sie die AWS Ressourcen, die Sie sichern möchten
- Automatisierung geplanter Sicherungen
- Festlegen von Aufbewahrungsrichtlinien
- Überwachen aller neuesten Sicherungs- und Wiederherstellungsaktivitäten

Amazon EFS ist nativ in integriert AWS Backup. Sie können die EFS-Konsole, die API und AWS Command Line Interface die (AWS CLI) verwenden, um automatische Backups für Ihr Dateisystem zu aktivieren. Automatische Backups verwenden einen Standard-Backup-Plan mit den AWS Backup empfohlenen Einstellungen für automatische Backups. Weitere Informationen finden Sie unter [Automatische Sicherungen](#). Sie können auch verwenden, AWS Backup um Ihre eigenen Backup-Pläne [manuell festzulegen](#), in denen Sie die Backup-Häufigkeit, den Zeitpunkt der Sicherung, die Aufbewahrungsdauer von Backups und eine Lebenszyklusrichtlinie für Backups angeben. Sie können diesem Sicherungsplan dann Amazon-EFS-Dateisysteme oder andere AWS -Ressourcen zuweisen.

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. Bei jedem inkrementellen Backup AWS Backup behält 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 während Ihrer Sicherung in AWS Backup auf Ihre Amazon-EFS-Dateisysteme zugreifen und diese ändern. 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.

Backup-Leistung

Im Allgemeinen können Sie die folgenden Backup- und Wiederherstellungsraten mit erwarten AWS Backup. Die Raten können für einige Workloads geringer sein, z. B. für Workloads, die eine große Datei oder ein großes Verzeichnis enthalten.

- Backup-Rate von 1 000 Dateien pro Sekunde oder 300 Megabyte pro Sekunde (MBps je nachdem, welcher Wert langsamer ist.
- Wiederherstellungsrate von 500 Dateien pro Sekunde oder 150 MBps je nachdem, welcher Wert langsamer ist.

Die maximale Dauer eines Backup-Vorgangs in AWS Backup beträgt 30 Tage.

Die Verwendung von verbraucht keine akkumulierten AWS Backup Burst-Guthaben und wird nicht auf die Dateioptionslimits für den Allzweck-Leistungsmodus angerechnet. Weitere Informationen finden Sie unter [Kontingente für Amazon-EFS-Dateisysteme](#).

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. Während des nächsten geplanten Backups setzt zu dem

Zeitpunkt AWS Backup fort, an dem es aufgehört hat. Sie können den Status all Ihrer Sicherungen in der [AWS Backup -Managementkonsole](#) anzeigen.

EFS-Speicherklassen

Sie können verwenden AWS Backup , um alle Daten in einem EFS-Dateisystem zu sichern, 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 Archiv-Speicherklasse gesichert wird.

Bei der Wiederherstellung eines Wiederherstellungspunkts werden alle Dateien in der Standardspeicherklasse wiedergestellt. Weitere Informationen zu Speicherklassen finden Sie unter [EFS-Speicherklassen](#) und [Verwaltung des Dateisystemspeichers](#).

IAM-Berechtigungen zum Erstellen und Wiederherstellen von Sicherungen

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 finden Sie unter [Identitäts- und Zugriffsmanagement für Amazon Elastic File System](#) und [Verwendung von IAM zur Steuerung des Dateisystemdatenzugriffs](#).

On-Demand-Backups

Mithilfe der [AWS Backup -Managementkonsole](#) oder der CLI können Sie eine einzelne Ressource bei Bedarf in einem Sicherungsdepot 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.

Gleichzeitige Sicherungen

AWS Backup begrenzt Backups auf ein gleichzeitiges Backup pro Ressource. Daher können geplante oder On-Demand-Sicherungen fehlschlagen, wenn bereits ein Sicherungsauftrag ausgeführt

wird. Weitere Informationen zu diesen AWS Backup -Limits finden Sie unter [AWS Backup -Limits](#) im -Entwicklerhandbuch.

Automatische Sicherungen

Wenn Sie ein Dateisystem mit der Amazon-EFS-Konsole erstellen, sind automatische Sicherungen standardmäßig aktiviert. Sie können automatische Sicherungen aktivieren, nachdem Sie Ihr Dateisystem mit der CLI oder API erstellt haben. Der Standard-EFS-Backup-Plan verwendet die AWS Backup empfohlenen Einstellungen für automatische Backups – tägliche Backups mit einem Aufbewahrungszeitraum von 35 Tagen. Die mit dem standardmäßigen EFS-Sicherungsplan erstellten Sicherungen werden in einem standardmäßigen EFS-Sicherungstresor gespeichert, der ebenfalls von EFS in Ihrem Namen erstellt wird. Der Standard-Sicherungsplan und Sicherungstresor kann normalerweise nicht gelöscht werden. Sie können die Standardeinstellungen des Backup-Plans mithilfe der AWS Backup Konsole bearbeiten. Weitere Informationen finden Sie unter [Option 3: Automatische Backups erstellen](#) im Entwicklerhandbuch für AWS Backup . Sie können alle Ihre automatischen Sicherungen anzeigen und die Standardeinstellungen für den EFS-Sicherungsplan mithilfe der [AWS Backup -Konsole](#) bearbeiten. Sie können automatische Sicherungen jederzeit über die Amazon EFS-Konsole oder CLI deaktivieren, wie im folgenden Abschnitt beschrieben.

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.

Note

Automatische Backups sind von der AWS Backup Service-Opt-Out-Konfiguration ausgenommen. Weitere Informationen finden Sie unter [Erste Schritte mit AWS Backup](#) im AWS Backup -Developerhandbuch.

Ein- oder Ausschalten automatischer Sicherungen für bestehende Dateisysteme

Nachdem Sie ein Dateisystem erstellt haben, können Sie automatische Sicherungen mithilfe der Konsole, der CLI oder der EFS-API ein- oder ausschalten.

Ein- oder Ausschalten automatischer Sicherungen für ein bestehendes Dateisystem (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.

Ein- oder Ausschalten automatischer Sicherungen für ein bestehendes Dateisystem (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"
```

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"
```

EFS reagiert mit der neuen Sicherungsrichtlinie.

```
{
  "BackupPolicy": {
    "Status": "DISABLING"
  }
}
```

Verwenden von AWS Backup zur manuellen Konfiguration von Backups

Wenn Sie verwenden AWS Backup , um Ihre Dateisystem-Backups manuell einzurichten, erstellen Sie zunächst einen Backup-Plan. Der Sicherungsplan definiert den Zeitplan, das Sicherungsfenster, die Aufbewahrungsrichtlinie, die Lebenszyklusrichtlinie und Tags. Sie können einen Backup-Plan mithilfe der [AWS Backup -Managementkonsole](#), der AWS CLI oder der AWS Backup -API erstellen. Im Rahmen eines Sicherungsplans können Sie das Folgende festlegen:

- Zeitplan – Wenn die Sicherung erfolgt
- Sicherungsfenster – Der Zeitrahmen, in dem die Sicherung gestartet werden muss
- Lebenszyklus – Wann ein Wiederherstellungspunkt in den selten genutzten Speicher verschoben werden und wann er gelöscht werden sollte
- Sicherungsdepot – Welches Depot verwendet wird, um von der Sicherungsregel erstellte Wiederherstellungspunkte zu organisieren

Nach dem Erstellen des Sicherungsplans weisen Sie die spezifischen Amazon-EFS-Dateisysteme entweder mit Tags oder der Amazon-EFS-Dateisystem-ID dem Sicherungsplan zu . Nachdem ein Plan zugewiesen ist, beginnt AWS Backup entsprechend dem von Ihnen definierten Sicherungsplan automatisch mit der Sicherung des Amazon-EFS-Dateisystems in Ihrem Namen. Sie können die AWS Backup Konsole verwenden, um Backup-Konfigurationen zu verwalten oder Backup-Aktivitäten zu überwachen. Weitere Informationen finden Sie im [AWS Backup -Entwicklerhandbuch](#).

Note

Sockets und Named Pipes werden nicht unterstützt und Sicherungen werden ausgelassen.

Wiederherstellen eines Wiederherstellungspunkts

Mithilfe der [AWS Backup -Konsole](#) oder der CLI können Sie einen Wiederherstellungspunkt in einem neuen EFS-Dateisystem oder einem bestehenden Dateisystem wiederherstellen. Sie können eine vollständige Wiederherstellung durchführen, die das gesamte Dateisystem wiederherstellt. Sie können auch bestimmte Dateien und Verzeichnisse mithilfe einer Teilwiederherstellung wiederherstellen. Um eine bestimmte Datei oder ein bestimmtes Verzeichnis wiederherzustellen, müssen Sie den relativen Pfad für den Mountingpunkt angeben. Wenn das Dateisystem beispielsweise in `/user/home/myname/efs` gemountet und der Dateipfad `„user/home/myname/efs/file1“` ist, geben Sie `„/file1“` ein. Pfade beachten die Groß- und Kleinschreibung und dürfen keine Sonderzeichen, Platzhalter und RegEx-Zeichenfolgen (Regular Expression) enthalten.

Note

Zum Wiederherstellen eines Wiederherstellungspunkts benötigen Benutzer die `backup:StartRestoreJob`-Genehmigung.

Wenn Sie entweder eine vollständige oder eine teilweise Wiederherstellung durchführen, wird der Wiederherstellungspunkt im Wiederherstellungsverzeichnis, `aws-backup-restore_timestamp-of-restore`, wiederhergestellt. Wenn die Wiederherstellung abgeschlossen ist, können Sie das Wiederherstellungsverzeichnis im Stammverzeichnis des Dateisystems sehen. Wenn Sie mehrere Wiederherstellungen für denselben Pfad versuchen, existieren möglicherweise mehrere Verzeichnisse, die die wiederhergestellten Elemente enthalten. Wenn die Wiederherstellung fehlschlägt, sehen Sie das Verzeichnis `aws-backup-failed-restore_timestamp-of-restore`. Sie müssen die Verzeichnisse `restore` und `failed-restore` manuell löschen, wenn Sie sie nicht mehr verwenden möchten.

Note

Für Teilweise Wiederherstellungen in einem vorhandenen EFS-Dateisystem stellt die Dateien und Verzeichnisse in einem neuen Verzeichnis im Stammverzeichnis des Dateisystems AWS Backup wieder her. Die vollständige Hierarchie der angegebenen Elemente bleibt im Wiederherstellungsverzeichnis erhalten. Wenn Verzeichnis A beispielsweise Unterverzeichnisse B, C und D enthält, AWS Backup behält die hierarchische Struktur bei, wenn A, B, C und D wiederhergestellt werden.

Nach dem Wiederherstellen eines Wiederherstellungspunkts werden Datenfragmente, die nicht im entsprechenden Verzeichnis wiederhergestellt werden können, im Verzeichnis `aws-backup-lost+found` abgelegt. Fragmente können in dieses Verzeichnis verschoben werden, wenn Änderungen an dem Dateisystem vorgenommen werden, während die Sicherung ausgeführt wird.

Löschen eines Backups

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.

So löschen Sie einen EFS-Wiederherstellungspunkt in AWS Backup

1. Öffnen Sie die - AWS Backup Konsole unter <https://console.aws.amazon.com/backup>.
2. Wählen Sie im Navigationsbereich Sicherungstresore aus.
3. Wählen Sie in der Liste Backup-Tresore die Option `aws/efs/automatic-backup-vault` aus.
4. Wählen Sie auf der Seite mit den Tresordetails in der oberen rechten Ecke der Seite die Option Zugriff verwalten aus. Die Seite Zugriffsrichtlinie bearbeiten wird angezeigt.
5. Um alle Aktionen im EFS-Sicherungstresor zuzulassen, suchen Sie die Zeile "Effect": "Deny", im JSON-Editor und bearbeiten Sie die Zeile, sodass sie als "Effect": "Allow", zu lesen ist.
6. Wählen Sie Richtlinie speichern aus, um Ihre Änderungen zu speichern.
7. Scrollen Sie auf der Seite mit den Tresordetails nach unten zum Abschnitt Sicherungen und wählen Sie die Wiederherstellungspunkte aus, die Sie aus der Liste der Sicherungen löschen möchten. Wählen Sie dann Aktionen und anschließend Löschen aus.
8. Folgen Sie den Anweisungen, um das Löschen zu bestätigen. Wählen Sie dann Wiederherstellungspunkte löschen aus.

Replizieren von Dateisystemen

Sie können ein Replikat Ihres EFS-Dateisystems in der von Ihnen bevorzugten AWS-Region erstellen. Wenn Sie die Replikation auf einem EFS-Dateisystem aktivieren, repliziert Amazon Elastic File System (Amazon EFS) die Daten und Metadaten im Quelldateisystem automatisch und transparent in ein Zieldateisystem. Im Katastrophenfall oder bei der Durchführung von Gameday-Übungen können Sie ein Failover auf Ihr Replikat-Dateisystem und dann auf das primäre Dateisystem zurückgreifen, um den Betrieb wieder aufzunehmen. Um den Prozess der Erstellung des Zieldateisystems und dessen Synchronisation mit dem Quelldateisystem zu verwalten, verwendet Amazon EFS eine Replikationskonfiguration. Weitere Informationen zum Erstellen einer Replikationskonfiguration für ein Dateisystem finden Sie unter [Replikationskonfiguration](#).

Nachdem eine Replikationskonfiguration für ein Dateisystem erstellt wurde, synchronisiert Amazon EFS die Quell- und Zieldateisysteme automatisch. Am Quelldateisystem vorgenommene Änderungen werden nicht auf point-in-time konsistente Weise in das Zieldateisystem übertragen, sondern auf der Grundlage der zuletzt synchronisierten Zeit 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 Quelldateisystem vorgenommen wurden, werden in das Zieldateisystem repliziert, während Änderungen, die nach der letzten Synchronisierung am Quelldateisystem vorgenommen wurden, möglicherweise nicht repliziert werden. Weitere Informationen finden Sie unter [Überwachung des Replikationsstatus](#).

Die Replikation ist in allen AWS-Regionen verfügbar, in denen EFS verfügbar ist. Um die Replikation in einer Region verwenden zu können, die standardmäßig deaktiviert ist, müssen Sie sich zunächst für die Region anmelden. Weitere Informationen finden Sie unter [Verwalten von AWS-Regionen](#) in der Allgemeinen Referenz zu AWS. Wenn Sie sich später von einer Region abmelden, unterbricht Amazon EFS alle Replikationsaktivitäten für die Region. Um die Replikationsaktivitäten für die Region wieder aufzunehmen, müssen Sie sich erneut für die AWS-Region anmelden.

Note

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

- [Replikationskonfiguration](#)
- [Erstellen der Replikationskonfiguration](#)
- [Anzeigen von Replikationskonfigurationen](#)
- [Löschen von Replikationskonfigurationen](#)
- [Überwachung des Replikationsstatus](#)

Replikationskonfiguration

Wenn Sie die Replikationskonfiguration für Ihr Dateisystem erstellen, wählen Sie die AWS-Region aus, in der die Replikation erstellt werden soll, und ob auf ein neues oder vorhandenes Zieldateisystem repliziert werden soll.

Note

Ein Dateisystem kann nur Teil einer Replikationskonfiguration sein. Das Quelldateisystem kann in einer anderen Replikationskonfiguration kein Zieldateisystem sein.

Replizieren in ein neues Dateisystem

Amazon EFS erstellt automatisch ein neues Dateisystem und kopiert die Daten und Metadaten auf dem Quelldateisystem in ein neues schreibgeschütztes Zieldateisystem in der von Ihnen ausgewählten AWS-Region. Das Zieldateisystem wird mit den folgenden Eigenschaften erstellt:

- **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 Schlüssel AWS Key Management Service (AWS KMS) 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 Zielsystem erstellt wurde.

- **Automatische Sicherungen** – Für Zielsysteme, 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 [Automatische Sicherungen](#).
- **Leistungsmodus** – Der Leistungsmodus des Zielsystems entspricht dem des Quelldateisystems, es sei denn, das Zielsystem verwendet One Zone-Speicher. In diesem Fall wird der Allzweck-Leistungsmodus verwendet. Der Leistungsmodus kann nicht geändert werden.
- **Durchsatzmodus** – Der Durchsatzmodus des Zielsystems entspricht dem des Quelldateisystems. Nachdem das Dateisystem erstellt wurde, können Sie den Modus ändern.

Wenn der Durchsatzmodus des Quelldateisystems Bereitgestellt ist, entspricht der bereitgestellte Durchsatz des Zielsystems dem des Quelldateisystems, es sei denn, der bereitgestellte Betrag der Quelldatei überschreitet das Limit für die Region des Zielsystems. Wenn die vom Quelldateisystem bereitgestellte Menge das Limit der Region für das Zielsystem überschreitet, entspricht die bereitgestellte Durchsatzmenge des Zielsystems dem Limit der Region. Weitere Informationen finden Sie unter [Amazon-EFS-Kontingente, die Sie erhöhen können](#).

- **Lebenszyklusverwaltung** – Das Lebenszyklusmanagement ist auf dem Zielsystem nicht aktiviert. Nachdem das Zielsystem erstellt wurde, können Sie es aktivieren. Weitere Informationen finden Sie unter [Verwaltung des Dateisystemspeichers](#).

Replizieren in ein vorhandenes Dateisystem

EFS repliziert die Daten und Metadaten auf dem Quelldateisystem in das Zielsystem und die AWS-Region, das Sie auswählen. Während der Replikation identifiziert EFS Datenunterschiede zwischen den Dateisystemen und wendet die Unterschiede auf das Zielsystem an.

Bei der Replikation in ein vorhandenes Dateisystem gelten folgende Anforderungen.

- Der Replikationsüberschreibschutz des Zieldateisystems muss deaktiviert werden. Der Replikationsüberschreibschutz verhindert, dass das Dateisystem als Ziel in einer Replikationskonfiguration verwendet wird. Weitere Informationen zum Deaktivieren des Schutzes finden Sie unter [Schutz des Dateisystems](#).

Das Deaktivieren des Replikationsüberschreibschutzes erfordert Berechtigungen für `elasticfilesystem:UpdateFileSystemProtection` action. Weitere Informationen finden Sie unter [AWSverwaltete Richtlinie: AmazonElasticFileSystemFullAccess](#).

- 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](#).

Schutz des Dateisystems

Wenn Sie ein Amazon-EFS-Dateisystem erstellen, ist der 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.

Für die Deaktivierung des Replikationsüberschreibschutzes sind Berechtigungen für die Aktion `elasticfilesystem:UpdateFileSystemProtection` erforderlich. Weitere Informationen finden Sie unter [AWSverwaltete Richtlinie: AmazonElasticFileSystemFullAccess](#).

Der Status des Replikationsüberschreibschutzes für ein Amazon-EFS-Dateisystem kann einen der in der folgenden Tabelle beschriebenen Statuswerte haben.

Status des Dateisystems	Beschreibung
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.

Status des Dateisystems	Beschreibung
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.

So deaktivieren Sie den Replikationsüberschreibschutz (Konsole)

1. Melden Sie sich bei der AWS Management Console an 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.

Erforderliche Berechtigungen

Amazon EFS verwendet die mit dem EFS Service verknüpfte Rolle namens `AWSServiceRoleForAmazonElasticFileSystem`, um den Status der Replikation zwischen den Quell- und Zieldateisystemen zu synchronisieren. Um die EFS-Replikation verwenden zu können, müssen Sie die folgenden Berechtigungen konfigurieren, damit eine IAM-Entität (z. B. Benutzer, Gruppe oder Rolle) eine servicegebundene Rolle, eine Replikationskonfiguration und ein Dateisystem erstellen kann.

- `elasticfilesystem:CreateReplicationConfiguration*`
- `elasticfilesystem>DeleteReplicationConfiguration*`
- `elasticfilesystem:DescribeFileSystem`
- `elasticfilesystem:DescribeReplicationConfigurations*`
- `elasticfilesystem>CreateFileSystem*`

- `iam:CreateServiceLinkedRole` – siehe das Beispiel in [Verwendung von serviceverknüpften Rollen für Amazon EFS](#).

Note

* Sie können stattdessen die verwaltete `AmazonElasticFileSystemFullAccess`-Richtlinie verwenden, um automatisch alle erforderlichen EFS-Berechtigungen zu erhalten. Weitere Informationen finden Sie unter [AWSverwaltete Richtlinie: AmazonElasticFileSystemFullAccess](#).

Kosten

Um die Replikation zu erleichtern, erstellt Amazon EFS versteckte Verzeichnisse und Metadaten im Zielsystem. Dies entspricht etwa 12 MiB an gemessenen Daten, die Ihnen in Rechnung gestellt werden. Weitere Informationen über die Ermittlung des Dateisystemspeichers finden Sie unter [Messung: Wie Amazon EFS die Größe von Dateisystemen und Objekten meldet](#).

Leistung

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. 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 [Überwachung des Replikationsstatus](#).

Sie können mit der Konsole, der AWS Command Line Interface (AWS CLI), der API und Amazon überwachen, wann die letzte erfolgreiche Synchronisierung stattgefunden hat CloudWatch. Verwenden Sie CloudWatch in die [TimeSinceLastSync](#) EFS-Metrik. Weitere Informationen finden Sie unter [Überwachung des Replikationsstatus](#).

Mounten eines Zielsystems

Amazon EFS erstellt keine Mount-Ziele, wenn es das Zielsystem erstellt. Um ein Zielsystem zu mounten, müssen Sie ein oder mehrere Mount-Ziele erstellen. Weitere Informationen finden Sie unter [Verwenden der EFS-Mountinghilfe zum Mounten von EFS-Dateisystemen](#).

Da ein Zielsystem schreibgeschützt ist, solange es Mitglied einer Replikationskonfiguration ist, schlagen alle Schreibvorgänge in diesem System fehl. Sie können das Zielsystem jedoch für schreibgeschützte Anwendungsfälle wie Tests und Entwicklung verwenden.

Failover und Failback des Dateisystems

Im Notfall oder bei der Durchführung von Gameday-Übungen können Sie ein Failover auf Ihr Replikatdateisystem durchführen, indem Sie die Replikationskonfiguration löschen. Nachdem die Replikationskonfiguration gelöscht wurde, ist das Replikat schreibbar und Sie können es in Ihrem Anwendungsworkflow verwenden. Wenn der Notfall behoben ist oder die Gameday-Übung 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 [Leistung](#).

Erstellen der Replikationskonfiguration

Sie können die Amazon-EFS-Konsole, die API oder die AWS CLI verwenden, um ein EFS-Dateisystem zu replizieren. In den folgenden Abschnitten finden Sie detaillierte Anweisungen zur Verwendung der einzelnen Methoden.

Gehen Sie wie folgt vor, um eine Replikationskonfiguration (Konsole) zu erstellen:

1. Melden Sie sich bei der AWS Management Console an 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 und dann im Abschnitt Replikation die Option Replikation erstellen aus. Die Seite Replikation erstellen wird geöffnet.
4. Definieren Sie im Abschnitt Replikationseinstellungen die Replikationseinstellungen:
 - a. Wählen Sie für die Replikationskonfiguration aus, ob das Dateisystem auf ein neues oder ein vorhandenes Dateisystem repliziert werden soll.
 - b. Wählen Sie für Ziel-AWS-Region die AWS-Region aus, in die das Dateisystem repliziert werden soll.
5. Wenn Sie ein neues Zieldateisystem replizieren, definieren Sie im Abschnitt Einstellungen für das Zieldateisystem die Einstellungen des Zieldateisystems.
 - a. Wählen Sie unter Dateisystemtyp eine Speicheroption für das Dateisystem aus.
 - Wählen Sie Regional aus, um ein Dateisystem zu erstellen, das Daten redundant in mehreren geografisch getrennten Availability Zones innerhalb einer AWS-Region speichert.

- Wählen Sie One Zone und anschließend die Availability Zone aus, um ein Dateisystem zu erstellen, das Daten redundant innerhalb einer einzelnen Availability Zone in einer AWS-Region speichert.

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. Amazon EFS verwendet standardmäßig Ihren AWS Key Management Service (AWS KMS)-Serviceschlüssel für Amazon EFS (`aws/elasticfilesystem`). Um einen anderen KMS-Schlüssel zu verwenden, wählen Sie einen KMS-Schlüssel oder geben Sie den ARN für einen vorhandenen Schlüssel ein.

 Important

Der KMS-Schlüssel kann nicht geändert werden, nachdem das Dateisystem erstellt wurde.

6. Wenn Sie in ein vorhandenes Zieldateisystem replizieren, wählen Sie EFS durchsuchen aus und dann das Dateisystem. Der Pfad zu Ihrem Zieldateisystem wird im Feld Ziel angezeigt.

Wenn der Replikationsüberschreibschutz für das Dateisystem aktiviert ist, wird eine Warnung angezeigt, in der Sie aufgefordert werden, den Schutz zu deaktivieren. Um den Schutz zu deaktivieren, wählen Sie Schutz deaktivieren aus und schalten Sie dann den Replikationsüberschreibschutz aus. Nachdem Sie den Schutz deaktiviert haben, klicken Sie auf die Schaltfläche Aktualisieren, um die Meldung zu löschen.

7. Wählen Sie Replikation erstellen. Wenn Sie auf ein neues Dateisystem replizieren, wird eine Meldung angezeigt, in der Sie aufgefordert werden, die Replikation zu bestätigen. Geben Sie Bestätigen in das Eingabefeld ein und klicken Sie dann auf Replikation erstellen.

Der Abschnitt Replikation wird mit den Replikationsdetails angezeigt. Der Wert für den Replikationsstatus lautet anfänglich Aktiviert und Letzte Synchronisierung ist leer. Wenn der Status Aktiviert lautet, zeigt Letzte Synchronisierung den Wert Anfängliche Synchronisierung läuft.

8. Um die Konfigurationsinformationen des Zielsystems zu sehen, wählen Sie die Dateisystem-ID über dem Zielsystem aus. Die Seite mit den Dateisystemdetails für das Zielsystem wird in einer neuen Browser-Registerkarte angezeigt (abhängig von Ihren Browsereinstellungen).

Gehen Sie wie folgt vor, um eine Replikationskonfiguration (CLI) zu erstellen:

Um die Replikationskonfiguration abzurufen, verwenden Sie den `create-replication-configuration`-CLI-Befehl. 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. In diesem Beispiel wird der `Region`-Parameter verwendet, um ein Zielsystem in der `eu-west-2` AWS-Region zu erstellen. Der `KmsKeyId`-Parameter gibt die KMS-Schlüssel-ID zum Verschlüsseln des Zielsystems an.

```
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 diesem 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 des Kontos für Amazon EFS (aws/elasticfilesystem) verschlüsselt.

```
aws efs create-replication-configuration \  
--source-file-system-id fs-0123456789abcdef1 \  
--destinations AvailabilityZoneName=us-west-2a
```

Anzeigen von Replikationskonfigurationen

Um die Replikationskonfiguration eines Dateisystems anzuzeigen, können Sie die Amazon EFS-Konsole oder die AWS CLI verwenden.

Gehen Sie wie folgt vor, um eine Replikationskonfiguration (Konsole) anzuzeigen:

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.

Der Status Angehalten entsteht, wenn Sie sich nach der Erstellung der Replikationskonfiguration aus der Quell- oder Zielregion abmelden. Um die Replikation für das Dateisystem wieder aufzunehmen, müssen Sie sich erneut für die AWS-Region. Weitere Informationen finden Sie unter [Verwalten von AWS-Regionen](#) im Allgemeinen Referenzhandbuch zu AWS.

Der Status Replizieren tritt ein, nachdem eine Replikation erstellt wurde, wobei das Dateisystem entweder das Quell- oder das Zieldateisystem ist.

Der Status Fehler tritt ein, wenn sich entweder das Quell- oder das Zielsystem (oder beide) in einem ausgefallenen Zustand befinden und nicht wiederhergestellt werden können. Weitere Informationen finden Sie unter [Überwachung des Replikationsstatus](#). Zur Wiederherstellung müssen Sie die Replikationskonfiguration löschen und dann die letzte Sicherung des ausgefallenen Dateisystems (entweder das Quell- oder das Ziel) in einem neuen Dateisystem wiederherstellen.

- 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 Zielsystem stattgefunden hat. Alle Änderungen an Daten im Quelldateisystem, 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.
- Replikationsdateisysteme listet jedes Dateisystem in der Replikationskonfiguration nach seiner Dateisystem-ID, der Rolle, die es in der Replikationskonfiguration spielt (entweder Quelle oder Ziel), der AWS-Region, in der es sich befindet, und seiner Berechtigung auf. Ein Quelldateisystem hat die Berechtigung Schreibbar und ein Zielsystem hat die Berechtigung Schreibgeschützt.

Gehen Sie wie folgt vor, um eine Replikationskonfiguration (CLI) anzuzeigen:

Um die Replikationskonfiguration anzuzeigen, verwenden Sie den `describe-replication-configurations`-CLI-Befehl. Sie können die Replikationskonfiguration entweder für ein bestimmtes Dateisystem oder alle Replikationskonfigurationen für eine bestimmte AWS-Konto in einer AWS-Region anzeigen. Der äquivalente API-Befehl lautet [DescribeReplicationConfigurations](#).

Verwenden Sie den `file-system-id`-URI-Anforderungsparameter, um die Replikationskonfiguration für ein Dateisystem anzuzeigen. Sie können die ID eines Quell- oder eines Zielsystems angeben.

```
aws efs describe-replication-configurations --file-system-id fs-0123456789abcdef1
```

```
{
  "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"
      }
    ]
  }
]
}

```

Um alle Replikationskonfigurationen für ein Konto in einer AWS-Region anzuzeigen, geben Sie den `file-system-id`-Parameter nicht an.

```
aws efs describe-replication-configurations
```

```

{
  "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, wird das Zielsystem beschreibbar und der Replikationsüberschreibschutz wird erneut aktiviert. Weitere Informationen finden Sie unter [Failover und Failback des Dateisystems](#).

Das Löschen einer Replikationskonfiguration und das Ändern des Zielsystems, 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 Zielsystems 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 Zielsystem löschen, indem Sie die Konsole, die CLI oder die API verwenden.

Gehen Sie wie folgt vor, um eine Replikationskonfiguration (Konsole) zu löschen:

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 Zielsystem 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.

Gehen Sie wie folgt vor, um eine Replikationskonfiguration (CLI) zu löschen:

Um eine Replikationskonfiguration zu löschen, verwenden Sie den `delete-replication-configuration`-CLI-Befehl. Der äquivalente API-Befehl lautet [DeleteReplicationConfiguration](#).

Verwenden Sie den `source-file-system-id`-Parameter, um anzugeben, welche Replikationskonfiguration Sie löschen.

```
aws efs --region us-west-2 delete-replication-configuration \
--source-file-system-id fs-0123456789abcdef1
```

Überwachung des Replikationsstatus

In einer Replikationskonfiguration können Sie den Zeitpunkt überwachen, zu dem die letzte erfolgreiche Synchronisierung 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. Um zu überwachen, wann die letzte Replikation erfolgreich abgeschlossen wurde, können Sie die Konsole, CLI, API oder Amazon verwenden CloudWatch.

- In der Konsole – Die Eigenschaft Zuletzt synchronisiert im Abschnitt Dateisystemdetails > Replikation zeigt den Zeitpunkt an, zu dem die letzte erfolgreiche Synchronisierung zwischen der Quelle und dem Ziel abgeschlossen wurde.

- In der CLI oder API – 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 an, die seit Abschluss der letzten erfolgreichen Synchronisierung verstrichen ist. Weitere Informationen finden Sie unter [Amazon- CloudWatch Metriken für Amazon EFS](#).

Sie können den Status einer Replikationskonfiguration auch mithilfe der Konsole, CLI oder API überwachen. Eine Replikationskonfiguration kann einen der in der folgenden Tabelle beschriebenen Statuswerte haben.

Replikationsstatus	Beschreibung
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 unterbricht gerade die Replikation, da die Region für eines oder beide Dateisysteme in der Replikationskonfiguration deaktiviert wurde.
PAUSED	Die Replikation wird angehalten, da die Region für eines oder beide Dateisysteme in der Replikationskonfiguration deaktiviert wurde. Um die Replikation fortzusetzen, müssen Sie sich erneut für die AWS-Region anmelden. Weitere Informationen finden Sie unter Verwalten von AWS-Regionen im Allgemeinen Referenzhandbuch zu AWS.
ERROR	Eines (oder beide) der Dateisysteme in der Replikationskonfiguration ist ausgefallen und kann nicht wiederhergestellt werden. Um auf die Dateisystemdaten zuzugreifen, stellen Sie eine Sicherungskopie dieses fehlgeschlagenen Dateisystems in einem neuen Dateisystem wieder her. Weitere

Replikationsstatus	Beschreibung
	Informationen finden Sie unter Wiederherstellen eines Wiederherstellungspunkts .

Anleitungen für Amazon Elastic File System

Dieser Abschnitt enthält Anleitungen, mit Hilfe Sie Amazon EFS näher kennenlernen und die durchgehende Einrichtung testen können.

Themen

- [Exemplarische Anleitung: Erstellen eines Amazon EFS-Dateisystems und das Mounten auf einer Amazon EC2 EC2-Instance mithilfe der AWS CLI](#)
- [Komplettlösung: Einrichten eines Apache-Web-Servers und Bereitstellen von Amazon EFS-Dateien](#)
- [Anleitung: Erstellen Sie beschreibbare Unterverzeichnisse für einzelne Benutzer und konfigurieren Sie das automatische erneute Mounting bei Neustarts](#)
- [Exemplarische Vorgehensweise: Erstellen und Bereitstellen eines lokalen Dateisystems mit VPN AWS Direct Connect](#)
- [Exemplarische Vorgehensweise: Mounten eines Dateisystems aus einer anderen VPC](#)
- [Exemplarische Anleitung: Erzwingen der Verschlüsselung auf einem Amazon EFS-Dateisystem im Ruhezustand](#)
- [Exemplarische Vorgehensweise: Root-Squashing mithilfe der IAM-Autorisierung für NFS-Clients aktivieren](#)

Exemplarische Anleitung: Erstellen eines Amazon EFS-Dateisystems und das Mounten auf einer Amazon EC2 EC2-Instance mithilfe der AWS CLI

In dieser schrittweisen Anleitung wird die AWS CLI um die Amazon EFS-API zu erkunden. In dieser Anleitung erstellen Sie ein verschlüsseltes Amazon EFS-Dateisystem, mounten es auf einer Amazon EC2 EC2-Instance in Ihrer VPC und testen die Einrichtung.

Note

Diese Anleitung ähnelt der Übung „Erste Schritte“. In der [Erste Schritte](#) Übung verwenden Sie die Konsole, um EC2- und Amazon EFS-Ressourcen zu erstellen. In dieser schrittweisen Anleitung verwenden Sie die AWS CLI. Dies dient hauptsächlich dazu, sich mit der Amazon EFS-API vertraut zu machen.

In dieser schrittweisen Anleitung erstellen Sie wie folgt AWS-Ressourcen in Ihrem Konto:

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

Sie fügen diesen Sicherheitsgruppen Regeln hinzu, um entsprechenden ein-/ausgehenden Zugriff zu genehmigen. Damit kann Ihre EC2-Instance mithilfe eines NFSv4.1 TCP-Standardports über das Mounting-Ziel eine Verbindung mit dem Dateisystem herstellen.

- Eine Amazon 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 [Amazon EFS – Funktionsweise](#).

Anschließend testen Sie das Dateisystem auf Ihrer EC2-Instance. Der Schritt für die Bereinigung am Ende der Anleitung stellt Informationen zum Entfernen dieser Ressourcen bereit.

In dieser schrittweisen Anleitung werden alle Ressourcen in der Region USA West (Oregon) erstellt (us-west-2) enthalten. Welcher auch immer AWS-Region Sie verwenden, stellen sicher, dass Sie es konsistent verwenden. Alle Ihre Ressourcen — Ihre VPC-, EC2-Ressourcen und Amazon EFS-Ressourcen — müssen dieselbe sein AWS-Region aus.

Bevor Sie beginnen

- Sie können die Root-Anmeldeinformationen Ihres AWS-Kontos um sich bei der Konsole anzumelden und die Übung „Erste Schritte“ zu absolvieren. Allerdings empfiehlt AWS Identity and Access Management (IAM), dass Sie nicht die Root-Anmeldeinformationen Ihres AWS-Kontos aus. 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 [Einrichten](#).
- 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 [Aktualisieren der DNS-Unterstützung für Ihre VPC](#) im Amazon-VPC-Benutzerhandbuch.
- Das Internet-Gateway ist mit Ihrer VPC verbunden. Weitere Informationen finden Sie unter [Internet-Gateways](#) 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 Addressing in Your VPC \(IP-Adresszuweisung in Ihrem VPC\)](#) in Amazon VPC Benutzerhandbuch.
- Die VPC-Routing-Tabelle enthält eine Regel zum Senden des gesamten Internet-Datenverkehrs an das Internet-Gateway.
- Sie müssen die AWS CLI einrichten und die adminuser-Profil hinzufügen.

Einrichten von AWS CLI

Befolgen Sie die folgenden Anleitungen, um die AWS CLI und Benutzerprofile einzurichten.

Um das AWS CLI einzurichten

1. Herunterladen und Konfigurieren von AWS CLI. Eine Anleitung finden Sie unter den folgenden Themen im AWS Command Line Interface-Benutzerhandbuch.

[Einrichten mit der AWS-Befehlszeilenschnittstelle](#)

[Installieren der AWS-Befehlszeilenschnittstelle](#)

[Konfigurieren der AWS-Befehlszeilenschnittstelle](#)

2. Richten Sie Profile ein.

Sie speichern Ihre Anmeldeinformationen in der AWS CLI-Datei config. Die CLI-Beispielbefehle in dieser Anleitung geben das adminuser-Profil an. 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 obige Profil legt auch den Standardwert fest AWS-Region aus. Wenn Sie im CLI-Befehl keine Region angeben, wird von der Region us-west-2 ausgegangen.

- 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
```

Nächster Schritt

[Schritt 1: Erstellen Amazon EC2 EC2-Ressourcen](#)

Schritt 1: Erstellen Amazon EC2 EC2-Ressourcen

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 Sie ein Amazon EFS-Dateisystem und stellen es auf dieser Instance bereit.

Themen

- [Schritt 1.1: Erstellen von zwei Sicherheitsgruppen](#)

- [Schritt 1.2: Hinzufügen von Regeln zu den Sicherheitsgruppen, um ein-/ausgehendem Zugriff zu autorisieren](#)
- [Schritt 1.3: Starten einer EC2-Instance](#)

Schritt 1.1: Erstellen von zwei Sicherheitsgruppen

In diesem Abschnitt erstellen Sie in Ihrer VPC Sicherheitsgruppen für Ihre EC2-Instance und Ihr Amazon EFS Mountingziel. Später in dieser Anleitung weisen Sie diese Sicherheitsgruppen einer EC2-Instance und einem Amazon EFS -Mountingziel zu. Weitere Informationen zu -Sicherheitsgruppen finden Sie unter [Sicherheitsgruppen für EC2-VPC](#) im Amazon EC2-Benutzerhandbuch für Linux-Instances aus.

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 einer Sicherheitsgruppe (`efs-walkthrough1-mt-sg`) für Ihr Amazon EFS Mount-Target. 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 -Mountingziel von Amazon EFS (dem Sie diese Sicherheitsgruppen zu einem späteren Zeitpunkt in dieser Anleitung zuweisen).

Schritt 1.2: Hinzufügen von Regeln zu den Sicherheitsgruppen, um ein-/ausgehendem Zugriff zu autorisieren

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 Zugriff auf eingehenden Datenverkehr auf die Sicherheitsgruppe für das Amazon EFS-Mounting-Ziel (efs-walkthrough1-mt-sg) enthalten.

Führen Sie an der Eingabeaufforderung den AWS CLI-Befehl `authorize-security-group-ingress` aus. Verwenden Sie hierbei das `adminuser`-Profil, 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:
 - Schlüsselpaarname:
 - Einführende Informationen finden Sie unter [Einrichten von Amazon EC2](#) im Amazon EC2-Benutzerhandbuch für Linux-Instances aus.
 - Anweisungen zum Erstellen einer .pem-Datei finden Sie unter [Erstellen eines Schlüsselpaares](#) im Amazon EC2-Benutzerhandbuch für Linux-Instances aus.
 - Die ID des Amazon-Systemabbilds (AMI), das Sie starten möchten.

Für den zum Starten einer EC2-Instance verwendeten AWS CLI-Befehl ist die ID der AMI, die Sie bereitstellen möchten, als Parameter erforderlich. In dieser Übung wird das Amazon Linux HVM AMI verwendet.

Note

Sie können die meisten allgemeinen Linux-basierten AMIs 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 die Amazon Linux HVM-AMI finden Sie die neuesten IDs unter [Amazon Linux-AMI](#). Sie wählen den ID-Wert wie folgt aus der Amazon Linux AMI-IDs-Tabelle aus:

- Wählen Sie die Region US West Oregon (USA West (Oregon)) aus. In dieser schrittweisen 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-Regionaus. 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 zur Erstellung des DNS-Namens Ihres Mountingziels, den Sie zum Mounting eines Dateisystems auf der EC2-Instance verwenden.
2. Führen Sie den AWS CLI-Befehl `run-instances` 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 [Bevor Sie beginnen](#).

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. Sie erreichen dies mit dem folgenden AWS CLI `create-tags`-Befehl.

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

Nächster Schritt

[Schritt 2: Erstellen von Amazon EFS-Ressourcen](#)

Schritt 2: Erstellen von Amazon EFS-Ressourcen

In diesem Schritt führen Sie folgende Aufgaben aus:

- Erstellen eines verschlüsselten Amazon EFS-Dateisystems.
- Aktivieren Sie die Lebenszyklusverwaltung.
- Erstellen Sie ein Mounting-Ziel in der Availability Zone, in der Sie Ihre EC2-Instance gestartet haben.

Themen

- [Schritt 2.1: Erstellen eines Amazon EFS-Dateisystems](#)
- [Schritt 2.2: Aktivieren des Lebenszyklusmanagements](#)
- [Schritt 2.3: Erstellen Sie ein Mount-Ziel](#)

Schritt 2.1: Erstellen eines Amazon EFS-Dateisystems

In diesem Schritt erstellen Sie ein Amazon 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 Folgendes aus `AWSCLI create-file-system`befehl.

```
$ 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 Mount-Ziel](#) ein Mounting-Ziel für dieses Dateisystem erstellen.

Schritt 2.2: Aktivieren des Lebenszyklusmanagements

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

So aktivieren Sie die Lebenszyklusverwaltung

- Führen Sie an der Eingabeaufforderung 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 Mount-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.
- ID des VPC-Subnetzes, in dem Sie die EC2-Instance in [Schritt 1](#) gestartet haben.

In dieser Anleitung erstellen Sie das Mounting-Ziel in demselben Subnetz, in dem Sie die EC2-Instance gestartet haben, Sie benötigen daher die Subnetz-ID (z. B. `subnet-example`).

- ID der Sicherheitsgruppe, die Sie im vorhergehenden Schritt für das Mounting-Ziel erstellt haben.

2. Führen Sie an der Eingabeaufforderung den folgenden AWS CLI-`create-mount-target`-Befehl aus.

```
$ 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
```

Nächster Schritt

[Schritt 3: Mounten Sie das Dateisystem auf der EC2-Instance](#)

Schritt 3: Mounten Sie das Dateisystem auf der EC2-Instance

In diesem Schritt führen Sie folgende Aufgaben aus:

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](#)

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 erfordert nicht, dass Ihre Amazon EC2 EC2-Instance 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 mit Ihrer EC2-Instance her:
 - Um eine Verbindung mit Ihrer Instance von einem Computer unter macOS oder Linux aus herzustellen, geben Sie die PEM-Datei für Ihren SSH-Befehl mit der Option `-i` und dem Pfad zu Ihrem privaten Schlüssel an.
 - Für die Herstellung einer Verbindung mit Ihrer Instance von einem Computer mit Windows aus können Sie MindTerm oder PuTTY verwenden. Wenn Sie PuTTY verwenden möchten, müssen Sie dies installieren und die `.pem`-Datei auf die folgende Weise zu einer `.ppk`-Datei konvertieren.

Weitere Informationen finden Sie unter den folgenden Themen im Amazon EC2-Benutzerhandbuch für Linux-Instances:

- [Herstellung einer Verbindung zu Ihrer Linux-Instance von Windows mit PuTTY](#)
- [Herstellen einer Verbindung mit Ihrer Linux-Instance per SSH](#)

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 die Amazon Linux AMI 2016.03.0 Amazon Linux AMI Wenn Sie Ihre Amazon EC2 EC2-Instance starten, müssen Sie nicht installieren `nfs-utils` weil es standardmäßig bereits im AMI enthalten ist.

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

Jetzt mounten Sie das Dateisystem auf Ihrer EC2-Instance.

1. Erstellen Sie ein Verzeichnis („EFS-Mountingpunkt“).

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

2. Mounten Sie das Amazon EFS-Dateisystem.

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

3. Nachdem Sie das Amazon EFS-Dateisystem auf Ihrer EC2-Instance gemountet 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 ein Amazon EFS-Dateisystem erfolgreich erstellt und auf Ihrer EC2-Instance in Ihrer VPC gemountet.

Das Dateisystem, das Sie gemountet haben, bleibt bei Neustarts nicht erhalten. Für ein automatisches erneutes Mounting des Verzeichnisses können Sie die Datei `fstab` verwenden. Weitere Informationen finden Sie unter [Automatische Remounting beim Neustart](#). Wenn Sie eine

Auto Scaling-Gruppe zum Starten von EC2-Instances verwenden, können Sie auch Skripts in einer Startkonfiguration einrichten. Ein Beispiel finden Sie unter [Komplettlösung: Einrichten eines Apache-Web-Servers und Bereitstellen von Amazon EFS-Dateien](#).

Nächster Schritt

[Schritt 4: Bereinigen](#)

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 Amazon EFS-Ressourcen (Dateisystem, Mounting-Ziel).

Um zu löschenAWSIn dieser schrittweisen Anleitung erstellte Ressourcen

1. Beenden Sie die EC2-Instance, die Sie für diese Übung erstellt haben.

```
$ aws ec2 terminate-instances \
--instance-ids instance-id \
--profile adminuser
```

Sie können EC2-Ressourcen auch über die Konsole löschen. Detaillierte Anweisungen finden Sie unter [Beenden einer Instance](#) im Amazon EC2-Benutzerhandbuch für Linux-Instances aus.

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.

Detaillierte Anweisungen finden Sie unter [Löschen einer Sicherheitsgruppe](#) im Amazon EC2-Benutzerhandbuch für Linux-Instances aus.

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
```

Komplettlösung: Einrichten eines Apache-Web-Servers und Bereitstellen von Amazon EFS-Dateien

Sie können EC2-Instances haben, auf denen der Apache-Webserver ausgeführt wird, der Dateien bereitstellt, die in Ihrem Amazon EFS-Dateisystem gespeichert sind. Es kann sich um eine EC2-Instance handeln, oder wenn Ihre Anwendung dies erfordert, können Sie mehrere EC2-Instances

haben, die Dateien aus Ihrem Amazon EFS-Dateisystem bereitstellen. Die folgenden Verfahren werden beschrieben.

- [Einrichten eines Apache-Webserver auf einer EC2-Instance.](#)
- [Einrichten eines Apache-Webserver auf mehreren EC2-Instances durch Erstellen einer Auto Scaling-Gruppe.](#) Sie können mehrere EC2-Instances mit Amazon EC2 Auto Scaling erstellen, einem AWS Service, mit dem Sie die Anzahl der EC2-Instances in einer Gruppe entsprechend Ihren Anwendungsanforderungen erhöhen oder verringern können. Wenn Sie mehrere Webserver ausführen, benötigen Sie auch einen Load Balancer, um den Anforderungsdatenverkehr unter ihnen aufzuteilen.

 Note

Für beide Verfahren erstellen Sie alle Ressourcen in der Region USA West (Oregon) (Oregonus-west-2).

Einzelne EC2-Instance, die Dateien bereitstellt

Folgen Sie den Schritten, um einen Apache-Webserver auf einer EC2-Instance einzurichten, um Dateien bereitzustellen, die Sie in Ihrem Amazon EFS-Dateisystem erstellen.

1. Befolgen Sie die Schritte in der Übung „Erste Schritte“, sodass Sie über eine funktionierende Konfiguration verfügen, die aus folgenden Komponenten besteht:
 - Amazon EFS-Dateisystem
 - EC2-Instance
 - Dateisystem, das auf der EC2-Instance gemountet ist

Detaillierte Anweisungen finden Sie unter [Erste Schritte mit Amazon Elastic File System](#).

Notieren Sie Folgendes, während Sie die Schritte ausführen:

- Öffentlicher DNS-Name der EC2-Instance.
- Öffentlicher DNS-Name des Mounting-Ziels, das in derselben Availability Zone erstellt wurde, in der Sie die EC2-Instance gestartet haben.

2. (Optional) Sie können Dateisystem vom Mountingpunkt entfernen, den Sie in der Übung „Erste Schritte“ erstellt haben.

```
$ sudo umount ~/efs-mount-point
```

In dieser Anleitung erstellen Sie einen anderen Bereitstellungspunkt für das Dateisystem.

3. Installieren Sie auf Ihrer EC2-Instance den Apache-Webserver und konfigurieren Sie diesen wie folgt:
 - a. Stellen Sie eine Verbindung mit Ihrer EC2-Instance her und installieren Sie den Apache-Webserver.

```
$ sudo yum -y install httpd
```

- b. Starten Sie den Service.

```
$ sudo service httpd start
```

- c. Erstellen Sie einen Mountingpunkt.

Beachten Sie zunächst, dass DocumentRoot in der Datei `/etc/httpd/conf/httpd.conf` auf `/var/www/html` (DocumentRoot `"/var/www/html"`) zeigt.

Sie mounten Ihr Amazon EFS-Dateisystem in einem Unterverzeichnis unter dem Stammverzeichnis des Dokuments.

Erstellen Sie ein Unterverzeichnis mit `efs-mount-point` dem Namen, das Sie als Mount-Point für Ihr Dateisystem verwenden möchten, unter `/var/www/html`.

```
$ sudo mkdir /var/www/html/efs-mount-point
```

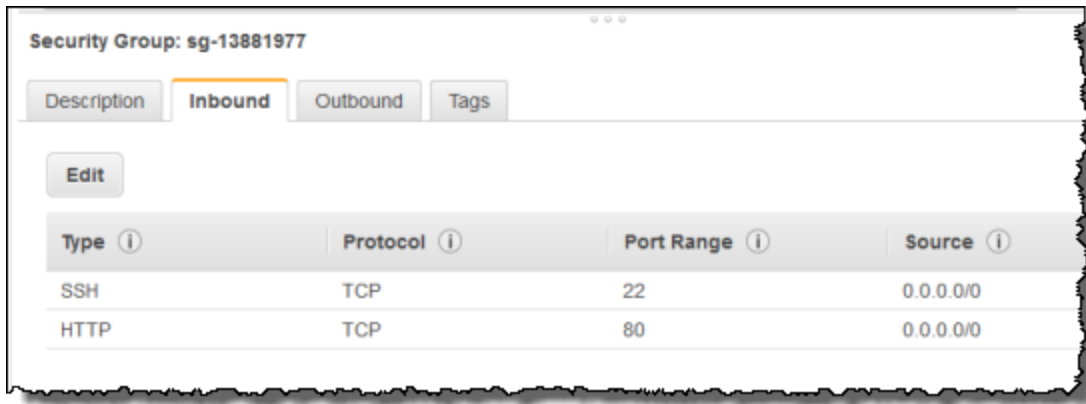
- d. Stellen Sie Ihr Amazon EFS-Dateisystem mithilfe des folgenden Befehls bereit. Ersetzen `file-system-id` Sie es durch die ID Ihres Dateisystems.

```
$ sudo mount -t efs file-system-id:/ /var/www/html/efs-mount-point
```

4. Testen Sie die Einrichtung.

- a. Fügen Sie eine Regel in der EC2-Instance-Sicherheitsgruppe hinzu, die Sie in der Übung „Erste Schritte“ erstellt haben, um von überall aus HTTP-Datenverkehr auf TCP-Port 80 zuzulassen.

Nachdem Sie die Regel hinzugefügt haben, besitzt die EC2-Instance-Sicherheitsgruppe die folgenden Regeln für den eingehenden Datenverkehr.



Detaillierte Anweisungen finden Sie unter [Erstellen von Sicherheitsgruppen mit der AWS Management Console](#).

- b. Erstellen Sie eine HTML-Beispieldatei.
 - i. Wechsle das Verzeichnis zum Mount-Point.

```
$ cd /var/www/html/efs-mount-point
```

- ii. Erstellen Sie ein aufgerufenes Unterverzeichnissampledirt und ändern Sie den Besitzer.

```
$ sudo mkdir sampledirt
$ sudo chown ec2-user sampledirt
$ sudo chmod -R o+r sampledirt
```

Ändern Sie das Verzeichnis, damit Sie Dateien imsampledirt Unterverzeichnis erstellen können.

```
$ cd sampledirt
```

- iii. Erstellen Sie eine hello.html-Beispieldatei.

```
$ echo "<html><h1>Hello from Amazon EFS</h1></html>" > hello.html
```

- c. Öffnen Sie ein Browserfenster und geben Sie die URL für den Zugriff auf die Datei ein. (Dies ist der öffentliche DNS-Name der EC2-Instance, gefolgt vom Dateinamen). Beispiel:

```
http://EC2-instance-public-DNS/efs-mount-point/sampled-dir/hello.html
```

Jetzt stellen Sie Webseiten bereit, die in einem Amazon EFS-Dateisystem gespeichert sind.

Note

Bei diesem Setup wird die EC2-Instance nicht so konfiguriert, dass der Webserver (httpd) beim Booten automatisch gestartet wird, und das Dateisystem wird auch nicht beim Booten gemountet. In der nächsten Anleitung erstellen Sie eine Startkonfiguration, um dies einzurichten.

Mehrere EC2-Instanzen, die Dateien bereitstellen

Folgen Sie den Schritten, um dieselben Inhalte in Ihrem Amazon EFS-Dateisystem von mehreren EC2-Instances aus bereitzustellen, um die Skalierbarkeit oder Verfügbarkeit zu verbessern.

1. Folgen Sie den Schritten in der [Schritt 1: Erstellen eines Amazon-EFS-Dateisystems](#) Übung, damit Sie ein Amazon EFS-Dateisystem erstellt und getestet haben.

Important

In dieser Anleitung verwenden Sie nicht die EC2-Instance, die Sie in der Übung „Erste Schritte“ erstellt haben. Stattdessen starten Sie neue EC2-Instances.

2. Erstellen Sie in Ihrer VPC einen Load Balancer, indem Sie die folgenden Schritte ausführen.
 - a. Definieren Sie einen Load Balancer

Wählen Sie im Abschnitt Basic Configuration (Grundlegende Konfiguration) die VPC aus, in der Sie auch die EC2-Instances erstellen, auf denen Sie das Dateisystem mounten.

Wählen Sie im Abschnitt Subnetze auswählen alle verfügbaren Subnetze aus. Weitere Informationen finden Sie im Skript `c1oud-config` im nächsten Abschnitt.

b. Zuweisen von Sicherheitsgruppen

Erstellen Sie eine neue Sicherheitsgruppe für den Load Balancer, um den HTTP-Zugriff von Port 80 von beliebigen Stellen aus zuzulassen, wie im Folgenden gezeigt:

- Typ: HTTP
- Protocol (Protokoll): TCP
- Portbereich: 80
- Quelle: Anywhere (0.0.0.0/0)

 Note

Wenn alles funktioniert, können Sie auch die Sicherheitsgruppenregel der EC2-Instance für den eingehenden Datenverkehr aktualisieren, damit diese HTTP-Datenverkehr nur vom Load Balancer zulässt.

c. Konfigurieren von Zustandsprüfungen

Legen Sie den Wert Ping Path (Ping-Pfad) auf `/efs-mount-point/test.html` fest. `efs-mount-point` ist das Unterverzeichnis, in dem Sie das Dateisystem gemounted haben. Sie fügen die Seite `test.html` in einem späteren Schritt dieses Verfahrens hinzu.

 Note

Fügen Sie keine EC2-Instances hinzu. Später erstellen Sie eine Auto Scaling-Gruppe, in der Sie die EC2-Instance starten und diesen Load Balancer angeben.

Informationen zum Erstellen eines Load-Balancers finden Sie unter [Erste Schritte mit Elastic Load Balancing](#) im Leitfaden Amazon-Load-Balancing-Benutzerhandbuch.

Erstellen Sie eine Auto Scaling-Gruppe mit zwei EC2-Instances. Zuerst erstellen Sie eine Startkonfiguration, die die Instances beschreibt. Anschließend erstellen Sie eine Auto

Scaling-Gruppe, indem Sie die Startkonfiguration angeben. Die folgenden Schritte enthalten Konfigurationsinformationen, die Sie angeben, um eine Auto Scaling Scaling-Gruppe von der Amazon EC2-Konsole aus zu erstellen.

1. Wählen Sie Launch Configurations (Startkonfigurationen) unter AUTO SCALING im linken Navigationsmenü.
2. Wählen Sie Create Auto Scaling group (Auto Scaling-Gruppe erstellen), um den Assistenten zu starten.
3. Wählen Sie Create launch configuration.
4. Wählen Sie unter Quick Start die neueste Version des Amazon Linux 2-AMI aus. Dies ist das gleiche AMI, das Sie in [Schritt 2: Erstellen Sie Ihre EC2-Ressourcen und starten Sie Ihre EC2-Instance](#) der Übung „Erste Schritte“ verwendet haben.
5. Führen Sie im Abschnitt Advanced (Erweitert) folgende Schritte aus:
 - Wählen Sie für IP Address Type (IP-Adresstyp) die Option Assign a public IP address to every instance (Jeder Instance eine öffentliche IP-Adresse zuweisen).
 - Kopieren Sie das folgende Skript und fügen Sie es in das Feld User data (Benutzerdaten) ein.

Sie müssen das Skript aktualisieren, indem Sie Werte für die *file-system-id* und *aws-Region angeben* (wenn Sie die Übung Erste Schritte befolgt haben, haben Sie das Dateisystem in der Region us-west-2 erstellt).

Beachten Sie im Skript Folgendes:

- Das Skript installiert den NFS-Client und den Apache-Webserver.
- Der Befehl "echo" schreibt den folgenden Eintrag in die Datei `/etc/fstab`, mit dem der DNS-Name des Dateisystems und das Unterverzeichnis identifiziert werden, auf dem dieses gemountet werden soll. Dieser Eintrag stellt sicher, dass die Datei nach jedem Neustart des Systems gemountet wird. Beachten Sie, dass der DNS-Name des Dateisystems dynamisch erstellt wird. Weitere Informationen finden Sie unter [Mounting auf Amazon EC2 mit einem DNS-Namen](#).

```
file-system-ID.efs.aws-region.amazonaws.com:/ /var/www/html/efs-mount-point  
nfs4 defaults
```

- Erzeugt ein `efs-mount-point` Unterverzeichnis und mountet das Dateisystem darauf.
- Erstellt ein `test.html` Seite, auf der ELB Health Check die Datei finden kann (beim Erstellen eines Load Balancers haben Sie diese Datei als Pingpunkt angegeben).

Weitere Informationen zu Benutzerdatenskripts finden Sie unter [Hinzufügen von Benutzerdaten](#) im Leitfaden Amazon-EC2-Benutzerhandbuch für Linux-Instances.

```
#cloud-config
package_upgrade: true
packages:
- nfs-utils
- httpd
runcmd:
- echo "$(curl -s http://169.254.169.254/latest/meta-data/placement/availability-zone).file-system-id.efs.aws-region.amazonaws.com:/ /var/www/html/efs-mount-point nfs4 defaults" >> /etc/fstab
- mkdir /var/www/html/efs-mount-point
- mount -a
- touch /var/www/html/efs-mount-point/test.html
- service httpd start
- chkconfig httpd on
```

6. Wählen Sie für Assign a security group (Sicherheitsgruppe zuweisen) die Option Select an existing security group (Eine vorhandene Sicherheitsgruppe wählen) und wählen Sie dann die Sicherheitsgruppe aus, die Sie für die EC2-Instance erstellt haben.
7. Konfigurieren Sie nun die Auto Scaling Scaling-Gruppendetails mithilfe der folgenden Informationen.
 - a. Geben Sie für Group size (Gruppengröße) den Wert **Start with 2 instances** ein. Sie erstellen nun zwei EC2-Instances.
 - b. Wählen Sie den VPC aus der Liste Network (Netzwerk) aus.
 - c. Wählen Sie ein Subnetz in derselben Availability Zone aus, die Sie verwendet haben, als Sie während der Erstellung der Startkonfiguration im vorherigen Schritt die ID des Mounting-Ziels im Benutzerdatenskript angegeben haben.
 - d. Im Abschnitt „Erweiterte Details“
 - i. Wählen Sie für Load Balancing (Lastausgleich) die Option Receive traffic from Elastic Load Balancer(s) (Datenverkehr von folgenden Elastic Load Balancern empfangen) und wählen Sie dann den Load Balancer aus, den Sie für diese Übung erstellt haben.
 - ii. Wählen Sie für Health Check Type (Art der Zustandsprüfung) die Option ELB.
8. Folgen Sie den Anweisungen zum Erstellen einer Auto Scaling Scaling-Gruppe unter [Einrichtung einer skalierten Anwendung mit Lastausgleich](#) im Amazon EC2 Auto Scaling Scaling-

Benutzerhandbuch. Verwenden Sie die Informationen in den vorherigen Tabellen, wenn zutreffend.

9. Nach Erstellung der Auto Scaling-Gruppe verfügen Sie über zwei EC2-Instances mit `nfs-utils`, auf denen der Apache-Webserver installiert ist. Stellen Sie auf jeder Instance sicher, dass Sie das `/var/www/html/efs-mount-point` Unterverzeichnis mit Ihrem Amazon EFS-Dateisystem installiert haben. Informationen zum Connect mit einer Linux-Instance finden Sie unter [Verbinden Sie sich mit der Linux-Instance](#) im Leitfaden Amazon-EC2-Benutzerhandbuch für Linux-Instances.

 **Note**

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

10. Erstellen Sie eine Beispielseite (`index.html`).

- a. Ändern Sie das Verzeichnis.

```
$ cd /var/www/html/efs-mount-point
```

- b. Erstellen Sie ein Unterverzeichnis für `sampledir` und ändern Sie den Besitzer. Ändern Sie das Verzeichnis, damit Sie im Unterverzeichnis für `sampledir` Dateien erstellen können. Wenn Sie [Einzelne EC2-Instance, die Dateien bereitstellt](#) gefolgt sind, haben Sie das Unterverzeichnis `sampledir` bereits erstellt, sodass Sie diesen Schritt überspringen können.

```
$ sudo mkdir sampledir
$ sudo chown ec2-user sampledir
$ sudo chmod -R o+r sampledir
$ cd sampledir
```

- c. Erstellen Sie eine `index.html`-Beispieldatei.

```
$ echo "<html><h1>Hello from Amazon EFS</h1></html>" > index.html
```

11. Jetzt können Sie die Einrichtung testen. Greifen Sie über den öffentlichen DNS-Namen des Load Balancer auf die Seite „`index.html`“ zu.

```
http://load balancer public DNS Name/efs-mount-point/sampled-dir/index.html
```

Der Load Balancer sendet eine Anforderung an eine der EC2-Instances, auf denen der Apache-Webserver ausgeführt wird. Dann stellt der Webserver die Datei bereit, die in Ihrem Amazon EFS-Dateisystem gespeichert ist.

Anleitung: Erstellen Sie beschreibbare Unterverzeichnisse für einzelne Benutzer und konfigurieren Sie das automatische erneute Mounting bei Neustarts

Nachdem Sie ein Amazon EFS-Dateisystem erstellt und es lokal auf Ihrer EC2-Instance gemountet haben, wird ein leeres Verzeichnis namens *Dateisystem-Stammverzeichnis verfügbar gemacht*. Eine häufiger Anwendungsfall ist das Erstellen eines „beschreibbaren“ Unterverzeichnisses unter diesem Dateisystem-Stammverzeichnis für jeden Benutzer, den Sie auf der EC2-Instance erstellen; anschließend mounten Sie dieses im Stammverzeichnis des Benutzers. Alle Dateien und Unterverzeichnisse, die der Benutzer in seinem Home-Verzeichnis erstellt, werden dann im Amazon EFS-Dateisystem erstellt.

In dieser Anleitung erstellen Sie zunächst den Benutzer „Mike“ auf Ihrer EC2-Instance. Anschließend mounten Sie ein Amazon EFS-Unterverzeichnis in das Home-Verzeichnis von Benutzer mike. Weiterhin wird erläutert, wie Sie das automatische erneute Mounting von Unterverzeichnissen bei Neustarts des Systems konfigurieren.

Angenommen, Sie haben ein Amazon EFS-Dateisystem erstellt und in einem lokalen Verzeichnis auf Ihrer EC2-Instance gemountet. Nennen wir dies *EFSroot*.

Note

Zum Mounting eines Amazon EFS-Dateisystems auf Ihrer EC2-Instance können Sie der [Erste Schritte](#) Übung folgen und ein Mounting-Mounting-System auf Ihrer EC2-Instance einrichten.

In den folgenden Schritten erstellen Sie einen Benutzer (mike), erstellen ein Unterverzeichnis für den Benutzer (*efsRoot/mike*), machen den Benutzer mike zum Eigentümer des Unterverzeichnisses,

gewähren ihm vollständige Berechtigungen und mounten schließlich das Amazon EFS-Unterverzeichnis im Home-Verzeichnis des Benutzers (/home/mike).

1. Erstellen des Benutzers „Mike“:

- Melden Sie sich bei Ihrer EC2-Instance an. Erstellen Sie unter Verwendung der Root-Berechtigungen (in diesem Fall mit dem Befehl `sudo`) den Benutzer `mike`, und weisen Sie ihm ein Kennwort zu.

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

Dadurch wird auch ein Stammverzeichnis für den Benutzer erstellt: /home/mike.

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

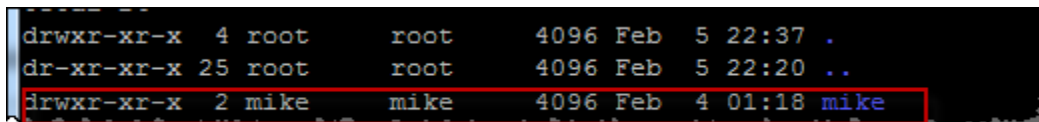
- ### a. Erstellen Sie ein Unterverzeichnis `mike` unter *EFSroot*.

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

Sie müssen *EFSroot* durch den Namen Ihres lokalen Verzeichnisses ersetzen.

- ### b. Der Root-Benutzer und die Root-Gruppe sind die Eigentümer des /mike-Unterverzeichnisses (Sie können dies überprüfen, indem Sie den `ls -l` Befehl verwenden). Um vollständige Berechtigungen für den Benutzer `mike` in diesem Unterverzeichnis zu aktivieren, gewähren Sie `mike` die Eigentümerschaft für dieses Verzeichnis.

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



```
drwxr-xr-x 4 root    root    4096 Feb  5 22:37 .  
dr-xr-xr-x 25 root    root    4096 Feb  5 22:20 ..  
drwxr-xr-x 2 mike    mike    4096 Feb  4 01:18 mike
```

3. Verwenden Sie den `mount`-Befehl zum Mounten des Unterverzeichnisses *EFSroot*/mike auf dem Stammverzeichnis von Mike.

```
$ 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 das Remote-Stammverzeichnis des Amazon EFS-Dateisystems.

Das Home-Verzeichnis von Benutzer mike ist jetzt ein Unterverzeichnis im Amazon EFS-Dateisystem, das von mike beschreibbar ist. Wenn Sie das Mounting dieses Mounting-Ziels aufheben, kann der Benutzer nicht mehr auf sein EFS-Verzeichnis zugreifen, ohne ein erneutes Mounting durchzuführen; dazu sind Root-Berechtigungen erforderlich.

Automatische Remountion beim Neustart

Mit der Datei `fstab` können Sie dafür sorgen, dass Ihr Dateisystem nach einem Systemneustart automatisch erneut gemountet wird. Weitere Informationen finden Sie unter [Automatisches Mounting des Amazon EFS-Dateisystems](#).

Exemplarische Vorgehensweise: Erstellen und Bereitstellen eines lokalen Dateisystems mit VPN AWS Direct Connect

In dieser schrittweisen Anleitung wird AWS Management Console verwendet, um ein Dateisystem auf einem Client vor Ort zu erstellen und zu mounten. Dazu verwenden Sie entweder eine AWS Direct Connect Verbindung oder eine Verbindung über eine AWS Virtual Private Network (VPC) AWS VPN.

Note

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

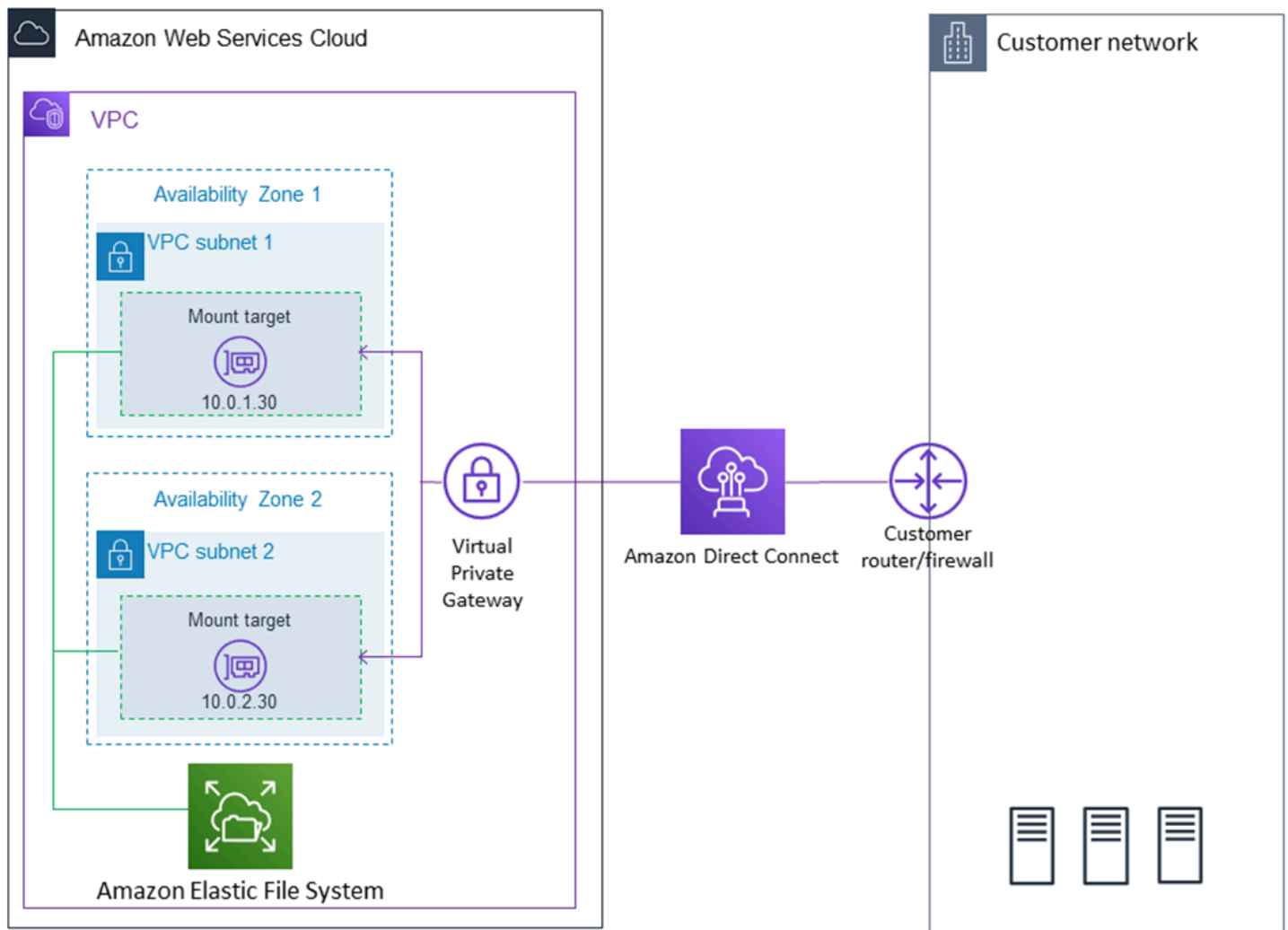
Themen

- [Bevor Sie beginnen](#)
- [Schritt 1: Erstellen Sie Ihre Amazon Elastic File System-Ressourcen](#)
- [Schritt 2: Installieren Sie den NFS-Client](#)
- [Schritt 3: Mounten Sie das Amazon EFS-Dateisystem auf Ihrem lokalen Client](#)
- [Schritt 4: Bereinigen Sie Ressourcen und schützen Sie Ihr AWS Konto](#)
- [Optional: Verschlüsseln von Daten während der Übertragung](#)

In dieser schrittweisen Anleitung nehmen wir an, dass Sie bereits über eine AWS Direct Connect- oder VPN-Verbindung verfügen. Falls nicht, können Sie jetzt mit dem Verbindungsprozess beginnen und hierher zurückkehren, wenn Ihre Verbindung hergestellt ist. Weitere Informationen zu AWS Direct Connect finden Sie im [AWS Direct Connect-Benutzerhandbuch](#). Weitere Informationen zum Einrichten einer VPN-Verbindung finden Sie unter [VPN-Verbindungen](#) im Amazon VPC-Benutzerhandbuch.

Wenn Sie eine AWS Direct Connect oder VPN-Verbindung haben, erstellen Sie ein Amazon 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 Client vor Ort aus. Der Bereinigungsschritt am Ende der Anleitung stellt Informationen zum Entfernen dieser Ressourcen bereit.

In der exemplarischen Vorgehensweise werden all diese Ressourcen in der Region USA West (Oregon) erstellt (`us-west-2`). Was auch immer AWS-Region Sie verwenden, stellen Sie sicher, dass Sie es konsistent verwenden. Alle Ihre Ressourcen — Ihre VPC, Ihr Mount-Ziel und Ihr Amazon 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 Mountingpunkts zu verweisen, wenn der erste Mountingpunkt vorübergehend nicht verfügbar ist. In diesem Szenario empfehlen wir, dass Sie zwei Clients vor Ort einsetzen, die mit Ihrem Dateisystem über verschiedene Availability Zones (AZs) verbunden sind, um eine größere Verfügbarkeit zu gewährleisten.

Bevor Sie beginnen

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 [Einrichten](#).

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 [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 Amazon Elastic File System-Ressourcen

In diesem Schritt erstellen Sie Ihr Amazon EFS-Dateisystem und mounten Ziele.

So erstellen Sie Ihr Amazon EFS-Dateisystem

1. Öffnen Sie die Amazon EFS-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. Markieren Sie die Kontrollkästchen für alle Availability Zones. 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 [Erstellen und Verwalten von Mount-Zielen und Sicherheitsgruppen](#).
5. Wählen Sie Next Step (Weiter) aus.
6. Benennen Sie Ihr Dateisystem, lassen Sie general purpose (allgemeiner Zweck) als Standardleistungsmodus ausgewählt, und klicken Sie auf Next Step (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 Security group (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 Mount-Ziels keine Sicherheitsgruppe angegeben haben, ordnet Amazon EFS ihr die Standardsicherheitsgruppe der VPC zu. 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. Verwenden Sie die AWS Management Console, um die Regel den Sicherheitsgruppen Ihres Mounting-Ziels in Ihrer VPC hinzuzufügen.

So ermöglichen Sie eingehenden Datenverkehr zum NFS-Port:

1. Melden Sie sich bei der AWS Management Console an und öffnen Sie die Amazon-EC2-Konsole unter <https://console.aws.amazon.com/EC2/>.
2. Wählen Sie unter NETWORK & SECURITY (Netzwerk und Sicherheit) die Option Security Groups (Sicherheitsgruppen) aus.
3. Wählen Sie die Sicherheitsgruppe für Ihr Dateisystem. Sie haben sich diese am Ende von [Schritt 1: Erstellen Sie Ihre Amazon Elastic File System-Ressourcen](#) notiert.
4. Wählen Sie im Registerkartenbereich unter der Liste der Sicherheitsgruppen die Registerkarte Inbound (Eingehend) aus.
5. Wählen Sie Edit (Bearbeiten) aus.
6. Klicken Sie auf Add Rule (Regel hinzufügen) und wählen Sie eine Regel des folgenden Typs aus:
 - Typ – NFS
 - Quelle – Anywhere (Beliebig)

Wir empfehlen, dass Sie für Tests nur die Quelle Anywhere (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 My IP (Meine IP) aus.

 Note

Sie müssen keine ausgehende Regel hinzufügen, da die Standardausgangsregel jeden Datenverkehr nach außen zulässt. Wenn Sie diese Standardausgangsregel nicht haben,

fügen Sie eine ausgehende Regel hinzu, um eine TCP-Verbindung auf dem NFS-Port zu öffnen, wobei die Sicherheitsgruppe des Mounting-Ziels als Ziel identifiziert wird.

Schritt 2: Installieren Sie den NFS-Client

In diesem Schritt installieren Sie den NFS-Client.

So installieren Sie den NFS-Client auf Ihrem lokalen 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: Verschlüsseln 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 Amazon 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 Amazon EFS-Dateisystem bereitgestellt haben, können Sie es mit dem folgenden Verfahren testen.

Um die Amazon EFS-Dateisystemverbindung zu testen

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
```

Sie können das Dateisystem auch automatisch mounten, indem Sie der Datei `/etc/fstab` einen Eintrag hinzufügen. Weitere Informationen finden Sie unter [Automatisches Mounting des Amazon EFS-Dateisystems](#).

 Warning

Verwenden Sie beim automatischen Mounten 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

Wenn Sie diese exemplarische Vorgehensweise abgeschlossen haben oder wenn Sie die exemplarischen Vorgehensweisen nicht näher untersuchen möchten, sollten Sie die folgenden Schritte ausführen, um Ihre Ressourcen zu bereinigen und Ihr Konto zu schützen. AWS

Um Ressourcen zu bereinigen und Ihre AWS-Konto

1. Hängen Sie das Amazon EFS-Dateisystem mit dem folgenden Befehl aus.

```
$ sudo umount ~/efs
```

2. Öffnen Sie die Amazon EFS-Konsole unter <https://console.aws.amazon.com/efs/>.
3. Wählen Sie das Amazon EFS-Dateisystem, das Sie löschen möchten, aus der Liste der Dateisysteme aus.
4. Klicken Sie bei Actions (Aktionen) auf Delete file system (Dateisystem löschen).

5. Geben Sie im Dialogfeld Dateisystem dauerhaft löschen die Dateisystem-ID für das Amazon EFS-Dateisystem ein, das Sie löschen möchten, und wählen Sie dann 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: Verschlüsseln von Daten während der Übertragung

Um Daten während der Übertragung zu verschlüsseln, verwenden Sie den Amazon EFS-Mount-Helfer anstelle des NFS-Clients. `amazon-efs-utils`

Das `amazon-efs-utils` Paket 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 [Verwenden der amazon-efs-utils Tools](#). 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 Client vor Ort. Folgende Distributionen werden unterstützt:

- Amazon Linux 2
- Amazon Linux
- Red Hat Enterprise Linux (und Derivate wie z. B. CentOS), Version 7 und höher
- 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 AWS Direct Connect oder VPN.

Zur Konfiguration amazon-efs-utils für die Verwendung in Ihrem AWS-Region

1. Öffnen Sie mit einem Texteditor Ihrer Wahl die Datei `/etc/amazon/efs/efs-utils.conf` zur Bearbeitung.
2. Suchen Sie die Zeile `dns_name_format = {fs_id}.efs.{region}.amazonaws.com`.
3. Ändern Sie dies beispielsweise `{region}` mit der ID für Ihre AWS Regionus-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 zu mounten, benötigen Sie die Dateisystem-ID, die Mount-Ziel-IP-Adresse für eines Ihrer Mount-Ziele und die des DateisystemsAWS-Region. 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 mount-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 Mountingbefehl in etwa folgendermaßen aus.

Example

```
sudo mount -t efs -o tls fs-12345678 ~/efs
```

Exemplarische Vorgehensweise: Mounten eines Dateisystems aus einer anderen VPC

In dieser exemplarischen Vorgehensweise richten Sie eine Amazon EC2 EC2-Instance ein, um ein Amazon 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 [Verwenden der amazon-efs-utils Tools](#).

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 Amazon EC2 EC2-Instances, die sich in einer VPC befinden, auf EFS-Dateisysteme in einer anderen VPC zugreifen, auch wenn die VPCs zu unterschiedlichen Konten gehören.

 Note

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

Themen

- [Bevor Sie beginnen](#)
- [Schritt 1: Ermitteln der Availability Zone-ID des EFS-Mounting-Ziels](#)
- [Schritt 2: Bestimmen der IP-Adresse des Mounting-Ziels](#)
- [Schritt 3: Hinzufügen eines Hosteintrags für das Mounting-Ziel](#)
- [Schritt 4: Mounting des Dateisystems mithilfe der EFS-Mountinghilfe](#)
- [Schritt 5: Bereinigen Sie Ressourcen und schützen Sie Ihr AWS Konto](#)

Bevor Sie beginnen

In dieser Anleitung wird Folgendes vorausgesetzt:

- 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 [Verwenden der amazon-efs-utils Tools](#).
- 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. Mit diesem Verbindungstyp können Sie Datenverkehr dazwischen über private IPv4 (Internet Protocol Version 4) oder IPv6-Adressen (Internet Protocol Version 6) weiterleiten. Sie können VPC-Peering verwenden, um VPCs innerhalb derselben AWS-Region oder zwischen AWS-Regionen zu verbinden. Weitere Informationen finden Sie unter [Erstellen und Akzeptieren einer VPC-Peering-Verbindung](#) im Amazon VPC Peering Guide.

- 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, mit dem Sie Ihre VPCs und lokale Netzwerke miteinander verbinden können. Weitere Informationen finden Sie unter [Erste Schritte mit Transit Gateways](#) im Amazon VPC Transit Gateways Guide.

Schritt 1: Ermitteln der Availability Zone-ID des EFS-Mounting-Ziels

Um eine hohe Verfügbarkeit Ihres Dateisystems zu gewährleisten, empfehlen wir, immer eine EFS-Mount-Ziel-IP-Adresse zu verwenden, die sich in derselben Availability Zone wie Ihr 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-Mount-Ziel in derselben Availability Zone-ID befinden. Diese Anforderung gilt, da die Namen der Availability Zones zwischen den Konten unterschiedlich sein können.

Um die Availability Zone-ID der EC2-Instance zu ermitteln

1. Stellen Sie eine Verbindung mit Ihrer EC2-Instance her:
 - Wenn Sie von einem Computer unter macOS oder Linux eine Verbindung mit Ihrer Instance herzustellen möchten, geben Sie die PEM-Datei für den SSH-Befehl an. Verwenden Sie dazu die `-i`-Option und den Pfad zu Ihrem privaten Schlüssel.
 - Wenn Sie von einem Computer mit Windows eine Verbindung zu Ihrer Instance herstellen möchten, können Sie PuTTY verwenden. MindTerm Zur Verwendung von PuTTY installieren Sie es und konvertieren Sie die PEM-Datei in eine PPK-Datei.

Weitere Informationen finden Sie unter den folgenden Themen im Amazon EC2 EC2-Benutzerhandbuch für Linux-Instances:

- [Herstellen einer Verbindung mit Ihrer Linux-Instance per SSH](#)
 - [Herstellung einer Verbindung zu Ihrer Linux-Instance von Windows mit PuTTY](#)
2. Ermitteln Sie die Availability Zone-ID, in der sich die EC2-Instance befindet, mithilfe des `describe-availability-zones` CLI-Befehls 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 Availability Zone-ID wird in der `ZoneId` Eigenschaft zurückgegeben `use2-az2`.

Schritt 2: Bestimmen der IP-Adresse des Mounting-Ziels

Da Sie nun die Availability Zone-ID der EC2-Instance kennen, können Sie jetzt die IP-Adresse des Mount-Ziels abrufen, das sich in derselben Availability Zone-ID befindet.

Um die IP-Adresse des Mount-Ziels in derselben Availability Zone-ID zu ermitteln

- 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 Mount-Ziel in der `use2-az2` Availability Zone-ID hat die IP-Adresse 10.0.2.153.

Schritt 3: Hinzufügen eines Hosteintrags für das Mounting-Ziel

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-System nach Bedarf ermöglichen. Weitere Informationen finden Sie unter [Verwendung von VPC-Sicherheitsgruppen für Amazon EC2-Instance und Mounting-Ziele](#).

Schritt 4: Mounting des Dateisystems mithilfe der EFS-Mountinghilfe

Um Ihr EFS-Dateisystem zu mounten, erstellen Sie zunächst ein Mounting-Verzeichnis auf der EC2-Instance. Anschließend können Sie mit der EFS-Mountinghilfe das Dateisystem entweder mit IAM-Autorisierung oder einem EFS-Zugriffspunkt mounten. Weitere Informationen erhalten Sie unter [Verwendung von IAM zur Steuerung des Dateisystemdatenzugriffs](#) und [Arbeiten mit Amazon EFS Access Points](#).

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-Zugriffspunkts

- Verwenden Sie den folgenden Befehl, um das Dateisystem mithilfe eines EFS-Zugriffspunkts zu mounten.

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

Nachdem Sie Ihr Amazon EFS-Dateisystem gemountet haben, können Sie es mit dem folgenden Verfahren testen.

Wenn Sie die Amazon EFS-Dateisystemverbindung testen möchten

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 [Verwenden von `/etc/fstab` mit der EFS-Mountinghilfe, um EFS-Dateisysteme automatisch erneut zu mounten](#).

Warning

Verwenden Sie beim automatischen Mounten 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: Bereinigen Sie Ressourcen und schützen Sie Ihr AWS Konto

Nachdem Sie diese exemplarische Vorgehensweise abgeschlossen haben oder wenn Sie die Walkthroughs nicht erkunden möchten, stellen Sie sicher, dass Sie die folgenden Schritte ausführen. Diese reinigen Ihre Ressourcen und schützen Ihr AWS-Konto.

Um Ressourcen zu schonen und Ihre zu schützen AWS-Konto

1. Wenn Sie das Amazon EFS-Dateisystem abstellen möchten, können Sie den folgenden Befehl verwenden.

```
$ sudo umount ~/efs
```

2. Öffnen Sie die Amazon EFS-Konsole unter <https://console.aws.amazon.com/efs/>.
3. Wählen Sie das Amazon EFS-Dateisystem, das Sie löschen möchten, aus der Liste der Dateisysteme aus.
4. Klicken Sie bei Actions (Aktionen) auf Delete file system (Dateisystem löschen).
5. Geben Sie im Dialogfeld „Dateisystem dauerhaft löschen“ die Dateisystem-ID für das Amazon EFS-Dateisystem ein, das Sie löschen möchten, und wählen Sie dann „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.

Exemplarische Anleitung: Erzwingen der Verschlüsselung auf einem Amazon EFS-Dateisystem im Ruhezustand

Im Folgenden finden Sie Details zum Erzwingen der Verschlüsselung im Ruhezustand mit Amazon CloudWatch und AWS CloudTrail aus. Diese exemplarische Vorgehensweise basiert auf der AWS Whitepaper [Verschlüsseln gespeicherter Daten mit Amazon EFS Encrypted File Systems](#) aus.

**Note**

Die in dieser exemplarische Vorgehensweise beschriebene Methode zur Durchsetzung der Erstellung von Amazon EFS-Dateisystemen, die im Ruhezustand verschlüsselt sind, ist veraltet. Die bevorzugte Methode zur Erzwingung der Erstellung gespeicherter Dateisysteme ist die Verwendung der `elasticfilesystem:EncryptedBedingungsschlüssel` für AWS Identity and Access Management Identitätsbasierte -Richtlinien. Weitere Informationen finden Sie unter [Beispiel: Erzwingen Sie die Erstellung verschlüsselter Dateisysteme](#). Sie können

diese exemplarische Vorgehensweise verwenden, um CloudWatch-Alarme zu erstellen, um zu überprüfen, dass Ihre IAM-Richtlinien die Erstellung unverschlüsselter Dateisysteme verhindern.

Erzwingen von Verschlüsselung im Ruhezustand

Ihr Unternehmen erfordert möglicherweise die Verschlüsselung aller gespeicherten Daten, die einer bestimmten Klassifizierung zugeordnet sind oder zu einer speziellen Anwendung oder Umgebung bzw. zu einem speziellen Workload gehören. Sie können Richtlinien für die Verschlüsselung von Daten im Ruhezustand für Amazon EFS-Dateisysteme mithilfe von aufdeckenden Kontrollen durchsetzen. Diese Kontrollen erkennen die Erstellung eines Dateisystems und überprüfen, ob die Verschlüsselung im Ruhezustand aktiviert ist.

Wenn ein Dateisystem ohne Verschlüsselung im Ruhezustand erkannt wird, können Sie auf verschiedene Weise reagieren. Die Möglichkeiten reichen vom Löschen des Dateisystems und der Mounting-Ziele bis zur Benachrichtigung eines Administrators.

Wenn Sie ein im Ruhezustand nicht verschlüsseltes Dateisystem löschen, die Daten aber behalten möchten, erstellen Sie zuerst ein neues, im Ruhezustand verschlüsseltes Dateisystem. Kopieren Sie als Nächstes die Daten in dieses neue Dateisystem. Nachdem die Daten kopiert wurden, können Sie das Dateisystem ohne Verschlüsselung im Ruhezustand löschen.

Erkennen von Dateisystemen, die im Ruhezustand unverschlüsselt sind

Sie können einen CloudWatch-Alarm zum Überwachen der CloudTrail-Protokolle für `CreateFileSystemEvent`. Lösen Sie den Alarm aus, um einen Administrator zu benachrichtigen, wenn für das erstellte Dateisystem keine Verschlüsselung im Ruhezustand definiert wurde.

Erstellen eines Metrikfilters

Zum Erstellen eines CloudWatch-Alarms, der ausgelöst wird, wenn ein unverschlüsseltes Amazon EFS-Dateisystem erstellt wurde, gehen Sie wie folgt vor.

Bevor Sie beginnen, müssen Sie einen vorhandenen Trail erstellt haben, der CloudTrail-Protokolle an eine CloudWatch-Logs-Protokollgruppe sendet. Weitere Informationen finden Sie unter [Senden von Ereignissen an CloudWatch Logs](#) im AWS CloudTrail-Benutzerhandbuch.

So erstellen Sie einen Metrikfilter

1. Öffnen Sie die CloudWatch-Konsole unter <https://console.aws.amazon.com/cloudwatch/>.
2. Wählen Sie im Navigationsbereich Logs (Logs) aus.
3. Wählen Sie in der Liste von Protokollgruppen die Protokollgruppe aus, die Sie für CloudTrail-Protokollereignisse erstellt haben.
4. Wählen Sie Create Metric Filter.
5. Wählen Sie auf der Seite Define Logs Metric Filter (Protokollmetrikfilter definieren) die Option Filter Pattern (Filtermuster) aus und geben Sie Folgendes ein:

```
{ ($.eventName = CreateFileSystem) && ($.responseElements.encrypted IS FALSE) }
```

6. Wählen Sie Assign Metric.
7. Geben Sie in Filter Name **UnencryptedFileSystemCreated** ein.
8. Geben Sie für Namespace der Metrik den Wert **CloudTrailMetrics** ein.
9. Geben Sie für Metrikname den Wert **UnencryptedFileSystemCreatedEventCount** ein.
10. Wählen Sie Show advanced metric settings.
11. Geben Sie in Metric Value (Metrikwert) **1** ein.
12. Wählen Sie Create Filter (Filter erstellen).

Einrichten eines Alarms

Befolgen Sie nach der Erstellung des Metrikfilters die folgenden Schritte, um einen Alarm zu erstellen.

So erstellen Sie einen Alarm

1. Wählen Sie auf der Seite Filters (Filter) für Log_Group_Name neben dem Filternamen UnencryptedFileSystemCreated die Option Create Alarm (Alarm erstellen) aus.
2. Legen Sie auf der Seite Create Alarm (Alarm erstellen) die folgenden Parameter fest:
 - Geben Sie für Name **Unencrypted File System Created** ein.
 - Definieren Sie bei Whenever (Jedes Mal) Folgendes:
 - Legen Sie is (ist) auf **> = 1** fest.
 - Legen Sie for: (für) auf **1** aufeinanderfolgenden Zeitraum fest.

- Wählen Sie bei Treat missing data as (Fehlende Daten behandeln als) die Option good (not breaching threshold) (gut [keine Verletzung des Schwellenwerts]) aus.
 - Nehmen Sie bei Actions (Aktionen) die folgenden Einstellungen vor:
 - Wählen Sie für Whenever this alarm die Option State is ALARM aus.
 - Wählen Sie bei Send notification to (Benachrichtigung senden an) die Option NotifyMe und New list (Neue Liste) aus. Geben Sie einen eindeutigen Themennamen für diese Liste ein.
 - Geben Sie bei Email list (E-Mail-Liste) die E-Mail-Adresse ein, an die die Benachrichtigungen gesendet werden sollen. Sie sollten eine E-Mail an diese Adresse als Bestätigung darüber erhalten, dass Sie den Alarm erstellt haben.
 - Für Alarm Preview (Alarm-Vorschau) legen Sie Folgendes fest:
 - Wählen Sie als Period (Zeitraum) 1 Minute aus.
 - Legen Sie für Statistic (Statistik) die Optionen Standard und Sum (Summe) fest.
3. Wählen Sie Create Alarm (Alarm erstellen) aus.

Testen des Alarms für die Erstellung von unverschlüsselten Dateisystemen

Sie können den Alarm testen, indem Sie ein Dateisystem ohne Verschlüsselung im Ruhezustand erstellen, wie im Folgenden beschrieben.

So testen Sie den Alarm durch Erstellen eines Dateisystems ohne Verschlüsselung im Ruhezustand

1. Melden Sie sich beim anAWS Management Consoleund öffnen Sie die Amazon EFS-Konsole unter<https://console.aws.amazon.com/efs/>aus.
2. Klicken Sie aufErstellen Sie -DateisystemSo zeigen Sie den anErstellen Sie - DateisystemDialogfeld (Dialogfeld).
3. Um ein Dateisystem zu erstellen, das im Ruhezustand unverschlüsselt ist, wählen SieAnpassen vonSo zeigen Sie den anDateisystemeinstellungenangezeigten.
4. FürAllgemeinesGeben Sie Folgendes ein.
 - a. (Optional) Geben Sie einen einNamefür das -Dateisystem.
 - b. Behalten SieVerwaltung des Lebenszyklus,Leistungsmodus, undDurchsatzmodusLegen Sie auf die Standardwerte fest.
 - c. Deaktivieren SieVerschlüsselungdurch Löschen vonVerschlüsselung gespeicherter Daten aktivierenaus.

5. Klicken Sie auf **Weiter So** fahren Sie zum **Netzwerkzugriff** Schritt im Konfigurationsprozess.
6. Wählen Sie den Standardwert aus **Virtual Private Cloud (VPC)** aus.
7. Für **Mount-Ziele** Wählen Sie den Standardwert aus **Sicherheitsgruppen** Für jedes Mounting-Ziel.
8. Klicken Sie auf **Weiter So** zeigen Sie den **an Dateisystem-Richtlinie** angezeigten.
9. Klicken Sie auf **Weiter So** fahren Sie zum **Überprüfen und erstellen** angezeigten.
10. Überprüfen Sie das Dateisystem und wählen Sie **Geben Sie einen Namen für den Benutzer ein** und klicken Sie dann auf **um Ihr Dateisystem zu erstellen und zu den Dateisysteme** angezeigten.

Der Trail protokolliert den **Create FileSystem**-Betrieb und übermittelt das Ereignis an die **CloudWatch-Logs-Protokollgruppe**. Das Ereignis löst Ihren **Metrikalarm** aus und **CloudWatch Logs** sendet Ihnen eine Benachrichtigung über die Änderung.

Exemplarische Vorgehensweise: Root-Squashing mithilfe der IAM-Autorisierung für NFS-Clients aktivieren

In dieser exemplarischen Vorgehensweise konfigurieren Sie Amazon EFS so, dass **Root-Zugriff** auf Ihr Amazon EFS-Dateisystem für alle **AWS Prinzipale** mit Ausnahme einer einzelnen **Management-Workstation** verhindert wird. Dazu konfigurieren Sie die **AWS Identity and Access Management (IAM)** -Autorisierung für **Network File System (NFS)** -Clients. Weitere Hinweise zur **IAM-Autorisierung für NFS-Clients in EFS** finden Sie unter [Verwendung von IAM zur Steuerung des Dateisystemdatenzugriffs](#).

Dazu müssen zwei **IAM-Berechtigungsrichtlinien** wie folgt konfiguriert werden:

- 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 EC2-Management-Workstation**, die **Root-Zugriff** auf das Dateisystem benötigt, mithilfe eines **Amazon EC2 EC2-Instance-Profils** eine **IAM-Identität** zu. Weitere Informationen zu **Amazon EC2 EC2-Instanzprofilen** finden Sie im **AWS Identity and Access Management Benutzerhandbuch** [unter Verwenden von Instanzprofilen](#).
- Weisen Sie die **AmazonElasticFileSystemClientFullAccess** **AWS-verwaltete Richtlinie** der **IAM-Rolle** der **Management Workstation** zu. Weitere Informationen zu **AWS verwalteten Richtlinien für EFS** finden Sie unter [Identitäts- und Zugriffsmanagement für Amazon Elastic File System](#).

Verwenden Sie die folgenden Verfahren, um Root-Squashing mit IAM-Autorisierung für NFS-Clients zu aktivieren.

Um den Root-Zugriff auf das Dateisystem zu verhindern

1. Öffnen Sie die Amazon Elastic File System-Konsole unter <https://console.aws.amazon.com/efs/>.
2. Wählen Sie Dateisysteme.
3. Wählen Sie auf der Seite File systems (Dateisysteme) das Dateisystem aus, auf dem Sie Root-Squashing aktivieren möchten.
4. Wählen Sie auf der Seite mit den Dateisystemdetails die Option Dateisystemrichtlinie und dann Bearbeiten aus. Die Seite File system policy (Dateisystemrichtlinie) wird angezeigt.

Amazon EFS > File systems > fs-0d4d7e9a948cfa250 > policy

File system policy

Policy options

Select one or more of these common policy options, or create a custom policy using the editor. [Learn more](#)

- ☒ Prevent root access by default*
- ☐ Enforce read-only access by default*
- ☐ Prevent anonymous access
- ☐ Enforce in-transit encryption for all clients

* Identity-based policies can override these default permissions.

► Grant additional permissions

Policy editor {JSON} Clear

```

1 {
2   "Version": "2012-10-17",
3   "Id": "efs-policy-wizard-aa2f0cf3-ec20-41d8-b862-f979c442382b",
4   "Statement": [
5     {
6       "Sid": "efs-statement-04fb2116-6c7d-4314-8bab-d5fcf28a07c1",
7       "Effect": "Allow",
8       "Principal": {
9         "AWS": "*"
10      },
11      "Action": [
12        "elasticfilesystem:ClientWrite",
13        "elasticfilesystem:ClientMount"
14      ],
15      "Condition": {
16        "Bool": {
17          "elasticfilesystem:AccessedViaMountTarget": "true"
18        }
19      }
20    }
21  ]
22 }
```

Manual changes will prevent the use of the policy options on the left until the editor is cleared.

Cancel Save

5. Wählen Sie unter Richtlinienoptionen die Option Root-Zugriff standardmäßig verhindern*. Das Richtlinien-JSON-Objekt wird im Richtlinien-Editor angezeigt.
6. Wählen Sie Save (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 an die Rolle der Arbeitsstation anhängen, gewährt IAM auf der Grundlage seiner Identitätsrichtlinie Root-Zugriff auf die Arbeitsstation.

So aktivieren Sie den Root-Zugriff von der Management-Workstation aus:

1. Öffnen Sie die IAM-Konsole unter <https://console.aws.amazon.com/iam/>.
2. Erstellen Rolle Rolle Rolle Rolle Rolle Rolle Amazon EC2EFS-client-root-access Rolle IAM erstellt ein Instanzprofil mit demselben Namen wie die von Ihnen erstellte EC2-Rolle.
3. Weisen Sie die AWS-verwaltete Richtlinie AmazonElasticFileSystemClientFullAccess der erstellten EC2-Rolle zu. Der Inhalt dieser Richtlinie wird im Folgenden dargestellt.

```
{
  "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 Actions (Aktionen) die Option Instance Settings (Instance-Einstellungen) und dann Attach/Replace IAM role (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 Apply (Anwenden).
5. Installieren Sie den EFS-Mount-Helfer auf der Management-Workstation. Weitere Informationen zum EFS-Mount-Helfer und dem amazon-efs-utils Paket finden Sie unter [Verwenden der amazon-efs-utils Tools](#).

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 EC2-Instance so konfigurieren, dass das Dateisystem automatisch mit IAM-Autorisierung gemountet wird. Weitere Informationen zum Bereitstellen eines EFS-Dateisystems IAM-Dateisystems IAM-Dateisystems finden Sie unter [Mounting mit IAM-Autorisierung](#).

Sicherheit in Amazon EFS

Das AWS [Modell der geteilten Verantwortung](#) gilt für den Datenschutz in Amazon Elastic File System. Wie in diesem Modell beschrieben, AWS ist für den Schutz der globalen Infrastruktur verantwortlich, die alle ausführt 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 Ihnen, -Anmeldeinformationen zu schützen AWS-Konto und einzelne Benutzer mit AWS IAM Identity Center oder AWS Identity and Access Management (IAM) einzurichten. 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).
- Verwenden Sie SSL/TLS für die Kommunikation mit - AWS Ressourcen. Wir benötigen TLS 1.2 und empfehlen TLS 1.3.
- Richten Sie die API- und Benutzeraktivitätsprotokollierung mit ein AWS CloudTrail.
- Verwenden Sie AWS Verschlüsselungslösungen zusammen mit allen Standardsicherheitskontrollen in 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 auf AWS über eine Befehlszeilenschnittstelle oder eine API FIPS-140-2-validierte kryptografische Module benötigen, verwenden Sie einen FIPS-Endpunkt. Weitere Informationen über verfügbare FIPS-Endpunkte finden Sie unter [Federal Information Processing Standard \(FIPS\) 140-2](#).

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 EFS oder anderen AWS-Services über die Konsole, API AWS CLI oder AWS SDKs arbeiten. 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](#)
- [Identitäts- und Zugriffsmanagement für Amazon Elastic File System](#)
- [Verwendung von IAM zur Steuerung des Dateisystemdatenzugriffs](#)
- [Steuerung des Netzwerkzugriffs auf Amazon EFS-Dateisysteme für NFS-Clients](#)
- [Mit Benutzern, Gruppen und Berechtigungen auf Network File System-\(NFS-\)Level arbeiten](#)
- [Arbeiten mit Amazon EFS Access Points](#)
- [Blockieren des öffentlichen Zugriffs](#)
- [Konformitätsprüfung für Amazon Elastic File System](#)
- [Belastbarkeit im Amazon Elastic File System](#)
- [Netzwerkisolierung von Amazon Elastic File System](#)

Datenverschlüsselung in Amazon EFS

Amazon EFS unterstützt zwei Formen der Verschlüsselung für Dateisysteme, die Verschlüsselung von Daten während der Übertragung und die Verschlüsselung im Ruhezustand. Sie können die Verschlüsselung von Daten im Ruhezustand aktivieren, wenn Sie ein Amazon EFS-Dateisystem erstellen. Sie können die Datenverschlüsselung während der Übertragung aktivieren, wenn Sie das Dateisystem mounten.

Wenn Sie für den Zugriff auf AWS über eine Befehlszeilenschnittstelle oder eine API FIPS-140-2-validierte kryptografische Module benötigen, verwenden Sie einen FIPS-Endpunkt. Weitere Informationen über verfügbare FIPS-Endpunkte finden Sie unter [Federal Information Processing Standard \(FIPS\) 140-2](#).

Verwendung von Verschlüsselung

Wenn Ihr Unternehmen Unternehmensrichtlinien oder gesetzlichen Vorschriften unterliegt, die eine Verschlüsselung von Daten und Metadaten im Ruhezustand vorschreiben, empfehlen wir Ihnen, ein Dateisystem zu erstellen, das im Ruhezustand verschlüsselt ist, und Ihr Dateisystem mit einer Verschlüsselung der Daten während der Übertragung zu mounten.

Verwandte Themen

Weitere Informationen zur Verschlüsselung mit Amazon EFS finden Sie in diesen verwandten Themen:

- [Arbeiten mit Amazon-EFS-Ressourcen](#)
- [Zugriffsverwaltung auf verschlüsselte Dateisysteme](#)
- [Tipps zur Amazon EFS-Leistung](#)
- [Amazon EFS-Protokolldateieinträge für encrypted-at-rest Dateisysteme](#)
- [Fehlerbehebung bei der Verschlüsselung](#)

Verschlüsseln von Daten im Ruhezustand

Sie können verschlüsselte Dateisysteme mithilfe der AWS CLI, AWS Management Console oder programmgesteuert über die Amazon-EFS-API oder eines der AWS SDKs erstellen. Ihr Unternehmen erfordert möglicherweise die Verschlüsselung aller Daten, die einer bestimmten Klassifizierung entsprechen oder zu einer speziellen Anwendung oder Umgebung bzw. zu einem speziellen Workload gehören.

Sobald Sie ein EFS-Dateisystem erstellt haben, können Sie dessen Verschlüsselungseinstellung nicht mehr ändern. Das bedeutet, dass Sie ein unverschlüsseltes Dateisystem nicht so ändern können, dass es verschlüsselt wird. Stattdessen müssen Sie ein neues, verschlüsseltes Dateisystem erstellen.

Note

Die AWS Schlüsselverwaltungsinfrastruktur verwendet von Federal Information Processing Standards (FIPS) 140-2 genehmigte kryptografische Algorithmen. Die Infrastruktur entspricht den Empfehlungen der National Institute of Standards and Technology (NIST) 800-57.

Erzwingung der Erstellung von im Ruhezustand verschlüsselten Amazon EFS-Dateisystemen

Sie können den `elasticfilesystem:Encrypted` IAM-Bedingungsschlüssel in AWS Identity and Access Management (IAM) identitätsbasierten Richtlinien verwenden, um zu steuern, ob Benutzer Amazon EFS-Dateisysteme erstellen können, die im Ruhezustand verschlüsselt sind. Weitere

Informationen zum Verwenden des Bedingungsschlüssels finden Sie unter [Beispiel: Erzwingen Sie die Erstellung verschlüsselter Dateisysteme](#).

Sie können auch Service-Kontrollrichtlinien (SCPs) innerhalb AWS Organizations von definieren, um die EFS-Verschlüsselung für alle in Ihrer Organisation zu AWS-Konto erzwingen. Weitere Informationen zu Service-Kontrollrichtlinien in finden Sie AWS Organizations unter [Service-Kontrollrichtlinien](#) im AWS Organizations -Benutzerhandbuch.

Verschlüsselung von Dateisystemen im Ruhezustand mithilfe der Konsole

Wenn Sie mit der Amazon EFS-Konsole ein neues Dateisystem erstellen, ist die Verschlüsselung im Ruhezustand standardmäßig aktiviert. Im folgenden Verfahren wird beschrieben, wie Sie für ein neues Dateisystem während der Erstellung in der Konsole die Verschlüsselung aktivieren.

Note

Die Verschlüsselung im Ruhezustand ist nicht standardmäßig aktiviert, wenn Sie ein neues Dateisystem mit der AWS CLI, API und den SDKs erstellen. Weitere Informationen finden Sie unter [Erstellen eines Dateisystems mithilfe der AWS CLI](#).

So verschlüsseln Sie ein neues Dateisystem mit der EFS-Konsole

1. Öffnen Sie die Amazon Elastic File System-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:
 - Die Verschlüsselung von Daten im Ruhezustand wurde mit Ihrem Standard- AWS KMS key für Amazon EFS (aws/elasticfilesystem) aktiviert.
 - Automatische Backups aktiviert – Weitere Informationen finden Sie unter [Sichern Ihrer Amazon-EFS-Dateisysteme](#).
 - Mounting-Ziele — Amazon EFS erstellt Mounting-Ziele mit den folgenden Einstellungen:
 - Befindet sich in jeder Availability Zone in der AWS-Region , in der das Dateisystem erstellt wird.

- Befindet sich in den Standard-Subnetzen der von Ihnen ausgewählten VPC.
- Verwenden Sie die Standard-Sicherheitsgruppe der VPC. Sie können Sicherheitsgruppen verwalten, nachdem das Dateisystem erstellt wurde.

Weitere Informationen finden Sie unter [Verwalten der Netzwerkzugänglichkeit des Dateisystems](#).

- General Purpose Performance-Modus – Weitere Informationen finden Sie unter [Leistungsmodi](#).
 - Elastischer Durchsatzmodus – Weitere Informationen finden Sie unter [Durchsatzmodi](#).
 - Lebenszyklusmanagement mit einer 30-Tage-Richtlinie – Weitere Informationen finden Sie unter [Verwaltung des Dateisystemspeichers](#).
6. Die Seite Dateisysteme erscheint mit einem Banner oben, das den Status des von Ihnen erstellten Dateisystems anzeigt. Wenn das Dateisystem verfügbar ist, erscheint im Banner ein Link, über den Sie die Detailseite des Dateisystems aufrufen können.

Sie haben jetzt ein neues encrypted-at-rest Dateisystem.

Funktionsweise der Verschlüsselung im Ruhezustand

Auf einem verschlüsselten Dateisystem werden Daten und Metadaten automatisch verschlüsselt, bevor sie auf das Dateisystem geschrieben werden. Umgekehrt werden bei Lesevorgängen Daten und Metadaten entschlüsselt, bevor sie an die Anwendung gesendet werden. Diese Vorgänge werden transparent von Amazon EFS gehandhabt, so dass Sie Ihre Anwendungen nicht ändern müssen.

Amazon EFS verwendet den branchenüblichen Verschlüsselungsalgorithmus AES-256 zur Verschlüsselung von EFS-Daten und -Metadaten im Ruhezustand. Weitere Informationen finden Sie unter [Grundlagen der Kryptographie](#) im AWS Key Management Service -Developer Guide.

So verwendet Amazon EFS AWS KMS

Amazon EFS lässt sich für die Schlüsselverwaltung in AWS Key Management Service (AWS KMS) integrieren. Amazon EFS verwendet vom Kunden verwaltete Schlüssel, um Ihr Dateisystem auf folgende Weise zu verschlüsseln:

- Verschlüsseln von Metadaten im Ruhezustand – Amazon EFS verwendet die Von AWS verwalteter Schlüssel für Amazon EFS , `aws/elasticfilesystem`, um Metadaten des Dateisystems (d. h. Dateinamen, Verzeichnisnamen und Verzeichnisinhalte) zu verschlüsseln und zu 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 standardmäßige vom Kunden verwaltete Schlüssel, `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 [Erstellen von Schlüsseln](#) im - AWS Key Management Service Entwicklerhandbuch.

Wenn Sie einen vom Kunden verwalteten Schlüssel (CMK) für die Ver- und Entschlüsselung von Dateidaten verwenden, können Sie die Schlüsselrotation aktivieren. Wenn Sie die Schlüsseldrehung aktivieren, dreht Ihren Schlüssel AWS KMS automatisch einmal pro Jahr. 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 [Deaktivieren, Löschen oder Widerrufen des Zugriffs für den KMS-Schlüssel eines Dateisystems](#).

 **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. Konto-IDs, AWS die für Amazon EFS spezifisch sind, werden jedoch in Ihren AWS CloudTrail Protokollen im Zusammenhang mit - AWS KMS Aktionen angezeigt. Weitere Informationen finden Sie unter [Amazon EFS-Protokolldateieinträge für encrypted-at-rest Dateisysteme](#).

Amazon-EFS-Schlüsselrichtlinien 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. In der folgenden Liste werden alle AWS KMS-bezogenen Berechtigungen beschrieben, 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 serverseitig mit einem neuen vom Kunden verwalteten Schlüssel, ohne den Klartext der Daten clientseitig preiszugeben. 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 ist. Diese Berechtigung ist in der Standard-Schlüsselrichtlinie unter `kms:GenerateDataKey*` enthalten.
- `kms:CreateGrant` – (Erforderlich) Fügt einem Schlüssel eine Erteilung hinzu, um anzugeben, wer den Schlüssel unter welchen Bedingungen verwenden kann. Erteilungen sind eine alternative Berechtigungsmethode zu Schlüsselrichtlinien. Weitere Informationen zu Grants finden Sie unter [Verwendung von Grants](#) im AWS Key Management Service -Entwicklerhandbuch. Diese Berechtigung ist in der Standard-Schlüsselrichtlinie enthalten.
- `kms:DescribeKey` – (Erforderlich) Stellt detaillierte Informationen über den angegebenen kundenverwalteten Schlüssel bereit. Diese Berechtigung ist in der Standard-Schlüsselrichtlinie enthalten.
- `kms:ListAliases` – (Optional) Listet alle Schlüsselaliasnamen 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 Amazon-EFS-KMS-Richtlinie

Das KMS-Richtlinien-JSON für die 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": "*"
    }
  ]
}

```

Verschlüsseln von Daten während der Übertragung

Sie können Daten während der Übertragung mit einem Amazon EFS-Dateisystem verschlüsseln, ohne dass Sie Ihre Anwendungen ändern müssen.

Datenverschlüsselung während der Übertragung mit TLS

Um die Verschlüsselung von Daten während der Übertragung für Ihr Amazon EFS-Dateisystem zu aktivieren, müssen Sie Transport Layer Security (TLS) aktivieren, wenn Sie Ihr Dateisystem mit der Amazon EFS-Mountinghilfe mounten. Weitere Informationen finden Sie unter [Verwenden der EFS-Mountinghilfe zum Mounten von EFS-Dateisystemen](#).

Wenn die Verschlüsselung von Daten während der Übertragung als Mountingoption für Ihr Amazon EFS-Dateisystem deklariert ist, initialisiert die Mountinghilfe einen Client-Stunnel-Vorgang. Stunnel ist ein Open-Source-Netzwerk-Relay für unterschiedliche Einsatzzwecke. Der Client-Stunnel-Vorgang überwacht einen lokalen Port auf eingehenden Datenverkehr und die Mountinghilfe leitet Network File System(NFS)-Client-Datenverkehr an diesen lokalen Port um. Die Mountinghilfe verwendet TLS Version 1.2 für die Kommunikation mit dem Dateisystem.

So mounten Sie Ihr Amazon EFS-Dateisystem mit der Mountinghilfe bei aktivierter Verschlüsselung der Daten während der Übertragung.

1. Greifen Sie über Secure Shell (SSH) auf das Terminal für die Instance zu und melden Sie sich mit dem entsprechenden Benutzernamen an. Weitere Informationen hierzu finden Sie unter [Herstellen einer Verbindung mit Ihrer Linux-Instance mit SSH](#) im Amazon EC2 Benutzerhandbuch für Linux-Instances.
2. Führen Sie den folgenden Befehl aus, um das Dateisystem zu mounten.

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

Funktionsweise der Verschlüsselung während der Übertragung

Um die Verschlüsselung von Daten während der Übertragung zu ermöglichen, stellen Sie eine Verbindung zu Amazon EFS über TLS her. Wir empfehlen die Verwendung der EFS-Mountinghilfe zum Mounten Ihres Dateisystems, da sie den Einhängenvorgang im Vergleich zum Einhängen mit NFS mount vereinfacht. Die EFS-Mountinghilfe verwaltet den Prozess mit stunnel für TLS. Sie können die Datenverschlüsselung während der Übertragung aber auch ohne die Mountinghilfe aktivieren. Gehen Sie dazu allgemein betrachtet wie folgt vor.

So aktivieren Sie die Verschlüsselung von Daten während der Übertragung, ohne die EFS-Mountinghilfe zu verwenden

1. Laden Sie `stunnel` herunter und installieren Sie es, und notieren Sie sich den Port, auf dem die Anwendung lauscht. Anweisungen dazu finden Sie unter [Upgraden von stunnel](#).
2. Führen Sie `stunnel` aus, um sich mit Ihrem Amazon EFS-Dateisystem an Port 2049 über TLS zu verbinden.
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 lauscht der `stunnel`-Prozess, der von der EFS-Mountinghilfe verwendet wird, an einem lokalen Port zwischen 20049 und 21049 und verbindet sich mit Amazon EFS an Port 2049.

 Note

Wenn Sie die Amazon EFS-Mountinghilfe mit TLS verwenden, erzwingt die Mountinghilfe standardmäßig die Überprüfung des Hostnamens des Zertifikats. Die Amazon EFS-Mountinghilfe verwendet das `stunnel`-Programm für die TLS-Funktionalität. In manchen Linux-Versionen ist keine `Stunnel`-Version enthalten, die diese TLS-Features standardmäßig unterstützt. Wenn Sie eine solche Linux-Version verwenden, können Sie ein Amazon EFS Dateisystem nicht mit TLS mounten.

Nachdem Sie das `amazon-efs-utils` Paket installiert haben, finden Sie Informationen zum Upgrade der `Stunnel`-Version Ihres Systems unter [Upgraden von stunnel](#).

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,rsize=1048576,wsiz=1048576,namlen=255,hard,proto=tcp,port=20127,timeo=600)
```

Bei der Einbindung mit TLS und der Amazon EFS-Mountinghilfe konfigurieren Sie Ihren NFS-Client so um, dass er an einem lokalen Port eingebunden 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. Ersetzen Sie *efs-mount-point* durch den lokalen Pfad, in dem Sie das Dateisystem gemountet haben.

```
grep -E "Successfully mounted.*efs-mount-point" /var/log/amazon/efs/mount.log | tail -1
```

Wenn Sie die Mountinghilfe zum Verschlüsseln von Daten während der Übertragung verwenden, wird auch ein Vorgang namens `amazon-efs-mount-watchdog` erstellt. Dieser Vorgang stellt sicher, dass der Stunnel-Vorgang für jedes Mounting läuft, und stoppt den Stunnel, wenn das Amazon EFS-Dateisystem ausgehängt wird. Wenn der Stunnel-Vorgang aus irgendeinem Grund unerwartet beendet wird, wird er vom Watchdog-Vorgang neu gestartet.

Identitäts- und Zugriffsmanagement für Amazon Elastic File System

AWS Identity and Access Management (IAM) ist ein AWS-Service, mit dem Administratoren den Zugriff auf AWS-Ressourcen sicher steuern können. IAM-Administratoren kontrollieren, wer authentifiziert (angemeldet) und autorisiert werden kann (über Berechtigungen verfügt), um Amazon EFS-Ressourcen zu verwenden. IAM ist ein AWS-Service, den Sie ohne zusätzliche Kosten verwenden können.

Themen

- [Zielgruppe](#)
- [Authentifizierung mit Identitäten](#)
- [Verwalten des Zugriffs mit Richtlinien](#)
- [So funktioniert Amazon Elastic File System mit IAM](#)
- [Beispiele für identitätsbasierte Richtlinien für Amazon Elastic File System](#)
- [Beispiele für ressourcenbasierte Richtlinien für Amazon Elastic File System](#)
- [AWSverwaltete Richtlinien für Amazon EFS](#)
- [Verwenden von Tags mit Amazon EFS](#)

- [Verwendung von serviceverknüpften Rollen für Amazon EFS](#)
- [Fehlerbehebung bei Identität und Zugriff auf Amazon Elastic File System](#)

Zielgruppe

Wie Sie AWS Identity and Access Management (IAM) verwenden, hängt von der Arbeit ab, die Sie in Amazon EFS ausführen.

Servicebenutzer — Wenn Sie den Amazon EFS-Service für Ihre Arbeit verwenden, stellt Ihnen Ihr Administrator die Anmeldeinformationen und Berechtigungen zur Verfügung, die Sie benötigen. Da Sie für Ihre Arbeit mehr Amazon EFS-Funktionen verwenden, benötigen Sie möglicherweise zusätzliche Berechtigungen. Wenn Sie die Featuresweise der Zugriffskontrolle nachvollziehen, wissen Sie bereits, welche Berechtigungen Sie von Ihrem Administrator anfordern müssen. Wenn Sie auf eine Funktion in Amazon EFS nicht zugreifen können, finden Sie weitere Informationen unter [Fehlerbehebung bei Identität und Zugriff auf Amazon Elastic File System](#).

Service-Administrator — Wenn Sie in Ihrem Unternehmen für die Amazon EFS-Ressourcen verantwortlich sind, haben Sie wahrscheinlich vollen Zugriff auf Amazon EFS. Es ist Ihre Aufgabe, zu bestimmen, auf welche Funktionen und Ressourcen von Amazon EFS Ihre Servicebenutzer zugreifen sollen. Sie müssen dann Anträge an Ihren IAM-Administrator stellen, um die Berechtigungen Ihrer Servicenutzer zu ändern. Lesen Sie die Informationen auf dieser Seite, um die Grundkonzepte von IAM nachzuvollziehen. Weitere Informationen darüber, wie Ihr Unternehmen IAM mit Amazon EFS verwenden kann, finden Sie unter [So funktioniert Amazon Elastic File System mit IAM](#).

IAM-Administrator — Wenn Sie ein IAM-Administrator sind, möchten Sie vielleicht mehr darüber erfahren, wie Sie Richtlinien schreiben können, um den Zugriff auf Amazon EFS zu verwalten. Beispiele für identitätsbasierte Amazon EFS-Richtlinien, die Sie in IAM verwenden können, finden Sie unter [Beispiele für identitätsbasierte Richtlinien für Amazon Elastic File System](#)

Authentifizierung mit Identitäten

Authentifizierung ist die Art, wie Sie sich mit Ihren Anmeldeinformationen bei AWS anmelden. Die Authentifizierung (Anmeldung bei AWS) muss als Root-Benutzer des AWS-Kontos, als IAM-Benutzer oder durch Übernahme einer IAM-Rolle erfolgen.

Sie können sich bei AWS als Verbundidentität mit Anmeldeinformationen anmelden, die über eine Identitätsquelle bereitgestellt werden. Benutzer von AWS IAM Identity Center. (IAM Identity Center), die Single-Sign-on-Authentifizierung Ihres Unternehmens und Anmeldeinformationen für Google oder Facebook sind Beispiele für Verbundidentitäten. Wenn Sie sich als Verbundidentität anmelden, hat

der Administrator vorher mithilfe von IAM-Rollen einen Identitätsverbund eingerichtet. Wenn Sie auf AWS mithilfe des Verbunds zugreifen, übernehmen Sie indirekt eine Rolle.

Je nachdem, welcher Benutzertyp Sie sind, können Sie sich bei der AWS Management Console oder beim AWS-Zugriffsportal anmelden. Weitere Informationen zum Anmelden bei AWS finden Sie unter [So melden Sie sich bei Ihrem AWS-Konto an](#) im Benutzerhandbuch von AWS-Anmeldung.

Bei programmgesteuerten Zugriff auf AWS bietet AWS ein Software Development Kit (SDK) und eine Command Line Interface (CLI, Befehlszeilenschnittstelle) zum kryptographischen Signieren Ihrer Anforderungen mit Ihren Anmeldeinformationen. Wenn Sie keine AWS-Tools verwenden, müssen Sie Anforderungen selbst signieren. Weitere Informationen zur Verwendung der empfohlenen Methode zum eigenen Signieren von Anforderungen finden Sie unter [Signieren von AWS-API-Anforderungen](#) im IAM-Benutzerhandbuch.

Unabhängig von der verwendeten Authentifizierungsmethode müssen Sie möglicherweise zusätzliche Sicherheitsinformationen angeben. AWS empfiehlt beispielsweise die Verwendung von Multi-Faktor Authentifizierung (MFA), um die Sicherheit Ihres Kontos zu verbessern. Weitere Informationen finden Sie unter [Multi-Faktor-Authentifizierung](#) im AWS IAM Identity Center-Benutzerhandbuch und [Verwenden der Multi-Faktor-Authentifizierung \(MFA\) in AWS](#) im IAM-Benutzerhandbuch.

AWS-Konto-Root-Benutzer

Wenn Sie ein AWS-Konto neu erstellen, beginnen Sie mit einer Anmeldeidentität, die vollständigen Zugriff auf alle AWS-Services und Ressourcen des Kontos hat. Diese Identität wird als AWS-Konto-Root-Benutzer bezeichnet. Für den Zugriff auf den Root-Benutzer müssen Sie sich mit der E-Mail-Adresse und dem Passwort anmelden, die zur Erstellung des Kontos verwendet wurden. Wir raten ausdrücklich davon ab, den Root-Benutzer für Alltagsaufgaben zu verwenden. Schützen Sie Ihre Root-Benutzer-Anmeldeinformationen und verwenden Sie diese, um die Aufgaben auszuführen, die nur der Root-Benutzer ausführen kann. Eine vollständige Liste der Aufgaben, für die Sie sich als Root-Benutzer anmelden müssen, finden Sie unter [Aufgaben, die Root-Benutzer-Anmeldeinformationen erfordern](#) im IAM-Benutzerhandbuch.

Verbundidentität

Als bewährte Methode empfiehlt es sich, menschliche Benutzer, einschließlich Benutzer, die Administratorzugriff benötigen, aufzufordern, den Verbund mit einem Identitätsanbieter zu verwenden, um auf AWS-Services mit temporären Anmeldeinformationen zuzugreifen.

Eine Verbundidentität ist ein Benutzer aus dem Benutzerverzeichnis Ihres Unternehmens, ein Web Identity Provider, AWS Directory Service, das Identity-Center-Verzeichnis oder jeder Benutzer, der mit Anmeldeinformationen, die über eine Identitätsquelle bereitgestellt werden, auf AWS-Services zugreift. Wenn Verbundidentitäten auf AWS-Konten zugreifen, übernehmen sie Rollen und die Rollen stellen temporäre Anmeldeinformationen bereit.

Für die zentrale Zugriffsverwaltung empfehlen wir Ihnen, AWS IAM Identity Center zu verwenden. Sie können Benutzer und Gruppen im IAM Identity Center erstellen oder Sie können eine Verbindung mit einer Gruppe von Benutzern und Gruppen in Ihrer eigenen Identitätsquelle herstellen und synchronisieren, um sie in allen AWS-Konten und Anwendungen zu verwenden. Informationen zu IAM Identity Center finden Sie unter [Was ist IAM Identity Center?](#) im AWS IAM Identity Center-Benutzerhandbuch.

IAM-Benutzer und -Gruppen

Ein [IAM-Benutzer](#) ist eine Identität in Ihrem AWS-Konto mit bestimmten Berechtigungen für eine einzelne Person oder eine einzelne Anwendung. Wenn möglich, empfehlen wir, temporäre Anmeldeinformationen zu verwenden, anstatt IAM-Benutzer zu erstellen, die langfristige Anmeldeinformationen wie Passwörter und Zugriffsschlüssel haben. Bei speziellen Anwendungsfällen, die langfristige Anmeldeinformationen mit IAM-Benutzern erfordern, empfehlen wir jedoch, die Zugriffsschlüssel zu rotieren. Weitere Informationen finden Sie unter [Regelmäßiges Rotieren von Zugriffsschlüsseln für Anwendungsfälle, die langfristige Anmeldeinformationen erfordern](#) im IAM-Benutzerhandbuch.

Eine [IAM-Gruppe](#) ist eine Identität, die eine Sammlung von IAM-Benutzern angibt. Sie können sich nicht als Gruppe anmelden. Mithilfe von Gruppen können Sie Berechtigungen für mehrere Benutzer gleichzeitig angeben. Gruppen vereinfachen die Verwaltung von Berechtigungen, wenn es zahlreiche Benutzer gibt. Sie könnten beispielsweise einer Gruppe mit dem Namen IAMAdmins Berechtigungen zum Verwalten von IAM-Ressourcen erteilen.

Benutzer unterscheiden sich von Rollen. Ein Benutzer ist einer einzigen Person oder Anwendung eindeutig zugeordnet. Eine Rolle kann von allen Personen angenommen werden, die sie benötigen. Benutzer besitzen dauerhafte Anmeldeinformationen. Rollen stellen temporäre Anmeldeinformationen bereit. Weitere Informationen finden Sie unter [Erstellen eines IAM-Benutzers \(anstatt einer Rolle\)](#) im IAM-Benutzerhandbuch.

IAM-Rollen

Eine [IAM-Rolle](#) ist eine Identität in Ihrem AWS-Konto mit spezifischen Berechtigungen. Sie ist einem IAM-Benutzer vergleichbar, ist aber nicht mit einer bestimmten Person verknüpft. Sie können vorübergehend eine IAM-Rolle in der AWS Management Console übernehmen, indem Sie [Rollen wechseln](#). Sie können eine Rolle annehmen, indem Sie eine AWS CLI oder AWS-API-Operation aufrufen oder eine benutzerdefinierte URL verwenden. Weitere Informationen zu Methoden für die Verwendung von Rollen finden Sie unter [Verwenden von IAM-Rollen](#) im IAM-Benutzerhandbuch.

IAM-Rollen mit temporären Anmeldeinformationen sind in folgenden Situationen hilfreich:

- **Verbundbenutzerzugriff** – Um einer Verbundidentität Berechtigungen zuzuweisen, erstellen Sie eine Rolle und definieren Berechtigungen für die Rolle. Wird eine Verbundidentität authentifiziert, so wird die Identität der Rolle zugeordnet und erhält die von der Rolle definierten Berechtigungen. Informationen zu Rollen für den Verbund finden Sie unter [Erstellen von Rollen für externe Identitätsanbieter](#) im IAM-Benutzerhandbuch. Wenn Sie IAM Identity Center verwenden, konfigurieren Sie einen Berechtigungssatz. Wenn Sie steuern möchten, worauf Ihre Identitäten nach der Authentifizierung zugreifen können, korreliert IAM Identity Center den Berechtigungssatz mit einer Rolle in IAM. Informationen zu Berechtigungssätzen finden Sie unter [Berechtigungssätze](#) im AWS IAM Identity Center-Benutzerhandbuch.
- **Temporäre IAM-Benutzerberechtigungen** – Ein IAM-Benutzer oder eine -Rolle kann eine IAM-Rolle übernehmen, um vorübergehend andere Berechtigungen für eine bestimmte Aufgabe zu erhalten.
- **Kontoübergreifender Zugriff** – Sie können eine IAM-Rolle verwenden, um einem vertrauenswürdigen Prinzipal in einem anderen Konto den Zugriff auf Ressourcen in Ihrem Konto zu ermöglichen. Rollen stellen die primäre Möglichkeit dar, um kontoübergreifendem Zugriff zu gewähren. In einigen AWS-Services können Sie jedoch eine Richtlinie direkt an eine Ressource anfügen (anstatt eine Rolle als Proxy zu verwenden). Informationen zu den Unterschieden zwischen Rollen und ressourcenbasierten Richtlinien für den kontoübergreifenden Zugriff finden Sie unter [So unterscheiden sich IAM-Rollen von ressourcenbasierten Richtlinien](#) im IAM-Benutzerhandbuch.
- **Serviceübergreifender Zugriff** – Einige AWS-Services verwenden Features in anderen AWS-Services. Wenn Sie beispielsweise einen Aufruf in einem Service tätigen, führt dieser Service häufig Anwendungen in Amazon EC2 aus oder speichert Objekte in Amazon S3. Ein Dienst kann dies mit den Berechtigungen des aufrufenden Prinzipals mit einer Servicerolle oder mit einer serviceverknüpften Rolle tun.
 - **Forward access sessions (FAS)** – Wenn Sie einen IAM-Benutzer oder eine IAM-Rolle zum Ausführen von Aktionen in AWS verwenden, gelten Sie als Prinzipal. Bei einigen Services

könnte es Aktionen geben, die dann eine andere Aktion in einem anderen Service initiieren. FAS verwendet die Berechtigungen des Prinzipals, der einen AWS-Service aufruft, in Kombination mit der Anforderung an den AWS-Service, Anforderungen an nachgelagerte Services zu stellen. FAS-Anforderungen werden nur dann gestellt, wenn ein Dienst eine Anforderung erhält, die Interaktionen mit anderen AWS-Services oder Ressourcen erfordert, um abgeschlossen werden zu können. In diesem Fall müssen Sie über Berechtigungen zum Ausführen beider Aktionen verfügen. Einzelheiten zu den Richtlinien für FAS-Anfragen finden Sie unter [Zugriffssitzungen weiterleiten](#).

- **Servicerolle** – Eine Servicerolle ist eine [IAM-Rolle](#), die ein Service übernimmt, 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.
- **Serviceverknüpfte Rolle** – Eine serviceverknüpfte Rolle ist ein Typ von Servicerolle, die mit einem AWS-Service verknüpft ist. Der Service kann die Rolle übernehmen, um eine Aktion in Ihrem Namen auszuführen. Serviceverknüpfte Rollen werden in Ihrem AWS-Konto angezeigt und gehören zum Service. Ein IAM-Administrator kann die Berechtigungen für serviceverbundene Rollen anzeigen, aber nicht bearbeiten.
- **Anwendungen in Amazon EC2** – Sie können eine IAM-Rolle verwenden, um temporäre Anmeldeinformationen für Anwendungen zu verwalten, die auf einer EC2-Instance ausgeführt werden und AWS CLI- oder AWS-API-Anforderungen durchführen. Das ist eher zu empfehlen, als Zugriffsschlüssel innerhalb der EC2-Instance zu speichern. Erstellen Sie ein Instance-Profil, das an die Instance angefügt ist, um eine AWS-Rolle einer EC2-Instance zuzuweisen und die Rolle für sämtliche Anwendungen der Instance bereitzustellen. Ein Instance-Profil enthält die Rolle und ermöglicht, dass Programme, die in der EC2-Instance ausgeführt werden, temporäre Anmeldeinformationen erhalten. Weitere Informationen finden Sie unter [Verwenden einer IAM-Rolle zum Erteilen von Berechtigungen für Anwendungen, die auf Amazon EC2-Instances ausgeführt werden](#) im IAM-Benutzerhandbuch.

Informationen dazu, wann Sie IAM-Rollen oder IAM-Benutzer verwenden sollten, finden Sie unter [Erstellen einer IAM-Rolle \(anstatt eines Benutzers\)](#) im IAM-Benutzerhandbuch.

Verwalten des Zugriffs mit Richtlinien

Für die Zugriffssteuerung in AWS erstellen Sie Richtlinien und weisen diese den AWS-Identitäten oder -Ressourcen zu. Eine Richtlinie ist ein Objekt in AWS, das, wenn es einer Identität oder Ressource zugeordnet wird, deren Berechtigungen definiert. AWS wertet diese Richtlinien aus, wenn

ein Prinzipal (Benutzer, Root-Benutzer oder Rollensitzung) eine Anforderung stellt. Berechtigungen in den Richtlinien bestimmen, ob die Anforderung zugelassen oder abgelehnt wird. Die meisten Richtlinien werden in AWS als JSON-Dokumente gespeichert. Weitere Informationen zu Struktur und Inhalten von JSON-Richtliniendokumenten finden Sie unter [Übersicht über JSON-Richtlinien](#) im IAM-Benutzerhandbuch.

Administratoren können mithilfe von AWS-JSON-Richtlinien festlegen, wer zum Zugriff auf was berechtigt ist. Das bedeutet, welcher Prinzipal kann Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen.

Standardmäßig haben Benutzer, Gruppen und Rollen keine Berechtigungen. Ein IAM-Administrator muss IAM-Richtlinien erstellen, die Benutzern die Berechtigung erteilen, Aktionen für die Ressourcen auszuführen, die sie benötigen. Der Administrator kann dann die IAM-Richtlinien zu Rollen hinzufügen, und Benutzer können die Rollen annehmen.

IAM-Richtlinien definieren Berechtigungen für eine Aktion unabhängig von der Methode, die Sie zur Ausführung der Aktion verwenden. Angenommen, es gibt eine Richtlinie, die Berechtigungen für die `iam:GetRole`-Aktion erteilt. Ein Benutzer mit dieser Richtlinie kann Benutzerinformationen über die AWS Management Console, die AWS CLI oder die AWS -API abrufen.

Identitätsbasierte Richtlinien

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 [Erstellen von IAM-Richtlinien](#) im IAM-Benutzerhandbuch.

Identitätsbasierte Richtlinien können weiter als Inline-Richtlinien oder verwaltete Richtlinien kategorisiert werden. Inline-Richtlinien sind direkt in einen einzelnen Benutzer, eine einzelne Gruppe oder eine einzelne Rolle eingebettet. Verwaltete Richtlinien sind eigenständige Richtlinien, die Sie mehreren Benutzern, Gruppen und Rollen in Ihrem AWS-Konto anfügen können. Verwaltete Richtlinien umfassen von AWS verwaltete und von Kunden verwaltete Richtlinien. Informationen dazu, wie Sie zwischen einer verwalteten Richtlinie und einer eingebundenen Richtlinie wählen, finden Sie unter [Auswahl zwischen verwalteten und eingebundenen Richtlinien](#) im IAM-Benutzerhandbuch.

Ressourcenbasierte Richtlinien

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](#). Prinzipale können Konten, Benutzer, Rollen, Verbundbenutzer oder AWS-Services umfassen.

Ressourcenbasierte Richtlinien sind Richtlinien innerhalb dieses Diensts. Sie können verwaltete AWS-Richtlinien von IAM nicht in einer ressourcenbasierten Richtlinie verwenden.

Zugriffssteuerungslisten (ACLs)

Zugriffssteuerungslisten (ACLs) steuern, welche Prinzipale (Kontomitglieder, Benutzer oder Rollen) auf eine Ressource zugreifen können. ACLs sind ähnlich wie ressourcenbasierte Richtlinien, verwenden jedoch nicht das JSON-Richtliniendokumentformat.

Amazon S3, AWS WAF und Amazon VPC sind Beispiele für Dienste, die ACLs unterstützen. Weitere Informationen zu ACLs finden Sie unter [Zugriffskontrollliste \(ACL\) – Übersicht](#) (Access Control List) im Amazon-Simple-Storage-Service-Entwicklerhandbuch.

Weitere Richtlinientypen

AWS unterstützt zusätzliche, weniger häufig verwendete Richtlinientypen. Diese Richtlinientypen können die maximalen Berechtigungen festlegen, die Ihnen von den häufiger verwendeten Richtlinientypen erteilt werden können.

- **Berechtigungsgrenzen** – Eine Berechtigungsgrenze ist ein erweitertes Feature, mit der Sie die maximalen Berechtigungen festlegen können, die eine identitätsbasierte Richtlinie einer IAM-Entität (IAM-Benutzer oder -Rolle) erteilen kann. Sie können eine Berechtigungsgrenze für eine Entität festlegen. Die daraus resultierenden Berechtigungen sind der Schnittpunkt der identitätsbasierten Richtlinien einer Entität und ihrer Berechtigungsgrenzen. Ressourcenbasierte Richtlinien, die den Benutzer oder die Rolle im Feld `Principal` angeben, werden nicht durch Berechtigungsgrenzen eingeschränkt. Ein ausdrückliches Ablehnen in einer dieser Richtlinien setzt das Zulassen außer Kraft. Weitere Informationen über Berechtigungsgrenzen finden Sie unter [Berechtigungsgrenzen für IAM-Entitäten](#) im IAM-Benutzerhandbuch.

- **Service-Kontrollrichtlinien (SCPs)** – SCPs sind JSON-Richtlinien, die die maximalen Berechtigungen für eine Organisation oder Organisationseinheit (OE) in AWS Organizations angeben. AWS Organizations ist ein Dienst für die Gruppierung und zentrale Verwaltung mehrerer AWS-Konten Ihres Unternehmens. Wenn Sie innerhalb einer Organisation alle Features aktivieren, können Sie Service-Kontrollrichtlinien (SCPs) auf alle oder einzelne Ihrer Konten anwenden. SCPs schränken Berechtigungen für Entitäten in Mitgliedskonten einschließlich des jeweiligen Root-Benutzer des AWS-Kontos ein. Weitere Informationen zu Organisationen und SCPs finden Sie unter [Funktionsweise von SCPs](#) im AWS Organizations-Benutzerhandbuch.
- **Sitzungsrichtlinien** – Sitzungsrichtlinien sind erweiterte Richtlinien, die Sie als Parameter übergeben, wenn Sie eine temporäre Sitzung für eine Rolle oder einen verbundenen Benutzer programmgesteuert erstellen. Die resultierenden Sitzungsberechtigungen sind eine Schnittmenge der auf der Identität des Benutzers oder der Rolle basierenden Richtlinien und der Sitzungsrichtlinien. Berechtigungen können auch aus einer ressourcenbasierten Richtlinie stammen. Eine explizite Zugriffsverweigerung in einer dieser Richtlinien setzt eine Zugriffserlaubnis außer Kraft. Weitere Informationen finden Sie unter [Sitzungsrichtlinien](#) im IAM-Benutzerhandbuch.

Mehrere Richtlinientypen

Wenn mehrere auf eine Anforderung mehrere Richtlinientypen angewendet werden können, sind die entsprechenden Berechtigungen komplizierter. Informationen dazu, wie AWS die Zulässigkeit einer Anforderung ermittelt, wenn mehrere Richtlinientypen beteiligt sind, finden Sie unter [Logik für die Richtlinienauswertung](#) im IAM-Benutzerhandbuch.

So funktioniert Amazon Elastic File System mit IAM

Bevor Sie IAM verwenden, um den Zugriff auf Amazon EFS zu verwalten, sollten Sie sich darüber informieren, welche IAM-Funktionen mit Amazon EFS verwendet werden können.

IAM-Funktionen, die Sie mit Amazon Elastic File System verwenden können

IAM-Funktion	Amazon EFS-Unterstützung
Identitätsbasierte Richtlinien	Ja
Ressourcenbasierte Richtlinien	Ja
Richtlinienaktionen	Ja

IAM-Funktion	Amazon EFS-Unterstützung
Richtlinienressourcen	Ja
Richtlinienbedingungsschlüssel (servicespezifisch)	Ja
ACLs	Nein
ABAC (Tags in Richtlinien)	Teilweise
Temporäre Anmeldeinformationen	Ja
Hauptberechtigungen	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 [Erstellen von IAM-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. Sie können den Prinzipal nicht in einer identitätsbasierten Richtlinie angeben, da er für den Benutzer oder die Rolle gilt, dem er zugeordnet

ist. 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 innerhalb von 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](#). Prinzipale können Konten, Benutzer, Rollen, Verbundbenutzer oder AWS-Services umfassen.

Um kontoübergreifenden Zugriff zu ermöglichen, können Sie ein gesamtes Konto oder IAM-Entitäten in einem anderen Konto als Prinzipal in einer ressourcenbasierten Richtlinie angeben. Durch das Hinzufügen eines kontoübergreifenden Auftraggebers zu einer ressourcenbasierten Richtlinie ist nur die halbe Vertrauensbeziehung eingerichtet. Wenn sich der Prinzipal und die Ressource in unterschiedlichen AWS-Konten befinden, muss ein IAM-Administrator im vertrauenswürdigen Konto auch der Prinzipalentität (Benutzer oder Rolle) die Berechtigung zum Zugriff auf die Ressource erteilen. Sie erteilen Berechtigungen, indem Sie der juristischen Stelle eine identitätsbasierte Richtlinie anfügen. Wenn jedoch eine ressourcenbasierte Richtlinie Zugriff auf einen Prinzipal in demselben Konto gewährt, ist keine zusätzliche identitätsbasierte Richtlinie erforderlich.

Weitere Informationen finden Sie unter [Wie sich IAM-Rollen von ressourcenbasierten Richtlinien unterscheiden](#) im IAM-Benutzerhandbuch.

Weitere Informationen zur Verwendung einer Ressourcenrichtlinie zur Steuerung des Datenzugriffs auf Dateisysteme finden Sie unter [Verwendung von IAM zur Steuerung des Dateisystemdatenzugriffs](#). Informationen zum Anhängen einer ressourcenbasierten Richtlinie an ein Dateisystem finden Sie unter [Erstellen von Dateisystemrichtlinien](#)

Beispiele für ressourcenbasierte Richtlinien in Amazon EFS

Beispiele für ressourcenbasierte Amazon EFS-Richtlinien finden Sie unter [Beispiele für ressourcenbasierte Richtlinien für Amazon Elastic File System](#)

Richtlinienaktionen für Amazon EFS

Unterstützt Richtlinienaktionen	Ja
---------------------------------	----

Administratoren können mithilfe von AWS-JSON-Richtlinien festlegen, wer zum Zugriff auf was berechtigt ist. Das heißt, welcher Prinzipal kann Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen.

Das Element `Action` einer JSON-Richtlinie beschreibt die Aktionen, mit denen Sie den Zugriff in einer Richtlinie zulassen oder verweigern können. Richtlinienaktionen haben normalerweise denselben Namen wie die zugehörige AWS-API-Operation. Es gibt einige Ausnahmen, z. B. Aktionen, die nur mit Genehmigung durchgeführt werden können und für die es keine passende API-Operation gibt. Es gibt auch einige Operationen, die mehrere Aktionen in einer Richtlinie erfordern. Diese zusätzlichen Aktionen werden als abhängige Aktionen bezeichnet.

Schließen Sie Aktionen in eine Richtlinie ein, um Berechtigungen zur Durchführung der zugeordneten Operation zu erteilen.

Eine Liste der Amazon EFS-Aktionen finden Sie unter [Von Amazon Elastic File System definierte Aktionen](#) in der Service Authorization Reference.

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 festlegen, wer zum Zugriff auf was berechtigt ist. Das bedeutet die Festlegung, welcher Prinzipal Aktionen für welche Ressourcen unter welchen Bedingungen ausführen kann.

Das JSON-Richtlinienelement `Resource` gibt die Objekte an, auf welche die Aktion angewendet wird. Anweisungen müssen entweder ein `Resource` oder ein `NotResource`-Element enthalten. Als bewährte Methode geben Sie eine Ressource mit dem zugehörigen [Amazon-Ressourcennamen \(ARN\)](#) an. Sie können dies für Aktionen tun, die einen bestimmten Ressourcentyp unterstützen, der als Berechtigungen auf Ressourcenebene bezeichnet wird.

Verwenden Sie für Aktionen, die keine Berechtigungen auf Ressourcenebene unterstützen, z. B. Auflistungsoperationen, einen Platzhalter (*), um anzugeben, dass die Anweisung für alle Ressourcen gilt.

```
"Resource": "*"

```

Eine Liste der Amazon EFS-Ressourcentypen und ihrer ARNs finden Sie unter [Von Amazon Elastic File System definierte Ressourcen](#) in der Service Authorization Reference. Informationen darüber, mit welchen Aktionen Sie den ARN jeder Ressource angeben 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](#)

Schlüssel für Richtlinienbedingungen für Amazon EFS

Unterstützt servicespezifische Richtlinienbedingungsschlüssel	Ja
---	----

Administratoren können mithilfe von AWS-JSON-Richtlinien festlegen, wer zum Zugriff auf was berechtigt ist. Das heißt, welcher Prinzipal kann Aktionen für welche Ressourcen und unter welchen Bedingungen ausführen.

Das Element `Condition` (oder `Condition block`) ermöglicht Ihnen die Angabe der Bedingungen, unter denen eine Anweisung wirksam ist. Das Element `Condition` ist optional. 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.

Wenn Sie mehrere `Condition`-Elemente in einer Anweisung oder mehrere Schlüssel in einem einzelnen `Condition`-Element angeben, wertet AWS diese mittels einer logischen AND-Operation aus. Wenn Sie mehrere Werte für einen einzelnen Bedingungsschlüssel angeben, wertet AWS die Bedingung mittels einer logischen OR-Operation aus. Alle Bedingungen müssen erfüllt werden, bevor die Berechtigungen der Anweisung gewährt werden.

Sie können auch Platzhaltervariablen verwenden, wenn Sie Bedingungen angeben. Beispielsweise können Sie einem IAM-Benutzer die Berechtigung für den Zugriff auf eine Ressource nur dann gewähren, wenn sie mit dessen IAM-Benutzernamen gekennzeichnet ist. Weitere Informationen finden Sie unter [IAM-Richtlinienelemente: Variablen und Tags](#) im IAM-Benutzerhandbuch.

AWS unterstützt globale Bedingungsschlüssel und servicespezifische Bedingungsschlüssel. Eine Liste aller globalen AWS-Bedingungsschlüssel finden Sie unter [Globale AWS-Bedingungskontextschlüssel](#) im IAM-Benutzerhandbuch.

Eine Liste der Amazon EFS-Bedingungsschlüssel finden Sie unter [Bedingungsschlüssel für Amazon Elastic File System](#) in der Service Authorization Reference. Informationen zu den Aktionen und Ressourcen, mit denen 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

Zugriffssteuerungslisten (ACLs) steuern, welche Prinzipale (Kontomitglieder, Benutzer oder Rollen) auf eine Ressource zugreifen können. ACLs sind ähnlich wie ressourcenbasierte 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 definiert werden. In AWS werden diese Attribute als Tags bezeichnet. Sie können Tags an IAM-Entitäten (Benutzer oder Rollen) und mehrere AWS-Ressourcen anfügen. Das Markieren von Entitäten und Ressourcen ist der erste Schritt von ABAC. Anschließend entwerfen Sie ABAC-Richtlinien, um Operationen zuzulassen, wenn das Tag des Prinzipals mit dem Tag der Ressource übereinstimmt, auf die sie zugreifen möchten.

ABAC ist in Umgebungen hilfreich, die schnell wachsen, und unterstützt Sie in Situationen, in denen die Richtlinienverwaltung mühsam wird.

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 `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 [Was ist ABAC?](#) im IAM-Benutzerhandbuch. Um ein Tutorial mit Schritten zur Einstellung von ABAC anzuzeigen, siehe [Attributbasierte Zugriffskontrolle \(ABAC\)](#) verwenden im IAM-Benutzerhandbuch.

Temporäre Anmeldeinformationen mit Amazon EFS verwenden

Unterstützt temporäre Anmeldeinformationen

Ja

Einige AWS-Services Featureieren nicht, wenn Sie sich mit temporären Anmeldeinformationen anmelden. Weitere Informationen, darunter welche AWS-Services mit temporären Anmeldeinformationen Featureieren, finden Sie unter [AWS-Services, die mit IAM Featureieren](#) im IAM-Benutzerhandbuch.

Sie verwenden temporäre Anmeldeinformationen, wenn Sie sich mit einer anderen Methode als einem Benutzernamen und einem Passwort bei der AWS Management Console anmelden. Wenn

Sie beispielsweise über den Single Sign-On (SSO)-Link Ihres Unternehmens auf AWS zugreifen, erstellt dieser Prozess automatisch temporäre Anmeldeinformationen. Sie erstellen auch automatisch temporäre Anmeldeinformationen, wenn Sie sich als Benutzer bei der Konsole anmelden und dann die Rollen wechseln. Weitere Informationen zum Wechseln von Rollen finden Sie unter [Wechseln zu einer Rolle \(Konsole\)](#) im IAM-Benutzerhandbuch.

Sie können mithilfe der AWS CLI- oder AWS-API manuell temporäre Anmeldeinformationen erstellen. Sie können dann diese temporären Anmeldeinformationen verwenden, um auf AWS zuzugreifen. AWS empfiehlt, dass Sie temporäre Anmeldeinformationen dynamisch generieren, anstatt langfristige Zugriffsschlüssel zu verwenden. Weitere Informationen finden Sie unter [Temporäre Sicherheitsanmeldeinformationen in IAM](#).

Serviceübergreifende Prinzipalberechtigungen für Amazon EFS

Unterstützt Forward Access Sessions (FAS)	Ja
---	----

Wenn Sie einen IAM-Benutzer oder eine IAM-Rolle zum Ausführen von Aktionen in AWS verwenden, gelten Sie als Prinzipal. Bei einigen Services könnte es Aktionen geben, die dann eine andere Aktion in einem anderen Service auslösen. FAS verwendet die Berechtigungen des Prinzipals, der einen AWS-Service aufruft, in Kombination mit der Anforderung an den AWS-Service, Anforderungen an nachgelagerte Services zu stellen. FAS-Anforderungen werden nur dann gestellt, wenn ein Dienst eine Anforderung erhält, die Interaktionen mit anderen AWS-Services oder Ressourcen erfordert, um abgeschlossen werden zu können. In diesem Fall müssen Sie über Berechtigungen zum Ausführen beider Aktionen verfügen. Einzelheiten zu den Richtlinien für FAS-Anfragen 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**

Durch das Ändern der Berechtigungen für eine Servicerolle kann die Amazon EFS-Funktionalität beeinträchtigt werden. Bearbeiten Sie Servicerollen nur, wenn Amazon EFS Sie dazu anleitet.

Servicebezogene Rollen für Amazon EFS

Unterstützt serviceverknüpfte Rollen	Ja
--------------------------------------	----

Eine serviceverknüpfte Rolle ist eine Art von Servicerolle, die mit einem AWS-Service verknüpft ist. Der Service kann die Rolle übernehmen, um eine Aktion in Ihrem Namen auszuführen. Serviceverknüpfte Rollen werden in Ihrem AWS-Konto angezeigt und gehören zum Service. Ein IAM-Administrator kann die Berechtigungen für serviceverbundene Rollen anzeigen, aber nicht bearbeiten.

Einzelheiten zum Erstellen oder Verwalten von serviceverknüpften Amazon EFS-Rollen finden Sie unter [Verwendung von serviceverknüpften Rollen für Amazon EFS](#).

Beispiele für identitätsbasierte Richtlinien für Amazon Elastic File System

Standardmäßig sind Benutzer und Rollen nicht berechtigt, Amazon EFS-Ressourcen zu erstellen oder zu ändern. Sie können auch keine Aufgaben über die AWS Management Console, die AWS Command Line Interface (AWS CLI) oder die AWS-API ausführen. Ein IAM-Administrator muss IAM-Richtlinien erstellen, die Benutzern die Berechtigung erteilen, Aktionen für die Ressourcen auszuführen, die sie benötigen. Der Administrator kann dann die IAM-Richtlinien zu Rollen hinzufügen, und Benutzer können die Rollen annehmen.

Informationen dazu, wie Sie unter Verwendung dieser beispielhaften JSON-Richtliniendokumente eine identitätsbasierte IAM-Richtlinie erstellen, finden Sie unter [Erstellen von IAM-Richtlinien](#) im IAM-Benutzerhandbuch.

Einzelheiten zu den von Amazon EFS definierten Aktionen und Ressourcentypen, einschließlich des Formats der ARNs für jeden Ressourcentyp, finden Sie unter [Aktionen, Ressourcen und Bedingungsschlüssel für Amazon Elastic File System](#) in der Service Authorization Reference.

Themen

- [Bewährte Methoden für Richtlinien](#)
- [Verwenden der Amazon EFS-Konsole](#)
- [Beispiel: Erlauben Sie Benutzern, ihre eigenen Berechtigungen einzusehen](#)
- [Beispiel: Erzwingen Sie die Erstellung verschlüsselter Dateisysteme](#)
- [Beispiel: Erzwingen Sie die Erstellung unverschlüsselter Dateisysteme](#)

Bewährte Methoden für Richtlinien

Identitätsbasierte Richtlinien legen fest, ob jemand Amazon EFS-Ressourcen in Ihrem Konto erstellen, darauf zugreifen oder diese löschen kann. Dies kann zusätzliche Kosten für Ihr verursachen AWS-Konto. Befolgen Sie beim Erstellen oder Bearbeiten identitätsbasierter Richtlinien die folgenden Anleitungen und Empfehlungen:

- Erste Schritte mit AWS-verwaltete Richtlinien und Umstellung auf Berechtigungen mit den geringsten Berechtigungen – Um Ihren Benutzern und Workloads Berechtigungen zu gewähren, verwenden Sie die AWS-verwaltete Richtlinien die Berechtigungen für viele allgemeine Anwendungsfälle gewähren. Sie sind in Ihrem AWS-Konto verfügbar. Wir empfehlen Ihnen, die Berechtigungen weiter zu reduzieren, indem Sie AWS-kundenverwaltete Richtlinien definieren, die speziell auf Ihre Anwendungsfälle zugeschnitten sind. Weitere Informationen finden Sie unter [AWS-verwaltete Richtlinien](#) oder [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 Service-Aktionen zu gewähren, wenn diese durch ein bestimmtes AWS-Service, wie beispielsweise AWS CloudFormation, verwendet werden. 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 zum IAM Access Analyzer](#) im IAM-Benutzerhandbuch.
- Bedarf einer Multi-Faktor-Authentifizierung (MFA) – Wenn Sie ein Szenario haben, das IAM-Benutzer oder Root-Benutzer in Ihrem AWS-Konto erfordert, 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 [Konfigurieren eines MFA-geschützten API-Zugriffs](#) im IAM-Benutzerhandbuch.

Weitere Informationen zu bewährten Methoden in IAM finden Sie unter [Bewährte Methoden für die Sicherheit in IAM](#) im IAM-Benutzerhandbuch.

Verwenden der Amazon EFS-Konsole

Um auf die Amazon Elastic File System-Konsole zugreifen zu können, benötigen Sie ein Mindestmaß an Berechtigungen. 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.

Für Benutzer, die nur Aufrufe an die AWS CLI oder AWS-API durchführen, müssen Sie keine Mindestberechtigungen in der Konsole erteilen. 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: Erlauben Sie Benutzern, ihre eigenen Berechtigungen einzusehen

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 enthält Berechtigungen für die Ausführung dieser Aktion auf der Konsole oder für die programmgesteuerte Ausführung über die AWS CLI oder die AWS-API.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ViewOwnUserInfo",
      "Effect": "Allow",
      "Action": [
        "iam:GetUserPolicy",
        "iam:ListGroupsWithUser",
        "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 Sie die Erstellung verschlüsselter Dateisysteme

Das folgende Beispiel zeigt eine identitätsbasierte Richtlinie, die Principals 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 Management Console, die oder die AWS CLI, 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.
```

Beispiel: Erzwingen Sie die Erstellung unverschlüsselter Dateisysteme

Das folgende Beispiel veranschaulicht eine identitätsbasierte Richtlinie, die Principals 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 Management Console, die AWS CLI, 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 Amazon EFS-Dateisysteme erzwingen, indem Sie eine AWS Organizations Service Control Policy (SCP) erstellen. Weitere Informationen zu Service-Control-Richtlinien finden Sie unter [Service-Control-Richtlinien](#) im AWS Organizations Benutzerhandbuch. AWS Organizations

Beispiele für ressourcenbasierte Richtlinien für Amazon Elastic File System

In diesem Abschnitt finden Sie Beispiele für Dateisystemrichtlinien, die Berechtigungen für verschiedene Amazon EFS-Aktionen gewähren oder verweigern. Die Amazon EFS-Dateisystemrichtlinien haben ein Limit von 20.000 Zeichen. Hinweise zu den Elementen einer ressourcenbasierten Richtlinie finden Sie unter [Ressourcenbasierte Richtlinien innerhalb von Amazon EFS](#).

Important

Wenn Sie einem einzelnen IAM-Benutzer oder einer einzelnen IAM-Rolle in einer Dateisystemrichtlinie die Erlaubnis erteilen, dürfen Sie diesen Benutzer oder diese Rolle nicht löschen oder neu erstellen, solange die Richtlinie im 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 einer Dateisystemrichtlinie finden Sie unter. [Erstellen von Dateisystemrichtlinien](#)

Themen

- [Beispiel: Erteilen Sie einer bestimmten AWS Rolle Lese- und Schreibzugriff](#)
- [Beispiel: Nur-Lese-Zugriff gewähren](#)
- [Beispiel: Zugriff auf einen EFS Access Point gewähren](#)

Beispiel: Erteilen Sie einer bestimmten AWS Rolle Lese- und Schreibzugriff

In diesem Beispiel weist die EFS-Dateisystemrichtlinie die folgenden Merkmale auf:

- Der Effekt ist Allow.
- Der Principal ist auf die Testing_Role in der festgelegt. AWS-Konto
- Die Aktion ist auf ClientMount (gelesen) und gesetzt. ClientWrite
- Die Bedingung für die Erteilung von Berechtigungen ist auf gesetztAccessedViaMountTarget.

```
{
  "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: Nur-Lese-Zugriff gewähren

Die folgende Dateisystemrichtlinie gewährt der ClientMount IAM-Rolle nur oder nur Leseberechtigungen. EfsReadOnly

```
{
  "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"
    }
  ]
}
```

Informationen zum Einrichten zusätzlicher Dateisystemrichtlinien, einschließlich der Verweigerung des Root-Zugriffs für alle IAM-Prinzipale, mit Ausnahme einer bestimmten Management-Workstation, finden Sie unter: [Exemplarische Vorgehensweise: Root-Squashing mithilfe der IAM-Autorisierung für NFS-Clients aktivieren](#)

Beispiel: Zugriff auf einen EFS Access Point gewähren

Sie verwenden eine EFS-Zugriffsrichtlinie, um einem NFS-Client einen anwendungsspezifischen Einblick in gemeinsam genutzte dateibasierte Datenmengen in einem EFS-Dateisystem zu bieten. Sie gewähren die Zugriffspunktberechtigungen auf dem Dateisystem mithilfe einer Dateisystemrichtlinie.

In diesem Beispiel für eine Dateirichtlinie wird ein Bedingungelement verwendet, um einem bestimmten Zugriffspunkt, der durch seinen ARN definiert ist, uneingeschränkten Zugriff auf das Dateisystem zu gewähren.

Weitere Informationen zur Verwendung von EFS-Zugriffspunkten finden Sie unter [Arbeiten mit Amazon EFS Access Points](#).

```
{
  "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" }
        }
      }
    ]
  }
}
```

AWSverwaltete Richtlinien für Amazon EFS

Eine von AWS verwaltete Richtlinie ist eine eigenständige Richtlinie, die von AWS erstellt und verwaltet wird. Von AWS verwaltete Richtlinien stellen Berechtigungen für viele häufige Anwendungsfälle bereit, damit Sie beginnen können, Benutzern, Gruppen und Rollen Berechtigungen zuzuweisen.

Beachten Sie, dass AWS-verwaltete Richtlinien möglicherweise nicht die geringsten Berechtigungen für Ihre spezifischen Anwendungsfälle gewähren, da sie für alle AWS-Kunden verfügbar sind. Wir empfehlen Ihnen, die Berechtigungen weiter zu reduzieren, indem Sie [kundenverwaltete Richtlinien](#) definieren, die speziell auf Ihre Anwendungsfälle zugeschnitten sind.

Die Berechtigungen, die in den von AWS verwalteten Richtlinien definiert sind, können nicht geändert werden. Wenn AWS Berechtigungen aktualisiert, die in einer von AWS verwalteten Richtlinie definiert werden, wirkt sich das Update auf alle Prinzipalidentitäten (Benutzer, Gruppen und Rollen) aus, denen die Richtlinie zugeordnet ist. AWS aktualisiert am wahrscheinlichsten eine von AWS verwaltete Richtlinie, wenn ein neuer AWS-Service gestartet wird oder neue API-Operationen für bestehende Dienste verfügbar werden.

Weitere Informationen finden Sie unter [Von AWS verwaltete Richtlinien](#) im IAM-Benutzerhandbuch.

AWSverwaltete 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 ermöglichenAWS Management Console.

Details zu Berechtigungen

Diese Richtlinie umfasst die folgenden Berechtigungen.

- **elasticfilesystem**— Ermöglicht Prinzipalen, alle Aktionen in der Amazon EFS-Konsole auszuführen. Es ermöglicht Prinzipalen auch das Erstellen (`elasticfilesystem:Backup`) und Wiederherstellen (`elasticfilesystem:Restore`) von Backups mithilfe von. AWS Backup
- **cloudwatch**— Ermöglicht Prinzipalen, CloudWatch Amazon-Dateisystem-Metriken und Alarmer 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, Virtual Private Clouds (VPCs) und VPC-Attributen, die mit einem Amazon 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.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "cloudwatch:DescribeAlarmsForMetric",
        "cloudwatch:GetMetricData",
        "ec2:CreateNetworkInterface",
        "ec2>DeleteNetworkInterface",
        "ec2:DescribeAvailabilityZones",
        "ec2:DescribeNetworkInterfaceAttribute",
        "ec2:DescribeNetworkInterfaces",
        "ec2:DescribeSecurityGroups",
```

```

        "ec2:DescribeSubnets",
        "ec2:DescribeVpcAttribute",
        "ec2:DescribeVpcs",
        "ec2:ModifyNetworkInterfaceAttribute",
        "elasticfilesystem:Backup",
        "elasticfilesystem:CreateFileSystem",
        "elasticfilesystem:CreateMountTarget",
        "elasticfilesystem:CreateTags",
        "elasticfilesystem:CreateAccessPoint",
        "elasticfilesystem:CreateReplicationConfiguration",
        "elasticfilesystem>DeleteFileSystem",
        "elasticfilesystem>DeleteMountTarget",
        "elasticfilesystem>DeleteTags",
        "elasticfilesystem>DeleteAccessPoint",
        "elasticfilesystem>DeleteFileSystemPolicy",
        "elasticfilesystem>DeleteReplicationConfiguration",
        "elasticfilesystem:DescribeAccountPreferences",
        "elasticfilesystem:DescribeBackupPolicy",
        "elasticfilesystem:DescribeFileSystems",
        "elasticfilesystem:DescribeFileSystemPolicy",
        "elasticfilesystem:DescribeLifecycleConfiguration",
        "elasticfilesystem:DescribeMountTargets",
        "elasticfilesystem:DescribeMountTargetSecurityGroups",
        "elasticfilesystem:DescribeReplicationConfigurations",
        "elasticfilesystem:DescribeTags",
        "elasticfilesystem:DescribeAccessPoints",
        "elasticfilesystem:ModifyMountTargetSecurityGroups",
        "elasticfilesystem:PutAccountPreferences",
        "elasticfilesystem:PutBackupPolicy",
        "elasticfilesystem:PutLifecycleConfiguration",
        "elasticfilesystem:PutFileSystemPolicy",
        "elasticfilesystem:UpdateFileSystem",
        "elasticfilesystem:UpdateFileSystemProtection",
        "elasticfilesystem:TagResource",
        "elasticfilesystem:UntagResource",
        "elasticfilesystem:ListTagsForResource",
        "elasticfilesystem:Restore",
        "kms:DescribeKey",
        "kms:ListAliases"
    ],

    "Sid": "ElasticFileSystemFullAccess",
    "Effect": "Allow",
    "Resource": "*"

```

```
    },  
    {  
      "Action": "iam:CreateServiceLinkedRole",  
      "Sid": "CreateServiceLinkedRoleForEFS",  
      "Effect": "Allow",  
      "Resource": "*",  
      "Condition": {  
        "StringEquals": {  
          "iam:AWSServiceName": [  
            "elasticfilesystem.amazonaws.com"  
          ]  
        }  
      }  
    }  
  ]  
}
```

AWSverwaltete Richtlinie: AmazonElasticFileSystemReadOnlyAccess

Sie können die AmazonElasticFileSystemReadOnlyAccess-Richtlinie an Ihre IAM-Identitäten anfügen.

Diese Richtlinie gewährt nur Lesezugriff auf Amazon EFS über die AWS Management Console.

Details zu Berechtigungen

Diese Richtlinie umfasst die folgenden Berechtigungen.

- **elasticfilesystem**— Ermöglicht Prinzipalen, Attribute von Amazon EFS-Dateisystemen zu beschreiben, darunter Kontoeinstellungen, Sicherungs- und Dateisystemrichtlinien, Lebenszykluskonfiguration, Mount-Ziele und deren Sicherheitsgruppen, Tags und Zugriffspunkte 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 Principals, Availability Zones, Netzwerkschnittstellen und deren Attribute, Sicherheitsgruppen, Subnetze, VPCs und deren Attribute in der Amazon EFS-Konsole anzuzeigen.
- **kms**— Ermöglicht Prinzipalen, Aliase für AWS KMS Schlüssel in der Amazon EFS-Konsole aufzulisten.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "cloudwatch:DescribeAlarmsForMetric",
        "cloudwatch:GetMetricData",
        "ec2:DescribeAvailabilityZones",
        "ec2:DescribeNetworkInterfaceAttribute",
        "ec2:DescribeNetworkInterfaces",
        "ec2:DescribeSecurityGroups",
        "ec2:DescribeSubnets",
        "ec2:DescribeVpcAttribute",
        "ec2:DescribeVpcs",
        "elasticfilesystem:DescribeAccountPreferences",
        "elasticfilesystem:DescribeBackupPolicy",
        "elasticfilesystem:DescribeFileSystems",
        "elasticfilesystem:DescribeFileSystemPolicy",
        "elasticfilesystem:DescribeLifecycleConfiguration",
        "elasticfilesystem:DescribeMountTargets",
        "elasticfilesystem:DescribeMountTargetSecurityGroups",
        "elasticfilesystem:DescribeTags",
        "elasticfilesystem:DescribeAccessPoints",
        "elasticfilesystem:DescribeReplicationConfigurations",
        "elasticfilesystem:ListTagsForResource",
        "kms:ListAliases"
      ],
      "Resource": "*"
    }
  ]
}
```

AWSverwaltete Richtlinie: Zugriff AmazonElasticFileSystemClientReadWrite

Sie können die AmazonElasticFileSystemClientReadWriteAccess Richtlinie an eine IAM-Entität anhängen.

Diese Richtlinie gewährt dem Client Lese- und Schreibzugriff auf ein Amazon EFS-Dateisystem. Diese Richtlinie ermöglicht NFS-Clients das Mounten, Lesen und Schreiben in Amazon EFS-Dateisysteme.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticfilesystem:ClientMount",
        "elasticfilesystem:ClientWrite",
        "elasticfilesystem:DescribeMountTargets"
      ],
      "Resource": "*"
    }
  ]
}
```

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. Abonnieren Sie den RSS-Feed auf der Amazon EFS-Seite, um automatische Benachrichtigungen über Änderungen an dieser [Dokumentverlauf](#) Seite zu erhalten.

Änderung	Beschreibung	Datum
Aktualisieren Sie eine bestehende Richtlinie	<p>Richtlinie: AmazonElasticFileSystemFullAccess</p> <p>Amazon EFS hat eine neue Berechtigung hinzugefügt, die es Prinzipalen ermöglicht, den Schutz auf einem Dateisystem zu deaktivieren und zu aktivieren. Die Berechtigungen sind erforderlich, damit Amazon EFS in ein vorhandenes Dateisystem replizieren kann.</p>	27. November 2023
Aktualisieren Sie eine bestehende Richtlinie	<p>Richtlinie: AmazonElasticFileSystemServiceRolePolicy</p> <p>Amazon EFS hat neue Berechtigungen hinzugefügt, um es Prinzipalen zu ermöglichen, Amazon EFS-Replikationen zu erstellen, zu beschreiben und zu löschen und Amazon EFS-Dateisysteme</p>	25 Januar 2022

Änderung	Beschreibung	Datum
	zu erstellen. Die Berechtigungen sind erforderlich, damit Amazon EFS die Konfiguration der Dateisystemreplikation im Namen des Benutzers verwalten kann.	
Aktualisieren Sie eine bestehende Richtlinie	<p>Richtlinie: AmazonElasticFileSystemReadOnlyAccess</p> <p>Amazon EFS hat eine neue Berechtigung hinzugefügt, die es Prinzipalen ermöglicht, Amazon EFS-Replikationen zu beschreiben. Die Berechtigungen sind erforderlich, damit Benutzer Konfigurationen für die Dateisystemreplikation einsehen können.</p>	25 Januar 2022
Aktualisieren Sie eine bestehende Richtlinie	<p>Richtlinie: AmazonElasticFileSystemFullAccess</p> <p>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 Konfigurationen für die Dateisystemreplikation verwalten können.</p>	25 Januar 2022
Die Tracking-Richtlinie wurde gestartet	<p>Richtlinie: AmazonElasticFileSystemClientReadWriteZugriff</p> <p>Gewährt NFS-Clients Lese- und Schreibrechte auf Amazon EFS-Dateisystemen.</p>	3. Januar 2022
Die Tracking-Richtlinie wurde gestartet	<p>Richtlinie: AmazonElasticFileSystemServiceRolePolicy</p> <p>Die serviceverknüpften Rollenberechtigungen für Amazon EFS.</p>	8. Oktober 2021

Änderung	Beschreibung	Datum
Aktualisieren Sie eine bestehende Richtlinie	<p>Richtlinie: AmazonElasticFileSystemFullAccess</p> <p>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.</p>	7. Mai 2021
Aktualisieren Sie eine bestehende Richtlinie	<p>Richtlinie: AmazonElasticFileSystemReadOnlyAccess</p> <p>Amazon EFS hat neue Berechtigungen hinzugefügt, um es den Prinzipalen zu ermöglichen, die Amazon EFS-Kontoeinstellungen zu beschreiben. Die Berechtigungen sind erforderlich, damit Benutzer die Kontoeinstellungen in der Amazon EFS-Konsole einsehen können.</p>	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

Verwenden von Tags mit Amazon EFS

Sie können Tags verwenden, um den Zugriff auf Amazon EFS-Ressourcen zu kontrollieren und um attributbasierte Zugriffskontrolle (ABAC) zu verwenden. Weitere Informationen finden Sie unter:

- [Markieren der Amazon-EFS-Ressourcen](#)
- [Zugriffssteuerung auf der Grundlage von Tags auf einer Ressource](#)
- [Wofür wird ABAC in verwendet AWS](#). im IAM-Benutzerhandbuch

 Note

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

Um während der Erstellung Tags auf Amazon EFS-Ressourcen anzuwenden, müssen Benutzer über bestimmte AWS Identity and Access Management (IAM) -Berechtigungen verfügen.

Erteilen von Berechtigung zum Markieren von Ressourcen während der Erstellung

Mit den folgenden Aktionen zur Erstellung der Amazon EFS-API mit -Tags können Sie Tags beim Erstellen der Ressource angeben.

- `CreateAccessPoint`
- `CreateFileSystem`

Damit Benutzer diese Möglichkeit erhalten, benötigen sie die Berechtigung zum Verwenden der Aktion, die Ressource wie `elasticfilesystem:CreateAccessPoint` oder `elasticfilesystem:CreateFileSystem`. Wenn Tags in der Aktion angegeben werden, mit der die Ressource erstellt wird, wird AWS eine zusätzliche Autorisierung für die `elasticfilesystem:TagResource` -Aktion aus, um die Berechtigungen der Benutzer zum Erstellen von Tags zu überprüfen. 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: Erteilen von Tags zu Dateisystemen nur zum Zeitpunkt der Erstellung

Mit der folgenden Beispielrichtlinie können Benutzer Dateisysteme erstellen und Tags nur während der Erstellung 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"
        }
    }
}
]
}

```

Verwenden 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 Steuerung auf zwei Arten bereitstellen:

- Sie können den Zugriff auf Amazon EFS-Ressourcen basierend auf Amazon EFS-Ressourcen basierend auf Amazon EFS-Ressourcen basierend auf -Ressourcen basierend auf Amazon EFS-Ressourcen basierend auf
- Sie können die Tags in einer IAM-Anforderungsbedingung übergeben werden können.

Informationen zur Verwendung von Tags zur Steuerung des Zugriffs auf AWS Ressourcen finden Sie im IAM-Benutzerhandbuch unter [Steuern des Zugriffs mithilfe von Tags](#).

Zugriffssteuerung auf der Grundlage von Tags auf einer Ressource

Um zu steuern, welche Aktionen ein Benutzer oder eine Rolle auf einer Amazon EFS-Ressource ausführen kann, können Sie Tags für die Ressource verwenden. Beispielsweise möchten Sie möglicherweise bestimmte API-Operationen für eine Dateisystemressource zulassen oder verweigern, die auf dem Schlüssel-Wert-Paar des Tags für die Ressource basieren.

Example Richtlinie: Erstellen Sie ein Dateisystem nur, wenn ein bestimmtes Tag verwendet wird

Die folgende Beispielrichtlinie ermöglicht es dem Benutzer, ein Dateisystem nur zu erstellen, wenn er es mit einem bestimmten Tag-Schlüssel-Wert-Paar taggt, in diesem Beispielkey=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

Mit der folgenden Beispielrichtlinie können Benutzer nur Dateisysteme löschen, die mit markiert sindDepartment=Finance.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "elasticfilesystem>DeleteFileSystem"
      ],
      "Resource": "arn:aws:elasticfilesystem:region:account-id:file-system/*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/Department": "Finance"
        }
      }
    }
  ]
}
```

Verwendung von serviceverknüpften Rollen für Amazon EFS

Amazon Elastic File System verwendet eine AWS Identity and Access Management (IAM) [serviceverknüpfte Rolle](#). Die serviceverknüpfte Rolle von Amazon EFS ist eine einzigartige Art von IAM-Rolle, die direkt mit Amazon EFS verknüpft ist. Die vordefinierte serviceverknüpfte Rolle von Amazon EFS umfasst Berechtigungen, die der Service für den Aufruf anderer in AWS-Services Ihrem Namen benötigt.

Eine servicegebundene Rolle vereinfacht die Einrichtung von Amazon EFS, da Sie die erforderlichen Berechtigungen nicht manuell hinzufügen müssen. Amazon EFS definiert die Berechtigungen seiner serviceverknüpften 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 serviceverknüpfte Rolle von Amazon EFS nur löschen, wenn Sie zuvor Ihre Amazon-EFS-Dateisysteme gelöscht haben. Dies schützt Ihre Amazon-EFS-Ressourcen, da Sie nicht versehentlich die Zugriffsberechtigung für die Ressourcen entfernen können.

Serviceverknüpfte Rollen ermöglichen auch, dass alle API-Aufrufe über AWS CloudTrail sichtbar sind. Das hilft bei Überwachungs- und Prüfungsanforderungen, da Sie alle in Ihrem Namen von Amazon EFS ausgeführten Aktionen nachverfolgen können. Weitere Informationen finden Sie unter [Protokolleinträge für dienstverknüpfte EFS-Rollen](#).

Serviceverknüpfte Rollenberechtigungen für Amazon EFS

Amazon EFS verwendet die serviceverknüpfte Rolle `AWSServiceRoleForAmazonElasticFileSystem`, um es Amazon EFS zu ermöglichen, AWS Ressourcen im Namen Ihrer EFS-Dateisysteme aufzurufen und zu verwalten.

Die serviceverknüpfte Rolle `AWSServiceRoleForAmazonElasticFileSystem` vertraut darauf, dass die folgenden Services die Rolle annehmen:

- `elasticfilesystem.amazonaws.com`

Die Richtlinie für Rollenberechtigungen erlaubt Amazon EFS, die in der Richtliniendefinition JSON enthaltenen Aktionen durchzuführen:

```
{
  "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"
    ],
    "Resource": "*"
}
]
}

```

 Note

Sie müssen die IAM-Berechtigungen manuell konfigurieren, AWS KMS wenn Sie ein neues Amazon EFS-Dateisystem erstellen, das im Ruhezustand verschlüsselt ist. Weitere Informationen hierzu finden Sie unter [Verschlüsseln von Daten im Ruhezustand](#).

Erstellen einer serviceverknüpften Rolle für Amazon EFS

Sie müssen Berechtigungen konfigurieren, damit eine IAM-Entität (z. B. Benutzer, Gruppe oder Rolle) eine servicegebundene Rolle erstellen kann. Fügen Sie dafür die `iam:CreateServiceLinkedRole` Berechtigung einer IAM-Entität hinzu, wie im folgenden Beispiel gezeigt.

```
{
  "Action": "iam:CreateServiceLinkedRole",
  "Effect": "Allow",
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "iam:AWSServiceName": [
        "elasticfilesystem.amazonaws.com"
      ]
    }
  }
}
```

Weitere Informationen finden Sie unter [Berechtigungen für serviceverknüpfte Rollen](#) im IAM-Benutzerhandbuch.

Sie müssen eine serviceverknüpfte Rolle nicht manuell erstellen. Wenn Sie Mountziele oder eine Replikationskonfiguration für Ihr EFS-Dateisystem in der AWS Management Console, der AWS CLI, oder der AWS API erstellen, erstellt Amazon EFS die serviceverknüpfte Rolle für Sie.

Wenn Sie diese serviceverknüpfte Rolle löschen und sie dann erneut erstellen müssen, können Sie dasselbe Verfahren anwenden, um die Rolle in Ihrem Konto neu anzulegen. Wenn Sie Mountziele oder eine Replikationskonfiguration für Ihr EFS-Dateisystem erstellen, erstellt Amazon EFS die serviceverknüpfte Rolle erneut für Sie.

Bearbeiten einer serviceverknüpften Rolle für Amazon EFS

Amazon EFS erlaubt Ihnen nicht, die `AWSServiceRoleForAmazonElasticFileSystem` serviceverknüpfte 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 unter [Bearbeiten einer serviceverknüpften Rolle](#) im IAM-Benutzerhandbuch.

Löschen einer serviceverknüpften Rolle für Amazon EFS

Wenn Sie eine Funktion 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.

Note

Wenn der Amazon-EFS-Service die Rolle verwendet, wenn Sie versuchen, die Ressourcen zu löschen, schlägt der Löschvorgang möglicherweise fehl. Wenn dies passiert, warten Sie einige Minuten und versuchen Sie es erneut.

Um Amazon EFS-Ressourcen zu löschen, die von der `AWSServiceRoleForAmazonElasticFileSystem`

Führen Sie die folgenden Schritte aus, um die von der verwendeten Amazon EFS-Ressourcen zu löschen `AWSServiceRoleForAmazonElasticFileSystem`. Das detaillierte Verfahren finden Sie unter [Schritt 4: Bereinigen Sie Ihre Ressourcen und schützen Sie Ihr AWS-Konto](#).

1. Stellen Sie auf Ihrer Amazon-EC2-Instance das Amazon-EFS-Dateisystem bereit.
2. Löschen Sie das Amazon-EFS-Dateisystem.
3. Löschen Sie die benutzerdefinierte Sicherheitsgruppe für das Dateisystem.

Warning

Wenn Sie die Standardsicherheitsgruppe für Ihre Virtual Private Cloud (VPC) verwendet haben, löschen Sie diese Sicherheitsgruppe nicht.

So löschen Sie die servicegebundene Rolle mit IAM

Verwenden Sie die IAM-Konsole, AWS CLI- oder AWS-API, um die `AWSServiceRoleForAmazonElasticFileSystem` serviceverknüpfte Rolle zu löschen. Weitere Informationen finden Sie unter [Löschen einer serviceverknüpften Rolle](#) im IAM-Benutzerhandbuch.

Fehlerbehebung bei Identität und Zugriff auf Amazon Elastic File System

Verwenden Sie die folgenden Informationen, um häufig auftretende Probleme zu diagnostizieren und zu beheben, die bei der Arbeit mit Amazon EFS und IAM auftreten können.

Themen

- [Ich bin nicht berechtigt, eine Aktion in Amazon EFS durchzuführen](#)
- [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 berechtigt, eine Aktion in Amazon EFS durchzuführen

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.

Wenden Sie sich an Ihren AWS-Administrator, falls Sie weitere Unterstützung benötigen. Ihr Administrator hat Ihnen Ihre Anmeldeinformationen zur Verfügung gestellt.

Ich bin nicht berechtigt, iam auszuführen: PassRole

Wenn Sie eine Fehlermeldung erhalten, dass Sie nicht berechtigt sind, die `iam:PassRole` Aktion durchzuführen, müssen Ihre Richtlinien aktualisiert werden, damit Sie eine Rolle an Amazon EFS übergeben können.

Einige AWS-Services erlauben die Übergabe einer vorhandenen Rolle an diesen Dienst, sodass keine neue Servicerolle oder serviceverknüpfte Rolle erstellt werden muss. 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.

Wenden Sie sich an Ihren AWS-Administrator, falls Sie weitere Unterstützung benötigen. 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, die Benutzer in anderen Konten oder Personen außerhalb Ihrer Organisation für den Zugriff auf Ihre Ressourcen verwenden können. Sie können festlegen, wem die Übernahme der Rolle anvertraut wird. Im Fall von Diensten, die ressourcenbasierte Richtlinien oder Zugriffskontrolllisten (Access Control Lists, ACLs) verwenden, können Sie diese Richtlinien verwenden, um Personen Zugriff auf Ihre Ressourcen zu gewähren.

Weitere Informationen dazu finden Sie hier:

- Informationen darüber, ob Amazon EFS diese Funktionen unterstützt, finden Sie unter [So funktioniert Amazon Elastic File System mit IAM](#).
- Informationen zum Gewähren des Zugriffs auf Ihre Ressourcen für alle Ihre AWS-Konten finden Sie unter [Gewähren des Zugriffs für einen IAM-Benutzer in einem anderen Ihrer AWS-Konto](#) im IAM-Benutzerhandbuch.
- Informationen dazu, wie Sie AWS-Konten-Drittanbieter Zugriff auf Ihre Ressourcen bereitstellen, finden Sie unter [Gewähren des Zugriffs auf AWS-Konten von externen Benutzern](#) im IAM-Benutzerhandbuch.

- 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 [So unterscheiden sich IAM-Rollen von ressourcenbasierten Richtlinien](#) im IAM-Benutzerhandbuch.

Verwendung von IAM zur Steuerung des Dateisystemdatenzugriffs

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 finden Sie CloudTrailunter [Protokollieren von Amazon EFS-API-Aufrufen mit AWS CloudTrail](#).

Important

Sie müssen die EFS-Mountinghilfe verwenden, um Ihre Amazon EFS-Dateisysteme einzuhängen, damit die IAM-Autorisierung zur Steuerung des Client-Zugriffs verwendet werden kann. 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.

Aktion	Beschreibung
<code>elasticfilesystem:ClientMount</code>	Ermöglicht den Nur-Lese-Zugriff auf ein Dateisystem.
<code>elasticfilesystem:ClientWrite</code>	Bietet Schreibrechte für ein Dateisystem.
<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	Beschreibung	Operator
<code>aws:SecureTransport</code>	Verwenden Sie diesen Schlüssel, um Clients zu verpflichten, TLS zu verwenden, wenn sie sich	Boolesch

EFS-Bedingungsschlüssel	Beschreibung	Operator
	mit einem EFS-Dateisystem verbinden.	
<code>aws:SourceIp</code>	Private IP-Adresse des Clients, der auf ein EFS-Dateisystem zugreift.	String
<code>elasticfilesystem:AccessPointArn</code>	ARN des EFS-Zugangspunkts, mit dem sich der Client verbindet.	String
<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 Elastic File System](#).

Steuerung des Netzwerkzugriffs auf Amazon 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 der Netzwerkebene verwenden, wie 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

- [Verwendung von VPC-Sicherheitsgruppen für Amazon EC2-Instance und Mounting-Ziele](#)
- [Quell-Ports für die Arbeit mit EFS](#)

- [Sicherheitsüberlegungen für den Netzwerkzugriff](#)
- [Arbeiten mit Schnittstellen-VPC-Endpunkten in Amazon EFS](#)

Verwendung von VPC-Sicherheitsgruppen für Amazon EC2-Instance und Mounting-Ziele

Wenn Sie Amazon EFS verwenden, geben Sie Amazon EC2-Sicherheitsgruppen für Ihre EC2-Instances und Sicherheitsgruppen für die mit dem Dateisystem verbundenen EFS-Mounting-Ziele an. Eine Sicherheitsgruppe fungiert als Firewall, und die Regeln, die Sie hinzufügen, definieren den Datenfluss. In der „Erste Schritte“-Übung haben Sie beim Starten der EC2-Instance eine Sicherheitsgruppe erstellt. Dann haben Sie dem EFS-Mounting-Ziel eine weitere Sicherheitsgruppe (die Standardsicherheitsgruppe für Ihre Standard-VPC) zugeordnet. Dieser Ansatz funktioniert für die „Erste Schritte“-Übung. Für eine Produktionsumgebung sollten Sie jedoch Sicherheitsgruppen mit möglichst geringen Nutzungsberechtigungen für EFS einrichten.

Sie können eingehenden und ausgehenden Datenverkehr für Ihr EFS-Dateisystem autorisieren. Dazu fügen Sie Regeln hinzu, die es Ihrer EC2-Instance erlauben, sich über das Mountinghilfe mit Ihrem Amazon EFS-Dateisystem zu verbinden, indem Sie den NFS-Port (Network File System) verwenden. Gehen Sie wie folgt vor, um Sicherheitsgruppen zu erstellen und zu aktualisieren.

So erstellen Sie Sicherheitsgruppen für EC2-Instances und Mounting-Ziele

1. Erstellen Sie zwei Sicherheitsgruppen in Ihrer VPC.

Anweisungen hierzu finden Sie in der Anleitung „So erstellen Sie eine Sicherheitsgruppe“ unter [Erstellen einer Sicherheitsgruppe](#) im Amazon VPC-Benutzerhandbuch.

2. Öffnen Sie die Amazon VPC Management Console unter <https://console.aws.amazon.com/vpc/> und überprüfen Sie die Standardregeln für diese Sicherheitsgruppen. Beide Sicherheitsgruppen sollten nur über eine Regel verfügen, die ausgehenden Datenverkehr zulässt.

So aktualisieren Sie den erforderlichen Zugriff für Ihre Sicherheitsgruppen:

1. Öffnen Sie die Amazon VPC-Konsole unter <https://console.aws.amazon.com/vpc/>.
2. Fügen Sie eine Regel für Ihre EC2-Sicherheitsgruppe hinzu, die eingehenden Datenverkehr über Secure Shell (SSH) von beliebigen Hosts erlaubt. Optional können Sie die Adresse der Source (Quelle) beschränken.

Sie müssen keine ausgehende Regel hinzufügen, da die Standardausgangsregel jeden Datenverkehr nach außen zulässt. Ist dies nicht der Fall, müssen Sie eine Regel für ausgehenden Datenverkehr hinzufügen, um eine TCP-Verbindung auf dem NFS-Port zu öffnen, wobei die Sicherheitsgruppe des Mounting-Ziels als Ziel identifiziert wird.

Anweisungen dazu finden Sie unter [Hinzufügen und Entfernen von Regeln](#) im Amazon VPC-Benutzerhandbuch.

3. Fügen der Mounting-Ziele für den gesamten eingehenden und ausgehenden Datenverkehr hinzu.
 - Fügen Sie eine eingehende Regel für die Sicherheitsgruppe mount target hinzu, um den eingehenden Zugriff von der EC2-Sicherheitsgruppe zu erlauben. Identifizieren Sie die EC2-Sicherheitsgruppe als Quelle.
 - Fügen Sie eine ausgehende Regel hinzu, um die TCP-Verbindung auf allen NFS-Ports zu öffnen. Geben Sie die EC2-Sicherheitsgruppe als Ziel an.

Anweisungen dazu finden Sie unter [Hinzufügen und Entfernen von Regeln](#) im Amazon VPC-Benutzerhandbuch.

4. Überprüfen Sie, ob beide Sicherheitsgruppen jetzt Zugriff auf eingehenden und ausgehenden Datenverkehr autorisieren.

Weitere Informationen über Sicherheitsgruppen finden Sie unter [Sicherheitsgruppen für EC2-VPC](#) im Amazon EC2 User Guide for Linux Instances.

Quell-Ports für die Arbeit mit 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 nicht autorisierte, nonkernel-Prozesse (-m owner --uid-owner 1-4294967294) eine Verbindung mit dem NFS-Port (-m tcp -p tcp -dport 2049) herstellen.

Sicherheitsüberlegungen für den Netzwerkzugriff

Ein NFS Version 4.1 (NFSv4.1) Client kann ein Dateisystem nur mounten, wenn er eine Netzwerkverbindung zum NFS-Port (TCP-Port 2049) eines der Mounting-Ziele des Dateisystems herstellen kann. Ebenso kann ein NFSv4.1-Client eine Benutzer- und Gruppen-ID beim Zugriff auf ein Dateisystem nur bestätigen, 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 in verbundenen VPCs
 - On-Premises-Server, die mit einer Amazon VPC über AWS Direct Connect und eine AWS Virtual Private Network (VPN) verbunden sind
- Netzwerk-Zugriffskontrolllisten (Access Control Lists, ACLs) der VPC-Subnetze des Clients und der Mounting-Ziele für den Zugriff von Stellen außerhalb der Subnetze des Mounting-Ziels – Um ein Dateisystem zu mounten, muss der Client eine TCP-Verbindung mit dem NFS-Port eines Mounting-Ziels herstellen und den zurückfließenden Datenverkehr empfangen können.
- 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 ein Mounting-Ziel besitzen, dessen Netzwerkschnittstelle eine Sicherheitsgruppe mit einer Regel besitzt, die auf dem NFS-Port von der Instance eingehende Verbindungen unterstützt, 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. Regeln für eingehende Verbindungen auf anderen Ports als dem NFS-Port sowie alle Regeln für ausgehende Verbindungen werden von Netzwerkschnittstellen nicht für Dateisystem-Mounting-Ziele verwendet.

- Die Mounting-Instance muss über eine Netzwerkschnittstelle mit einer Sicherheitsgruppe verfügen, die ausgehende Verbindungen über den NFS-Port eines der Mounting-Ziele des Dateisystems ermöglicht, Sie können ausgehende Verbindungen anhand von IP-Adressen (CIDR-Bereich) oder Sicherheitsgruppen aktivieren.

Weitere Informationen finden Sie unter [Erstellen und Verwalten von Mount-Zielen und Sicherheitsgruppen](#).

Arbeiten mit Schnittstellen-VPC-Endpunkten in Amazon EFS

Um eine private Verbindung zwischen Ihrer Virtual Private Cloud (VPC) und der Amazon EFS-API zu erstellen, können Sie einen Schnittstellen-VPC-Endpunkt erstellen. Der Endpunkt bietet sichere Konnektivität zur Amazon EFS-API, ohne dass ein Internet-Gateway, eine NAT-Instance oder eine VPN-Verbindung erforderlich ist. Weitere Informationen finden Sie unter [Interface VPC Endpoints](#) (Interface VPC-Endpunkte) im Amazon VPC-Benutzerhandbuch.

Schnittstellen-VPC-Endpunkte werden über bereitgestellt. Dies ist eine FunktionAWSPrivateLink, die private Kommunikation zwischenAWS Services unter Verwendung privater IP-Adressen ermöglicht. Erstellen Sie zur Verwendung mithilfe der Amazon VPC-KonsoleAWSPrivateLink, API oder CLI einen Schnittstellen-VPC-Endpunkt für Amazon EFS in Ihrer VPC. Auf diese Weise erstellen Sie eine elastic network interface NetzwerkSchnittstellen-EFS-API-API für Amazon EFS-API-Anfragen. Sie können auch von lokalen Umgebungen oder von anderen VPCs mit AWS VPN, AWS Direct Connect oder VPC-Peering auf einen VPC-Endpunkt zugreifen. Weitere Informationen finden Sie unter [Accessing ServicesAWSPrivateLink Through](#) im Amazon VPC-Benutzerhandbuch.

Einen Schnittstellenendpunkt für Amazon EFS erstellen

Um einen Schnittstellen-VPC-Endpunkt für Amazon EFS-Endpunkt für Amazon EFS-Endpunkt für Amazon EFS:

- **com.amazonaws.*region*.elasticfilesystem**— Erzeugt einen Endpunkt für Amazon EFS-API-Operationen.
- **com.amazonaws.*region*.elasticfilesystem-fips**— Erstellt einen Endpunkt für die Amazon EFS-API ([Federal Information Processing Standard \(FIPS\) 140-2](#)).

Eine vollständige Liste der Amazon EFS-Endpunkte finden Sie unter [Amazon Elastic File System](#) in der Allgemeine Amazon Web Services-Referenz.

Weitere Informationen zum Erstellen eines Schnittstellenendpunkts finden Sie unter [Creating an Interface Endpoint](#) im Amazon VPC-Benutzerhandbuch.

Erstellen einer VPC-Endpunktrichtlinie einer VPC-Endpunktrichtlinie EFS

Um den Zugriff auf die Amazon EFS-Endpunkt zu steuern, können Sie Ihrem VPC-Endpunkt eine AWS Identity and Access Management (IAM) -Endpunkt anfügen. 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 [Steuerung des Zugriffs auf Services mit VPC-Endpunkten](#) im Amazon VPC User Guide.

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": "*"
    }
  ]
}
```

Weitere Informationen finden Sie unter [Verwenden von VPC-Endpunktrichtlinie](#) im Amazon VPC Benutzerhandbuch.

Mit Benutzern, Gruppen und Berechtigungen auf Network File System-(NFS-)Level arbeiten

Themen

- [Datei- und Verzeichnisberechtigungen](#)
- [Beispiel für Amazon EFS-Dateisystem-Nutzungsfälle und Berechtigungen](#)
- [Benutzer- und Gruppen-ID-Berechtigungen für Dateien und Verzeichnisse in einem Dateisystem](#)
- [Kein Root-Squashing](#)
- [Zwischenspeichern von Berechtigungen](#)
- [Ändern des Besitzes an Dateisystemobjekten](#)
- [EFS-Zugangspunkte](#)

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 Amazon EFS Access Points](#).

Amazon EFS-Dateisystemobjekte haben einen Unix-ähnlichen Modus, der mit ihnen verbunden ist. 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 zugeordneten numerischen IDs, um die Berechtigungen zu prüfen, wenn ein Benutzer versucht, auf ein Dateisystemobjekt zuzugreifen.

Note

Das NFS-Protokoll unterstützt maximal 16 Gruppen-IDs (GIDs) pro Benutzer und alle weiteren GIDs werden von NFS-Client-Anfragen abgeschnitten. 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.

Datei- und Verzeichnisberechtigungen

Dateien und Verzeichnisse in einem EFS-Dateisystem unterstützen standardmäßige, Unix-ähnliche Lese-/Schreib- und Ausführungsberechtigungen basierend auf der Benutzer- und Gruppen-ID des NFSv4.1-Mounting-Clients, sofern nicht vom EFS-Zugangspunkt überschrieben. Weitere Informationen finden Sie unter [Mit Benutzern, Gruppen und Berechtigungen auf Network File System-\(NFS-\)Level arbeiten](#).

Note

Diese Zugriffssteuerungsebene ist davon abhängig, dass der NFSv4.1-Client hinsichtlich der Bestätigung der Benutzer- und Gruppen-ID vertrauenswürdig ist. Sie können AWS Identity and Access Management (IAM) ressourcenbasierte Richtlinien und Identitätsrichtlinien verwenden, um NFS-Clients zu autorisieren und Lese-, Schreib- und Root-Zugriffsberechtigungen bereitzustellen. Sie können EFS-Zugangspunkte verwenden, um die vom NFS-Client bereitgestellten Benutzer- und Gruppenidentitätsinformationen des Betriebssystems zu übergehen. Weitere Informationen finden Sie unter [Verwendung von IAM zur Steuerung des Dateisystemdatenzugriffs](#) und [Erstellen und Löschen von Zugangspunkten](#).

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.

Beispiel für Amazon EFS-Dateisystem-Nutzungsfälle und Berechtigungen

Nachdem Sie ein Amazon EFS-Dateisystem und Mountinghilfe für das Dateisystem in Ihrer VPC erstellt haben, können Sie das Remote-Dateisystem lokal auf Ihrer Amazon 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-step Anweisungen finden Sie unter [Anleitung: Erstellen Sie beschreibbare Unterverzeichnisse für einzelne Benutzer und konfigurieren Sie das automatische erneute Mounting bei Neustarts](#).
- Erlauben Sie Benutzern, in das Amazon EFS-Dateisystem-Root zu schreiben. Ein Benutzer mit Root-Berechtigung kann anderen Benutzern Zugriff auf das Dateisystem erteilen.
- Um die Eigentümerschaft des Amazon EFS-Dateisystems auf einen Nicht-Root-Benutzer und eine Nicht-Root-Gruppe zu ändern, gehen Sie wie folgt vor:

```
$ 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 Berechtigungen.

Benutzer- und Gruppen-ID-Berechtigungen für Dateien und Verzeichnisse in einem Dateisystem

Dateien und Verzeichnisse in einem Amazon EFS-Dateisystem unterstützen standardmäßige Lese-, Schreib- und Ausführungsberechtigungen im Unix-Stil, die auf der Benutzer-ID und den Gruppen-IDs basieren. 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-Zugangspunkte verwenden, um Benutzer-ID und Gruppen-IDs zu überschreiben, die vom NFS-Client verwendet werden. Wenn Benutzer versuchen, auf Dateien und Verzeichnisse zuzugreifen, prüft Amazon EFS ihre Benutzer-IDs und Gruppen-IDs, um sicherzustellen, dass jeder Benutzer die Berechtigung zum Zugriff auf die Objekte hat. Amazon EFS verwendet diese IDs auch, um den Eigentümer und Gruppeneigentümer 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 auf Linux-Systemen in der Datei `/etc/passwd` festgelegt. Die numerischen Gruppe-IDs sind in der Datei `/etc/group` enthalten. Diese Dateien definieren die Zuweisungen zwischen Namen und IDs. Außerhalb der EC2-Instance führt Amazon EFS keine Authentifizierung dieser IDs durch, auch nicht für die Root-ID von 0.

Wenn ein Benutzer von zwei verschiedenen EC2-Instances aus auf ein Amazon EFS-Dateisystem zugreift, zeigt sich ein unterschiedliches Verhalten, je nachdem, ob die UID des Benutzers auf diesen Instances dieselbe oder eine andere ist, 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. Beim Zugriff auf das Amazon EFS-Dateisystem von zwei verschiedenen EC2-Instances aus ist die Benutzererfahrung nicht die gleiche.

- 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 auch ein Daemon, ein sogenannter ID-Mapper, der die Zuweisung zwischen Benutzernamen und IDs verwaltet. 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 für eine Clientverbindung aktiviert werden, wenn die AWS Identity and Access Management (AWS IAM)-Identitäts- oder Ressourcenrichtlinie den Zugriff auf die `ClientRootAccess` Aktion nicht zulässt. 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 Dateisystemdatenzugriffs](#) und [Exemplarische Vorgehensweise: Root-Squashing mithilfe der IAM-Autorisierung für NFS-Clients aktivieren](#).

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 Amazon EFS Access Points](#).

Arbeiten mit Amazon EFS Access Points

Amazon EFS Access Points sind anwendungsspezifische Einstiegspunkte in ein EFS-Dateisystem, die das Verwalten des Anwendungszugriffs auf freigegebene Datensätze erleichtern. Zugriffspunkte können eine Benutzeridentität, einschließlich der POSIX-Gruppen des Benutzers, für alle Dateisystemanforderungen erzwingen, die über den Zugriffspunkt 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.

Mithilfe von AWS Identity and Access Management (IAM) -Richtlinien können Sie erzwingen, dass bestimmte Anwendungen einen Zugriffspunkt verwenden. Durch die Kombination von IAM-Richtlinien

mit Zugriffspunkten können Sie ganz einfach einen sicheren Zugriff auf bestimmte Datasets für Ihre Anwendungen bereitstellen.

Note

Sie müssen mindestens ein Mount-Ziel in Ihrem EFS-Dateisystem erstellen, um Access Points verwenden zu können.

Weitere Informationen zum Erstellen eines Zugriffspunkts finden Sie unter [Erstellen und Löschen von Zugangspunkten](#).

Themen

- [Erstellen eines Zugriffspunkts](#)
- [Mounten eines Dateisystems über einen Access Point](#)
- [Durchsetzung einer Benutzeridentität mithilfe eines Access Points](#)
- [Ein Stammverzeichnis mit einem Access Point erzwingen](#)
- [Verwenden von Access Points in IAM-Richtlinien](#)

Erstellen eines Zugriffspunkts

Mithilfe der API, der AWS Command Line Interface (AWS CLI) und der AWS Management Console EFS-API können Sie Access Points für ein vorhandenes Amazon EFS-Dateisystem erstellen. Ein Amazon EFS-Dateisystem kann [maximal 1.000 Access Points](#) haben. Sie können einen vorhandenen Access Point nicht ändern, nachdem er erstellt wurde.

step-by-step Verfahren zum Erstellen eines Access Points finden Sie unter [Erstellen und Löschen von Zugangspunkten](#).

Mounten eines Dateisystems über einen Access Point

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](#).

Durchsetzung einer Benutzeridentität mithilfe eines Access Points

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 Gruppen-IDs — Eine optionale Liste sekundärer Gruppen-IDs.

Wenn die Benutzererzwingung aktiviert ist, ersetzt Amazon EFS die Benutzer- und Gruppen-IDs des NFS-Clients durch die auf dem Access Point konfigurierte Identität für alle Dateisystemvorgänge. 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 Auswertung der Dateisystemberechtigungen die Benutzer-ID, die Gruppen-ID und die sekundären Gruppen-IDs des Zugriffspunkts. EFS ignoriert die IDs des NFS-Clients.

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.

Ein Stammverzeichnis mit einem Access Point erzwingen

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.

Das Stammverzeichnis für einen Access Point erstellen

Wenn im Dateisystem kein Stammverzeichnispfad für einen Access Point vorhanden ist, erstellt Amazon EFS dieses Stammverzeichnis automatisch mit den angegebenen Besitzverhältnissen und Berechtigungen. Amazon EFS erstellt das Stammverzeichnis nicht, wenn Sie bei der Erstellung nicht die Eigentümerschaft und die Berechtigungen für das Verzeichnis 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. Um ein Stammverzeichnis zu erstellen, müssen Sie den Besitz und die Berechtigungen für das Stammverzeichnis konfigurieren, indem Sie beim Erstellen eines Access Points die folgenden Attribute verwenden:

- `OwnerUid`— Die numerische POSIX-Benutzer-ID, die als Besitzer des Stammverzeichnisses verwendet werden soll.

- **OwnerGid**— Die numerische POSIX-Gruppen-ID, die als Eigentümergruppe 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 [Mit Benutzern, Gruppen und Berechtigungen auf Network File System-\(NFS-\)Level arbeiten](#).

Amazon EFS erstellt nur dann ein Zugriffspunkt, wenn das Verzeichnis OwnerUid, OwnerGID und 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 Zugriffspunkt 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, das Stammverzeichnis OwnerUid und die Berechtigungen wurden bei der Erstellung des Access Points angegeben. Wenn das Stammverzeichnis des Access Points bereits vor dem Mount-Zeitpunkt existiert, werden die vorhandenen Berechtigungen nicht vom Access Point überschrieben. Wenn Sie das Stammverzeichnis löschen, erstellt EFS es neu, wenn das Dateisystem das nächste Mal über den Zugriffspunkt gemountet wird.

Note

Wenn Sie das Stammverzeichnis nicht angeben, erstellt Amazon EFS das Stammverzeichnis nicht angeben, erstellt Amazon EFS das Stammverzeichnis nicht angeben. Alle Versuche beim Zugriffspunkt fehl.

Sicherheitsmodell für Access Point-Stammverzeichnisse

Wenn ein Root-Directory-Override aktiv ist, verhält sich Amazon EFS wie ein Linux-NFS-Server, bei dem die `dno_subtree_check` Option aktiviert ist.

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 vorhanden ist, legt EFS keine Dateihandles für Dateien außerhalb des angegebenen Stammverzeichnisses offen. In einigen Fällen kann ein Benutzer jedoch mithilfe eines out-of-band Mechanismus ein Datei-Handle für eine Datei außerhalb seines Access Points erhalten. 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 Access Points 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`.

```
{
  "Version": "2012-10-17",
  "Id": "MyFileSystemPolicy",
  "Statement": [
    {
      "Sid": "App1Access",
      "Effect": "Allow",
      "Principal": { "AWS": "arn:aws:iam::111122223333:role/app1" },
      "Action": [
        "elasticfilesystem:ClientMount",
        "elasticfilesystem:ClientWrite"
      ],
      "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"
    ],
    "Condition": {
        "StringEquals": {
            "elasticfilesystem:AccessPointArn" : "arn:aws:elasticfilesystem:us-
east-1:222233334444:access-point/fsap-89abcdef"
        }
    }
}

```

Blockieren des öffentlichen Zugriffs

Die Funktion für das Sperren des öffentlichen Zugriffs von Amazon EFS bietet Einstellungen für das Verwalten des öffentlichen Zugriffs auf Amazon EFS-Dateisysteme. Standardmäßig erlauben neue Amazon EFS-Dateisysteme keinen öffentlichen Zugriff. Sie können jedoch Dateisystemrichtlinien ändern, um den öffentlichen Zugriff zu ermöglichen.

Themen

- [Blockieren des öffentlichen ZugriffsAWS Transfer Family](#)
- [Die Bedeutung von „öffentlich“](#)

Blockieren des öffentlichen ZugriffsAWS Transfer Family

Wenn Sie Amazon EFS mit verwendenAWS Transfer Familywerden Dateisystemzugriffsanforderungen, die von einem Server der Transfer Family empfangen werden, der einem anderen Konto als dem Dateisystem gehört, gesperrt, wenn das Dateisystem den öffentlichen Zugriff zulässt. Amazon EFS wertet die IAM-Richtlinien des Dateisystems aus und wenn die Richtlinie öffentlich ist, blockiert es die Anforderung. ZugelasseneAWS Transfer FamilyZugriff auf Ihr Dateisystem, aktualisieren Sie Ihre Dateisystemrichtlinie, damit sie nicht als öffentlich angesehen wird.

Note

Die Verwendung von Transfer Family mit Amazon EFS ist standardmäßig deaktiviert für AWS-Kontos mit EFS-Dateisystemen mit Richtlinien, die den öffentlichen Zugriff ermöglichen, die vor dem 6. Januar 2021 erstellt wurden. Um den Zugriff auf Ihr Dateisystem mit Transfer Family zu aktivieren, wenden Sie sich an AWS Support.

Die Bedeutung von „öffentlich“

Bei der Auswertung, ob ein Dateisystem den öffentlichen Zugriff zulässt, geht Amazon EFS davon aus, dass die Dateisystemrichtlinie öffentlich ist. Dann evaluiert es die Dateisystemrichtlinie, um festzustellen, ob sie als nicht-öffentlich eingestuft werden kann. Um als nicht-öffentlich zu gelten, darf eine Dateisystemrichtlinie nur Zugriff für feste Werte (Werte, die keine Platzhalter aufweisen) gewähren, wie einen oder mehrere der folgenden Werte:

- einen Satz von Classless Inter-Domain Routings (CIDRs), unter Verwendung von `aws:SourceIp`. Weitere Informationen zu CIDR finden Sie unter [RFC 4632](#) auf der RFC-Editor-Website.
- Importieren in `&S3;AWS` Ein -Prinzipal, Benutzer, Rolle, oder Service-Prinzipal (z. `B.aws:PrincipalOrgID`)
- `aws:SourceArn`
- `aws:SourceVpc`
- `aws:SourceVpce`
- `aws:SourceOwner`
- `aws:SourceAccount`
- `elasticfilesystem:AccessedViaMountTarget`
- `aws:user`id, outside the pattern `"AROLEID:*`

Nach diesen Regeln gilt die folgende Beispielrichtlinie als öffentlich.

```
{
  "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"
    ]
}
]
}

```

Sie können diese Dateisystemrichtlinie nicht öffentlich machen, indem Sie den EFS-Bedingungsschlüssel verwenden `elasticfilesystem:AccessedViaMountTarget` Stellen Sie auf „true“. Sie können es verwenden `elasticfilesystem:AccessedViaMountTarget` Clients, die angegebenen EFS-Aktionen auf das EFS-Dateisystem mit einem Einhängeziel für das Dateisystem zugreifen, zuzulassen. Die folgende nicht-öffentliche Richtlinie verwendet die `elasticfilesystem:AccessedViaMountTarget` Bedingungsschlüssel auf „true“ festgelegt.

```

{
  "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"
      ],
      "Condition": {
        "Bool": {
          "elasticfilesystem:AccessedViaMountTarget": "true"
        }
      }
    }
  ]
}

```

}

Weitere Informationen über Amazon EFS-Bedingungsschlüssel finden Sie unter [EFS-Bedingungsschlüssel für Clients](#) aus. Weitere Hinweise zum Erstellen von Dateisystemrichtlinien finden Sie unter [Erstellen von Dateisystemrichtlinien](#) aus.

Konformitätsprüfung für Amazon Elastic File System

Informationen darüber, ob ein AWS-Service in den Geltungsbereich bestimmter Compliance-Programme fällt, finden Sie unter [AWS-Services in Geltungsbereich nach Compliance-Programm](#). Wählen Sie das Compliance-Programm, das Sie interessiert. Allgemeine Informationen finden Sie unter [AWS-Compliance-Programme](#).

Sie können Auditberichte von Drittanbietern unter AWS Artifact herunterladen. Weitere Informationen finden Sie unter [Berichte herunterladen in AWS Artifact](#).

Ihre Compliance-Verantwortung bei der Verwendung von AWS-Services ist von der Sensibilität Ihrer Daten, den Compliance-Zielen Ihres Unternehmens und den geltenden Gesetzen und Vorschriften abhängig. AWS stellt die folgenden Ressourcen zur Unterstützung der Compliance bereit:

- [Kurzanleitungen für Sicherheit und Compliance](#) – In diesen Bereitstellungsleitfäden werden Überlegungen zur Architektur erörtert und Schritte zum Bereitstellen von Basisumgebungen auf AWS zur Verfügung gestellt, die auf Sicherheit und Compliance ausgerichtet sind.
- [Erstellung einer Architektur mit HIPAA-konformer Sicherheit und Compliance in Amazon Web Services](#) – In diesem Whitepaper wird beschrieben, wie Unternehmen mithilfe von AWS HIPAA-berechtigte Anwendungen erstellen können.

Note

Nicht alle AWS-Services sind HIPAA-berechtigt. Weitere Informationen finden Sie in der [Referenz für HIPAA-berechtigte Services](#).

- [AWS-Compliance-Ressourcen](#) – Diese Arbeitsbücher und Leitfäden könnten für Ihre Branche und Ihren Standort relevant sein.
- [AWS-Compliance-Leitfäden für Kunden](#) – Verstehen Sie das Modell der geteilten Verantwortung aus dem Blickwinkel der Einhaltung von Vorschriften. In den Leitfäden werden die bewährten Methoden zum Schutz von AWS-Services zusammengefasst und die Leitlinien den Sicherheitskontrollen in verschiedenen Frameworks (einschließlich des National Institute of

Standards and Technology (NIST), des Payment Card Industry Security Standards Council (PCI) und der International Organization for Standardization (ISO)) zugeordnet.

- [Auswertung von Ressourcen mit Regeln](#) im AWS ConfigEntwicklerhandbuch – Der AWS Config-Service bewertet, wie gut Ihre Ressourcenkonfigurationen mit internen Praktiken, Branchenrichtlinien und Vorschriften übereinstimmen.
- [AWS Security Hub](#) – Dieser AWS-Service bietet einen umfassenden Überblick über Ihren Sicherheitsstatus innerhalb von AWS. Security Hub verwendet Sicherheitskontrollen, um Ihre AWS-Ressourcen zu bewerten und Ihre Einhaltung von Sicherheitsstandards und bewährten Methoden zu überprüfen. Eine Liste der unterstützten Services und Kontrollen finden Sie in der [Security-Hub-Steuerungsreferenz](#).
- [AWS Audit Manager](#) – Dieser AWS-Service hilft Ihnen, Ihre AWS-Nutzung kontinuierlich zu überprüfen, um den Umgang mit Risiken und die Compliance von Branchenstandards zu vereinfachen.

Belastbarkeit im Amazon Elastic File System

Die globale AWS-Infrastruktur ist um AWS-Regionen und Availability Zones herum aufgebaut. AWS-Regionen bieten mehrere physisch getrennte und isolierte Availability Zones, die mit einem Netzwerk mit geringer Latenz, hohem Durchsatz und hoher Redundanz verbunden sind. Mithilfe von Availability Zones können Sie Anwendungen und Datenbanken erstellen und ausführen, die automatisch Failover zwischen Zonen ausführen, ohne dass es zu Unterbrechungen kommt. Availability Zones sind besser verfügbar, fehlertoleranter und skalierbarer als herkömmliche Infrastrukturen mit einem oder mehreren Rechenzentren.

Amazon EFS-Dateisysteme sind resistent gegen einen oder mehrere Availability Zone-Ausfälle innerhalb einer AWS-Region aus. Die Mounting-Ziele selbst sind hochverfügbar. Beachten Sie beim Entwurf von Hochverfügbarkeit und Failover auf andere Availability Zones (AZs), dass die IP-Adressen und DNS für Ihre Mounting-Ziele in jeder AZ zwar statisch, aber redundante Komponenten sind, die von mehreren Ressourcen unterstützt werden. Weitere Informationen finden Sie unter [So funktioniert Amazon EFS mit Amazon EC2](#).

Weitere Informationen zu AWS-Regions und Availability Zones finden Sie unter [AWS Globale Infrastruktur](#) aus.

Netzwerkisolierung von Amazon Elastic File System

Als verwalteter Service ist Amazon Elastic File System durch AWS globale Netzwerksicherheit geschützt. Informationen zu AWS-Sicherheitsservices und wie AWS die Infrastruktur schützt, finden Sie unter [AWS-Cloud-Sicherheit](#). Informationen zum Entwerfen Ihrer AWS-Umgebung anhand der bewährten Methoden für die Infrastruktursicherheit finden Sie unter [Infrastrukturschutz](#) im 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.

Außerdem müssen Anforderungen mit einer Zugriffsschlüssel-ID und einem geheimen Zugriffsschlüssel signiert sein, der einem IAM-Prinzipal zugeordnet ist. Alternativ können Sie mit [AWS Security Token Service](#) (AWS STS) temporäre Sicherheitsanmeldeinformationen erstellen, um die Anforderungen zu signieren.

Diese APIs können von jedem Netzwerkstandort aus aufgerufen 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 VPCs aus zu kontrollieren. Dadurch wird der Netzwerkzugriff auf eine bestimmte Amazon EFS-Ressource effektiv nur von der spezifischen VPC innerhalb des AWS Netzwerks isoliert.

Amazon-EFS-Kontingente und -Limits

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 Funktionen von NFSv4.0 und 4.1](#)
- [Weitere Überlegungen](#)

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 [Konsole für Service Quotas](#) können Sie alle Amazon-EFS-Limits einsehen und eine Erhöhung des Kontingents für die Anzahl der EFS-Dateisysteme in einer AWS-Region 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.
- Elastischer Durchsatz pro Dateisystem für alle verbundenen Clients in einer AWS-Region.
- Bereitgestellter Durchsatz pro Dateisystem für alle verbundenen Clients in einem AWS-Region.

In den folgenden Tabellen sind die Standardkontingente für alle Ressourcen aufgeführt, die Sie ändern können.

Anzahl der Dateisysteme pro Kundenkonto

Ressource	Standardkontingent
Anzahl der Dateisysteme für jedes Kundenkonto in einem AWS-Region	1.000

Gesamter Elastic-Standarddurchsatz pro Dateisystem für alle verbundenen Clients in jedem AWS-Region

AWS-Region	Maximaler Lesedurchsatz	Maximaler Schreibdurchsatz (gemessener Durchsatz)
Region USA Ost (Ohio)	20 Gibibyte pro Sekunde (GiBps)	5 GiBps
Region USA Ost (Nord-Virginia)		
Region USA West (Oregon)		
Region Asien-Pazifik (Tokio)		
Region Europa (Irland)		
Alle anderen AWS-Regionen	3 GiBps	1 GiBps

Gesamter bereitgestellter Standarddurchsatz pro Dateisystem für alle verbundenen Clients in jedem 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)		

AWS-Region	Maximaler Lesedurchsatz	Maximaler Schreibdurchsatz (gemessener Durchsatz)
Region Europa (Irland)		
Alle anderen AWS-Regionen	3 GiBps	1 GiBps

Beantragen einer Kontingenterhöhung

Führen Sie die folgenden Schritte aus AWS Support, um eine Erhöhung dieser Kontingente über anzufordern. Das Amazon-EFS-Team überprüft jede Anforderung zur Erhöhung des Kontingents.

So fordern Sie eine Kontingenterhöhung über an 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 Zugangspunkte oder Verbindungen für jedes Dateisystem.
- Bursting-Durchsatzlimits in jeder AWS-Region.

In den folgenden Tabellen sind die allgemeinen Ressourcenkontingente und die Limits für den Bursting-Durchsatz aufgeführt, die nicht geändert werden können.

Allgemeine Ressourcenkontingente, die nicht geändert werden können

Ressource	Kontingent
Anzahl der Zugriffspunkte für jedes Dateisystem	1.000
Anzahl der Verbindungen für jedes Dateisystem	25,000
Anzahl der Mounting-Ziele pro Dateisystem in einer Availability Zone	1
Anzahl der Mountingziele für jede Virtual Private Cloud (VPC)	400
Anzahl der Sicherheitsgruppen pro Mounting-Ziel	5
Anzahl der Tags pro Dateisystem	50
Anzahl der VPCs pro 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 einem anderen AWS-Konto oder einer anderen VPC](#).

Gesamter Bursting-Durchsatz pro Dateisystem für alle verbundenen Clients in jedem 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)		

AWS-Region	Maximaler Lesedurchsatz	Maximaler Schreibdurchsatz
Region Europa (Irland)		
Alle anderen AWS-Regionen	3 GiBps	1 GiBps

Kontingente für NFS-Clients

Die folgenden Kontingente gelten für NFS-Clients, sofern es sich um einen Linux NFSv4.1-Client handelt:

- Der maximale Durchsatz, den Sie für jeden NFS-Client erreichen können, beträgt 500 Mebibyte pro Sekunde (MiBps). 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 können Dateien gleichzeitig öffnen.
- Bis zu 65.536 Dateien werden gleichzeitig auf der Instance 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 Datenmigrationsservice, wenn Sie große Datensätze von On-Premises-NFS-Servern zu EFS migrieren.
- Das NFS-Protokoll unterstützt maximal 16 Gruppen-IDs (GIDs) pro Benutzer und alle weiteren GIDs werden von NFS-Client-Anfragen abgeschnitten. 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
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 Funktionen von NFSv4.0 und 4.1

Obwohl Amazon EFS NFSv2 oder NFSv3 nicht unterstützt, unterstützt es sowohl NFSv4.1 als auch NFSv4.0, mit Ausnahme der folgenden Funktionen:

- pNFS
- 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.

- Zugriffssteuerungslisten (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-Datenaufbewahrung
- SetUID auf Verzeichnissen
- Nicht unterstützte Dateitypen bei Verwendung der Operation CREATE: Blockgeräte (NF4BLK), Zeichengeräte (NF4CHR), Attributverzeichnis (NF4ATTRDIR) und benanntes Attribut (NF4NAMEDATTR)
- 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 AWS-Regionen , in denen Sie Amazon-EFS-Dateisysteme erstellen können, finden Sie unter [Allgemeine AWS-Referenz](#).
- Amazon EFS unterstützt die Mount-Option `nconnect` nicht.
- Sie können ein Amazon-EFS-Dateisystem von Servern in On-Premises-Rechenzentren mit AWS Direct Connect und VPN erstellen. Weitere Informationen finden Sie unter [Mounting mit On-Premises-Clients](#).

Fehlerbehebung bei Amazon EFS

Hier finden Sie Informationen zur Behebung der folgenden Probleme für Amazon Elastic File System (Amazon EFS).

Themen

- [Fehlerbehebung bei Amazon EFS: Allgemeine Probleme](#)
- [Fehlerbehebung bei Fehlern mit Dateivorgängen](#)
- [Beheben von AML- und Kernel-Problemen](#)
- [Beheben von Mountingproblemen](#)
- [Fehlerbehebung bei der Verschlüsselung](#)

Fehlerbehebung bei Amazon EFS: Allgemeine Probleme

Verwenden Sie diese Informationen, um allgemeine Probleme mit Amazon EFS zu beheben. Informationen zur Leistung finden Sie unter [Amazon-EFS-Leistung](#).

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 zu bestätigen, dass Sie berechtigt sind, EFS-Dateisysteme mit den angegebenen Ressourcenbedingungen zu erstellen. Weitere Informationen finden Sie unter [Identitäts- und Zugriffsmanagement für Amazon Elastic File System](#).

Zugriff auf zulässige Dateien im NFS-Dateisystem verweigert

Wenn ein Benutzer, dem mehr als 16 Zugriffsgruppen-IDs (GIDs) zugewiesen sind, versucht, eine Operation auf einem NFS-Dateisystem durchzuführen, könnte ihm der Zugriff auf zulässige Dateien

auf dem Dateisystem verweigert werden. Dieses Problem tritt auf, weil das NFS-Protokoll maximal 16 GIDs pro Benutzer unterstützt und alle zusätzlichen GIDs aus der NFS-Client-Anfrage abgeschnitten werden, wie in [RFC 5531](#) definiert.

Maßnahme

Strukturieren Sie Ihre NFS-Benutzer- und Gruppenzuordnungen so um, dass jedem Benutzer nicht mehr als 16 Zugriffsgruppen (GIDs) zugewiesen werden.

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

Anwendungen, die mehrere Dateien parallel öffnen, können nicht die erwartete Leistungssteigerung der I/O-Parallelisierung nutzen.

Maßnahme

Dieses Problem tritt auf Network File System Version 4 (NFSv4) Clients und auf RHEL 6 Clients mit NFSv4.1 auf, da diese NFS Clients 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 NFSv4.0-Client Öffnungs- und Schließungsanfragen nach Benutzer- und Gruppen-IDs serialisiert. Diese Serialisierung

geschieht auch dann, wenn mehrere Prozesse oder mehrere Threads gleichzeitig Anforderungen ausgeben. Der Client sendet nur jeweils eine Öffnungs- oder Schließungsoperation an den NFS-Server, wenn alle IDs übereinstimmen. 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 die Benutzer-IDs für alle Öffnungsanforderungen unverändert lassen und stattdessen den Satz der Gruppen-IDs modifizieren.
- 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. Die Attributzwischenspeicherung 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.

Fehlerbehebung bei Fehlern mit Dateivorgängen

Wenn Sie auf Amazon EFS-Dateisysteme zugreifen, gelten bestimmte Einschränkungen für die Dateien im Dateisystem. Das Überschreiten dieser Einschränkungen führt zu Fehlern bei Dateivorgängen. Weitere Informationen zu client- und dateibasierten Limits in Amazon EFS finden Sie unter [Kontingente für NFS-Clients](#). Im Folgenden finden Sie einige gängige Fehler bei Dateivorgängen und die Einschränkungen, durch die diese hervorgerufen werden.

Themen

- [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.](#)

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 gleichzeitig Dateien öffnen. Ein Benutzerkonto, das mehrfach angemeldet ist, zählt als ein aktiver Benutzer.
- Bis zu 65.536 Dateien können für eine Instance gleichzeitig geöffnet sein. 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](#).

Befehl schlägt mit „E/A-Fehler“ fehl

Dieser Fehler tritt auf, wenn Sie mit einem der folgenden Probleme konfrontiert werden:

- Für mehr als 65.536 aktive Benutzerkonten für jede Instance 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.

- 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 von Oracle E-Business Suite 32-Bit-Datei-I/O-Schnittstellen 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.

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 die Eigentümerschaft einer Datei oder eines Verzeichnisses mit dem Linux- `chown`-Befehl nicht 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.

Beheben von Mountingproblemen

Nachfolgend finden Sie Informationen zur Fehlerbehebung bei Problemen mit Dateisystem-Mounting für Amazon EFS.

- [Das Dateisystem-Mounting auf der Windows Instance schlägt fehl](#)
- [Zugriff vom Server verweigert](#)
- [Automatisches Mounting schlägt fehl und die Instance reagiert nicht](#)
- [Mounting mehrerer Amazon EFS-Dateisysteme in /etc/fstab schlägt fehl](#)
- [Mounting-Befehl schlägt mit der Fehlermeldung „falscher fs-Typ“ fehl](#)
- [Der Mounting-Befehl schlägt mit der Fehlermeldung „Inkorrekte Mounting-Option“ fehl](#)
- [Mounting mit Zugangspunkt schlägt fehl](#)
- [Das Mounting des Dateisystems schlägt sofort nach der Erstellung des Dateisystems fehl](#)
- [Das Mounting des Dateisystems hängt und schlägt dann mit einem Timeout-Fehler fehl](#)
- [Mounting eines Dateisystems mit NFS unter Verwendung eines DNS-Namens schlägt fehl](#)
- [Das Mounting des Dateisystems schlägt mit der Fehlermeldung „nfs reagiert nicht“.](#)
- [Der Lebenszyklusstatus des Mounting-Ziels hängt fest](#)
- [Der Lebenszyklusstatus des Mounting-Ziels zeigt einen Fehler an](#)
- [Mounting reagiert nicht](#)
- [Gemounteter Client wird nicht mehr verbunden](#)
- [Operationen auf einem neu gemounteten Dateisystem geben den Fehler „bad file handle“ zurück](#)
- [Unmounten eines Dateisystems schlägt fehl](#)

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 mit IAM zu mounten, stellen Sie sicher, dass Sie die `-o iam`-Option im Mounting-Befehl 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 finden Sie unter [Verwendung von IAM zur Steuerung des Dateisystemdatenzugriffs](#) 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 die Amazon EFS-Mountinghilfe verwenden, muss der Befehl `ExecStart` auf EFS statt auf NFS4 verweisen, um Transport Layer Security (TLS) 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 wahrscheinlich, dass Ihre Linux-Verteilung 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 Amazon EFS Access Points](#).

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 programmgesteuert Dateisysteme erstellen und mounten, z. B. mit einer - AWS 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 Mounting-Ziel-Sicherheitsgruppe über eine eingehende Regel verfügt, die den NFS-Zugriff von der EC2-Sicherheitsgruppe zulässt.

Edit inbound rules ✕

Type i	Protocol i	Port Range i	Source i	Description i
NFS	TCP	2049	Custom sg- sg-1a1b1c1d	e.g. SSH for Admin Desktop ✕

Add Rule

NOTE: Any edits made on existing rules will result in the edited rule being deleted and a new rule created with the new details. This will cause traffic that depends on that rule to be dropped for a very brief period of time until the new rule can be created.

Cancel Save

Weitere Informationen finden Sie unter [Erstellen von 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,rsz=1048576,wsz=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 mit seinem DNS-Namen im `mount` Befehl mounten können, müssen Sie Folgendes tun:

- 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 einem anderen AWS-Konto oder einer anderen 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 Mounting-Ziels zeigt einen Fehler an

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 [Erstellen und Verwalten von Mount-Zielen und Sicherheitsgruppen](#).

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 [Verwenden der amazon-efs-utils Tools](#).

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-Mounting-Optionen](#).

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` das das Dateisystem bei der Ausführung von von der Dateisystemhierarchie trennt, und bereinigt dann alle Verweise auf das Dateisystem, 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](#).

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](#)
- [Ein ncrypted-at-rest 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.

Ein ncrypted-at-rest Dateisystem kann nicht erstellt werden

Sie haben versucht, ein neues encrypted-at-rest Dateisystem zu erstellen. Sie erhalten jedoch eine Fehlermeldung, dass nicht verfügbar AWS KMS ist.

Maßnahme

Dieser Fehler kann in dem seltenen Fall auftreten, dass in Ihrem vorübergehend nicht verfügbar AWS KMS ist AWS-Region. Wenn dies der Fall ist, warten Sie, bis AWS KMS wieder die volle Verfügbarkeit erreicht 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 AWS KMS aus 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 immer noch ein Problem auftritt oder wenn Sie ein Problem beim erneuten Aktivieren Ihres Schlüssels haben, wenden Sie sich an den - AWS Support.

Amazon-EFS-API

Die Amazon EFS-API ist ein Netzwerkprotokoll, das auf [HTTP \(RFC 2616\)](#) basiert. Für jeden API-Aufruf senden 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.

Der spezifische Amazon EFS-Vorgang, dem eine bestimmte Anforderung zugeordnet wird, wird durch eine Kombination aus der Methode der Anfrage (GET, PUT, POST oder DELETE) und dem der verschiedenen Muster bestimmt, mit denen die Anforderungs-URI übereinstimmt. Wenn der Vorgang PUT oder POST ist, extrahiert Amazon EFS Aufrufargumente aus dem Request-URI-Pfadsegment, den Abfrageparametern und dem JSON-Objekt im Anforderungstext.

Note

Die Operationsnamen, wie etwa `CreateFileSystem`, sind zwar nicht in den Wire-Daten zu sehen, haben aber Bedeutung in den Richtlinien von AWS Identity and Access Management (IAM). Weitere Informationen finden Sie unter [Identitäts- und Zugriffsmanagement für Amazon Elastic File System](#).

Der Operationsname wird auch verwendet, um Befehle in Befehlszeilentools und Elementen der AWS SDK-APIs zu benennen. Beispielsweise gibt es einen AWS CLI-Befehl mit dem Namen `create-file-system`, der eine Zuordnung zur `CreateFileSystem`-Operation herstellt.

Der Vorgangsname erscheint auch in den AWS CloudTrail Protokollen für Amazon EFS-API-Aufrufe.

API-Endpunkt

Der API-Endpunkt ist der DNS-Name, der in dem HTTP-URI für die API-Aufrufe als Host verwendet wird. Diese API-Endpunkte sind spezifisch für AWS-Regionen und haben die folgende Form.

```
elasticfilesystem.aws-region.amazonaws.com
```

Der Amazon EFS-API API Endpunkt für die Region USA West (Oregon) ist beispielsweise der folgende.

```
elasticfilesystem.us-west-2.amazonaws.com
```

Eine Liste der von AmazonAWS-Region EFS unterstützten Systeme (mit denen Sie Dateisysteme erstellen und verwalten können) finden Sie unter [Amazon Elastic File System](#) in der Allgemeine AWS-Referenz.

Der regionsspezifische API-Endpunkt definiert den Umfang der Amazon EFS-Ressourcen, auf die zugegriffen werden kann, wenn Sie einen API-Aufruf tätigen. Wenn Sie den `DescribeFileSystems` Vorgang beispielsweise mit dem vorherigen Endpunkt aufrufen, erhalten Sie eine Liste der Dateisysteme in der Region USA West (Oregon), die in Ihrem Konto erstellt wurden.

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. Für Beispiele vgl. [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 Elastic File System](#)
- [Aktionen](#)
- [Datentypen](#)

Arbeiten mit der Abfrage-API-Anforderungsrate für Amazon EFS

Amazon EFS-API-Anfragen werden für jedeAWS-Konto 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 variieren AWS-Regionen. Die gestellten API-Anforderungen werden dem Basiswert zugeschrieben 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.

Wiederholungsversuche oder Stapelverarbeitung

Ihre Anwendung muss möglicherweise eine API-Anfrage erneut versuchen, nachdem sie fehlgeschlagen ist, oder um mehrere Ressourcen zu 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 Schlafintervalls

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 [Error Retries and Exponential Backoff AWS 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

Erzeugt einen EFS-Zugriffspunkt. Ein Zugriffspunkt ist eine anwendungsspezifische Ansicht in ein EFS-Dateisystem, die einen Betriebssystembenutzer und eine Gruppe sowie einen Dateisystempfad auf jede über den Zugriffspunkt erfolgte Dateisystemanforderung anwendet. Der Betriebssystembenutzer und die Gruppe überschreiben alle vom NFS-Client bereitgestellten Identitätsinformationen. Der Dateisystempfad wird als Stammverzeichnis des Zugriffspunkts verfügbar gemacht. Anwendungen, die den Access Point verwenden, können nur auf Daten im eigenen Verzeichnis der Anwendung und in beliebigen Unterverzeichnissen zugreifen. Weitere Informationen finden Sie unter [Mounten eines Dateisystems mithilfe von EFS-Zugriffspunkten](#).

Note

Wenn mehrere Anfragen zum Erstellen von Zugriffspunkten auf demselben Dateisystem schnell hintereinander gesendet werden und das Dateisystem fast die Grenze von 1.000 Zugriffspunkten erreicht, kann es bei diesen Anfragen zu einer Drosselung der Antwort kommen. Dadurch wird sichergestellt, dass das Dateisystem die angegebene Zugriffspunktgrenze nicht überschreitet.

Diese Operation erfordert Berechtigungen für die Aktion `elasticfilesystem:CreateAccessPoint`.

Access Points können bei der Erstellung markiert werden. Wenn in der Erstellungsaktion Tags angegeben werden, führt IAM eine zusätzliche Autorisierung für die `elasticfilesystem:TagResource` Aktion durch, um zu überprüfen, ob Benutzer berechtigt sind, Tags zu erstellen. Daher müssen Sie explizite Berechtigungen für die Verwendung der Aktion `elasticfilesystem:TagResource` gewähren. Weitere Informationen finden Sie unter [Erteilen von Berechtigungen für Tag-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 von 1. Höchstlänge = 64 Zeichen.

Pattern: .+

Erforderlich: Ja

FileSystemId

Die ID des EFS-Dateisystems, auf das der Access Point Zugriff gewährt.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge von 128.

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 Dateisystemanfragen angewendet wurden, die über den Access Point gestellt wurden.

Typ: [PosixUser](#) Objekt

Erforderlich: Nein

RootDirectory

Gibt das Verzeichnis im EFS-Dateisystem an, das der Access Point als Stammverzeichnis Ihres Dateisystems für NFS-Clients bereitstellt, die den Access Point verwenden. Die Clients, die den Access Point verwenden, können nur auf das Stammverzeichnis und darunter zugreifen. Wenn das Path angegebene `RootDirectory` > nicht existiert, erstellt Amazon EFS es und wendet die `CreationInfo` Einstellungen an, wenn ein Client eine Verbindung zu einem Access Point herstellt. Wenn Sie `a` angeben `RootDirectory`, müssen Sie die `Path`, und die `CreationInfo`.

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 Zugriffspunkt fehl.

Typ: [RootDirectory](#) Objekt

Erforderlich: Nein

Tags

Erstellt Tags, die dem Access Point zugeordnet sind. Jedes Tag ist ein Schlüssel-Wert-Paar, jeder Schlüssel muss einzigartig sein. Weitere Informationen finden Sie unter [Markieren von AWS-Ressourcen](#) in der Allgemeinen AWS-Referenz.

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 Access Point zugeordnet ist.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge von 128.

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 Access Points.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge von 128.

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 von 1. Höchstlänge = 64 Zeichen.

Pattern: `.+`

FileSystemId

Die ID des EFS-Dateisystems, auf das der Zugriffspunkt angewendet wird.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge von 128.

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 Access Points.

Typ: Zeichenfolge

Zulässige Werte: `creating` | `available` | `updating` | `deleting` | `deleted` | `error`

Name

Der Name des Access Points. Dies ist der Wert des Name Tags.

Typ: Zeichenfolge

OwnerId

Identifiziert den AWS-Konto, dem die Access Point-Ressource gehört.

Typ: Zeichenfolge

Längenbeschränkungen: Die maximale Länge beträgt 14.

Pattern: `^(\d{12})|(\d{4}-\d{4}-\d{4})$`

PosixUser

Die vollständige POSIX-Identität, einschließlich Benutzer-ID, Gruppen-ID und sekundärer Gruppen-IDs auf dem Zugriffspunkt, die für alle Dateioperationen von NFS-Clients verwendet wird, die den Zugriffspunkt verwenden.

Typ: [PosixUser](#) Objekt

RootDirectory

Das Verzeichnis im EFS-Dateisystem, das der Access Point als Stammverzeichnis für NFS-Clients bereitstellt, die den Access Point verwenden.

Typ: [RootDirectory](#) Objekt

Tags

Die mit dem Access Point verknüpften Tags, dargestellt als Array von Tag-Objekten.

Typ: Array von [Tag](#)-Objekten

Fehler

AccessPointAlreadyExists

Wird zurückgegeben, wenn der Access Point, den Sie erstellen möchten, bereits existiert, und zwar mit dem Erstellungstoken, das Sie in der Anfrage angegeben haben.

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>.

HTTP Status Code: 403

BadRequest

Wird zurückgegeben, wenn die Anfrage falsch formatiert ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

HTTP Status Code: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene FileSystemId Wert im Wert des Anforderers nicht vorhanden ist. AWS-Konto

HTTP Status Code: 404

IncorrectFileSystemLifecycleState

Wird zurückgegeben, wenn der Lebenszyklusstatus des Dateisystems nicht „verfügbar“ ist.

HTTP-Statuscode: 409

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

HTTP Status Code: 500

ThrottlingException

Wird zurückgegeben, wenn die CreateAccessPoint API-Aktion zu schnell aufgerufen wird und sich die Anzahl der Access Points im Dateisystem dem [Grenzwert von 120](#) nähert.

HTTP-Statuscode: 429

Weitere Informationen finden Sie auch unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS-SDK für .NET](#)
- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS SDK für V3 JavaScript](#)
- [AWS-SDK für PHP V3](#)
- [AWS-SDK für Python](#)
- [AWS-SDK für 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 sich im Besitz des AWS-Konto des Aufrufers mit dem angegebenen Erstellungs-Token befindet, geht diese Operation wie folgt vor:

- 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 `-maxIOModus` ist ein Leistungstyp der vorherigen Generation, der für hoch parallelisierte Workloads entwickelt wurde, die höhere Latenzen tolerieren können als der `-generalPurposeModus`. Der `-MaxIOModus` wird für One-Zone-Dateisysteme oder Dateisysteme, die Elastic Durchsatz 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 über das Mount-Ziel in 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

One-Zone-Dateisysteme sind nicht in allen Availability Zones in AWS-Regionen verfügbar, 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 in allen AWS-Regionen verfügbar, in denen 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. Beim Erstellen eines verschlüsselten Dateisystems 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 verwenden `KmsKeyId`, müssen Sie den Parameter [CreateFileSystem: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-zA-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 erstellen. Erforderlich, wenn `ThroughputMode` auf `provisioned` festgelegt wird. Gültige Werte sind 1–3414 MiBps, wobei die Obergrenze von der Region abhängt. Um dieses Limit zu erhöhen, wenden Sie sich an den AWS 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 unter [Markieren von AWS-Ressourcen](#) in der Allgemeinen AWS-Referenz.

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 [Angaben 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. Zum Beispiel ist use1-az1 eine Availability Zone ID für die us-east-1 AWS-Region, und sie hat den gleichen Standort in jedem AWS-Konto.

Typ: Zeichenfolge

AvailabilityZoneName

Beschreibt die AWS-Availability-Zone, in der sich das Dateisystem befindet; sie 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: Boolesch

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 AWS KMS key zum Schutz des verschlüsselten Dateisystems.

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

Das AWS-Konto, das 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.

HTTP Status Code: 400

FileSystemAlreadyExists

Wird zurückgegeben, wenn das Dateisystem, das Sie erstellen möchten, bereits existiert, und zwar mit dem Erstellungstoken, das Sie angegeben haben.

HTTP-Statuscode: 409

FileSystemLimitExceeded

Wird zurückgegeben, wenn das AWS-Konto bereits die maximal zulässige Anzahl an Dateisystemen erstellt hat.

HTTP Status Code: 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.

HTTP Status Code: 503

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

HTTP Status Code: 500

ThroughputLimitExceeded

Wird zurückgegeben, wenn der Durchsatzmodus oder die Menge des bereitgestellten Durchsatzes nicht geändert werden können, da die Durchsatzgrenze von 1 024 Mbit/s erreicht wurde.

HTTP Status Code: 400

UnsupportedAvailabilityZone

Wird zurückgegeben, wenn die angeforderte Amazon EFS-Funktion in der angegebenen Availability Zone nicht verfügbar ist.

HTTP Status Code: 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 einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS-SDK für .NET](#)
- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS-SDK für PHP V3](#)
- [AWS-SDK für Python](#)
- [AWS-SDK für Ruby V3](#)

CreateMountTarget

Erstellt ein Mountingziel für ein Dateisystem. Sie können das Dateisystem dann auf EC2-Instances mounten, indem Sie das Mount-Ziel verwenden.

Sie können in jeder Availability Zone in Ihrer VPC ein Mount-Ziel erstellen. Alle EC2-Instances in einer VPC innerhalb einer bestimmten Availability Zone teilen sich ein einziges Mount-Ziel für ein bestimmtes Dateisystem. Wenn Sie mehrere Subnetze in einer Availability Zone haben, legen Sie in einem der Subnetze ein Mount-Ziel an. EC2-Instances müssen nicht im selben Subnetz wie das Mount-Ziel liegen, um auf das Dateisystem zugreifen zu können.

Sie können nur ein Mount-Ziel für ein One Zone-Dateisystem erstellen. Sie müssen dieses Mount-Ziel in derselben Availability Zone erstellen, in der sich das Dateisystem befindet. Verwenden Sie die `AvailabilityZoneId` Eigenschaften `AvailabilityZoneName` und im [DescribeFileSystems](#) Antwortobjekt, um diese Informationen abzurufen. Verwenden Sie bei der Erstellung des Mount-Ziels die Availability Zone, die der Availability Zone des Dateisystems `subnetId` zugeordnet ist.

Weitere Informationen finden Sie unter [Funktionsweise von Amazon EFS](#).

Um ein Mount-Ziel für ein Dateisystem zu erstellen, muss der Lebenszyklusstatus des Dateisystems `launavailable` sein. Weitere Informationen finden Sie unter [DescribeFileSystems](#).

Geben Sie in der Anfrage Folgendes an:

- Die Dateisystem-ID, für die Sie das Mount-Ziel erstellen.
- Eine Subnetz-ID, die Folgendes bestimmt:
 - Die VPC, in der Amazon EFS das Mount-Ziel erstellt
 - Die Availability Zone, in der Amazon EFS das Mount-Ziel erstellt
 - Der IP-Adressbereich, aus dem Amazon EFS die IP-Adresse des Mount-Ziels auswählt (wenn Sie in der Anfrage keine IP-Adresse angeben)

Nachdem das Mount-Ziel erstellt wurde, gibt Amazon EFS eine Antwort zurück, die eine `MountTargetId` und eine `IpAddress` beinhaltet. Diese IP-Adresse verwenden Sie beim Mounten des Dateisystems in einer EC2-Instance. Sie können außerdem den DNS-Namen des Mount-Ziels beim Mounten des Dateisystems verwenden. Die EC2-Instance, in der Sie das Dateisystem mit Hilfe des Mount-Ziels mounten, kann den DNS-Namen des Mount-Ziels in seine IP-Adresse auflösen.

Weitere Informationen finden Sie unter [Funktionsweise: Überblick über die Implementierung](#).

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. Das heißt, wenn das Dateisystem bereits ein oder mehrere Mount-Ziele hat, muss das in der Anfrage zum Hinzufügen eines weiteren Mount-Ziels angegebene Subnetz die folgenden Anforderungen erfüllen:

- Muss zur selben VPC gehören wie die Subnetze der vorhandenen Mount-Ziele.
- Darf nicht in derselben Availability Zone wie eines der Subnetze der vorhandenen Mount-Ziele liegen.

Wenn die Anfrage die Anforderungen erfüllt, geht Amazon EFS wie folgt vor:

- Erstellt ein neues Mount-Ziel im angegebenen Subnetz.
- Erstellt außerdem wie folgt eine neue Netzwerkschnittstelle im Subnetz:
 - Wenn die Anfrage eine `IpAddress` enthält, weist Amazon EFS diese IP-Adresse der Netzwerkschnittstelle zu. Andernfalls weist Amazon EFS eine freie Adresse im Subnetz zu (so wie es der Amazon EC2-CreateNetworkInterface-Aufruf tut, wenn eine Anfrage keine primäre private IP-Adresse angibt).
 - Wenn die Anforderung `SecurityGroups` liefert, ist diese Netzwerkschnittstelle diesen Sicherheitsgruppen zugeordnet. Andernfalls gehört sie zur Standard-Sicherheitsgruppe für die VPC des Subnetzes.
 - Weist die Beschreibung `Mount target fsmt-id for file system fs-id` zu, wobei *fsmt-id* die Mount-Ziel-ID und *fs-id* die FileSystemId ist.
 - Setzt die Eigenschaft `requesterManaged` der Netzwerkschnittstelle auf `true` und den Wert `requesterId` auf EFS.

Jedes Amazon EFS-Mount-Ziel hat eine entsprechende, vom Anforderer verwaltete EC2-Netzwerkschnittstelle. Nachdem die Netzwerkschnittstelle erstellt wurde, setzt Amazon EFS das Feld `NetworkInterfaceId` in der Beschreibung des Mount-Ziels auf die Netzwerkschnittstellen-ID und das Feld `IpAddress` auf seine Adresse. Wenn die Erstellung der Netzwerkschnittstelle fehlschlägt, schlägt die gesamte `CreateMountTarget`-Operation fehl.

Note

Der `CreateMountTarget` Aufruf kehrt erst zurück, nachdem die Netzwerkschnittstelle erstellt wurde. Solange der Status des Einhängeziels jedoch noch `creating`, können Sie den Erstellungsstatus des Einhängeziels überprüfen, indem Sie den

[DescribeMountTargets](#) Vorgang aufrufen, der unter anderem den Status des Einhängeziels zurückgibt.

Wir empfehlen Ihnen, in jeder Availability Zone ein Mounting-Ziel zu erstellen. Es gibt Kostenüberlegungen für die Verwendung eines Dateisystems in einer Availability Zone durch ein Mount-Ziel, das in einer anderen Availability Zone erstellt wurde. Weitere Informationen finden Sie unter [Amazon EFS](#). Wenn Sie immer ein für die Availability Zone der Instance lokales Mounting-Ziel verwenden, vermeiden Sie darüber hinaus ein Teilausfallszenario. Wenn die Availability Zone, in der Ihr Mount-Ziel erstellt wird, herunterfällt, können Sie nicht über dieses Mount-Ziel auf Ihr Dateisystem zugreifen.

Diese Operation erfordert Berechtigungen für die folgende Dateisystemaktion:

- `elasticfilesystem:CreateMountTarget`

Diese Operation erfordert außerdem Berechtigungen für die folgenden Amazon EC2-Aktionen:

- `ec2:DescribeSubnets`
- `ec2:DescribeNetworkInterfaces`
- `ec2:CreateNetworkInterface`

Anforderungssyntax

```
POST /2015-02-01/mount-targets HTTP/1.1
Content-type: application/json
```

```
{
  "FileSystemId": "string",
  "IpAddress": "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 von 128.

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

Gültige IPv4-Adresse innerhalb des Adressbereichs des angegebenen Subnetzes.

Typ: Zeichenfolge

Längenbeschränkungen: Mindestlänge von 7. Maximale Länge von 15.

Pattern: `^[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}$`

Erforderlich: Nein

SecurityGroups

Bis zu fünf VPC-Sicherheitsgruppen-IDs der Form `sg-xxxxxxx`. Diese müssen für dieselbe VPC wie das angegebene Subnetz sein.

Typ: Zeichenfolge-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.

Typ: Zeichenfolge

Längenbeschränkungen: Mindestlänge von 15. Maximale Länge von 47.

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",
  "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 Mount-Ziel befindet. Dies use1-az1 ist beispielsweise eine AZ-ID für die Region us-east-1 und sie hat in jeder Region denselben Standort. AWS-Konto

Typ: Zeichenfolge

AvailabilityZoneName

Der Name der Availability Zone, in der sich das Mount-Ziel befindet. Availability Zones werden den jeweiligen AWS-Konto Namen unabhängig voneinander zugeordnet. 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 von 1. Höchstlänge = 64 Zeichen.

Pattern: . +

FileSystemId

Die ID des Dateisystems, für das das Mount-Ziel bestimmt ist.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge von 128.

Pattern: `^(arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:file-system/fs-[0-9a-f]{8,40}|fs-[0-9a-f]{8,40})$`

IpAddress

Adresse, unter der das Dateisystem mithilfe des Mount-Ziels eingehängt werden kann.

Typ: Zeichenfolge

Längenbeschränkungen: Mindestlänge von 7. Maximale Länge von 15.

Pattern: `^[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}$`

LifeCycleState

Lebenszyklusstatus des Mount-Ziels.

Typ: Zeichenfolge

Zulässige Werte: `creating | available | updating | deleting | deleted | error`

MountTargetId

Vom System zugewiesene Mount-Ziel-ID.

Typ: Zeichenfolge

Längenbeschränkungen: Mindestlänge von 13. Maximale Länge von 45.

Pattern: `^fsmt-[0-9a-f]{8,40}$`

NetworkInterfaceId

Die ID der Netzwerkschnittstelle, die Amazon EFS bei der Erstellung des Mount-Ziels erstellt hat.

Typ: Zeichenfolge

OwnerId

AWS-KontoID, der die Ressource gehört.

Typ: Zeichenfolge

Längenbeschränkungen: Die maximale Länge beträgt 14.

Pattern: `^(\\d{12})|(\\d{4}-\\d{4}-\\d{4})$`

SubnetId

Die ID des Subnetzes des Mount-Ziels.

Typ: Zeichenfolge

Längenbeschränkungen: Mindestlänge von 15. Maximale Länge von 47.

Pattern: `^subnet-[0-9a-f]{8,40}$`

VpcId

Die Virtual Private Cloud (VPC) -ID, in der das Mount-Ziel konfiguriert ist.

Typ: Zeichenfolge

Fehler

AvailabilityZonesMismatch

Wird zurückgegeben, wenn sich die Availability Zone, die für ein Mount-Ziel angegeben wurde, von der Availability Zone unterscheidet, die für One Zone Storage angegeben wurde. Weitere Informationen finden Sie unter [Regionale Speicherredundanz und Speicherredundanz in einer Zone](#).

HTTP Status Code: 400

BadRequest

Wird zurückgegeben, wenn die Anfrage falsch formatiert ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

HTTP Status Code: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene FileSystemId Wert im Wert des Anforderers nicht vorhanden ist. AWS-Konto

HTTP Status Code: 404

IncorrectFileSystemLifeCycleState

Wird zurückgegeben, wenn der Lebenszyklusstatus des Dateisystems nicht „verfügbar“ ist.

HTTP-Statuscode: 409

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

HTTP Status Code: 500

IpAddressInUse

Wird zurückgegeben, wenn in der Anfrage ein angegeben wurdeIpAddress, das bereits im Subnetz verwendet wird.

HTTP-Statuscode: 409

MountTargetConflict

Wird zurückgegeben, wenn das Mount-Ziel eine der angegebenen Einschränkungen aufgrund der vorhandenen Mount-Ziele des Dateisystems verletzen würde.

HTTP-Statuscode: 409

NetworkInterfaceLimitExceeded

Das aufrufende Konto hat den Grenzwert für elastische Netzwerkschnittstellen für den spezifischen Bereich erreichtAWS-Region. 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).

HTTP-Statuscode: 409

NoFreeAddressesInSubnet

Wird zurückgegeben, wenn in der Anfrage nicht angegeben `IpAddress` wurde und es keine freien IP-Adressen im Subnetz gibt.

HTTP-Statuscode: 409

SecurityGroupLimitExceeded

Wird zurückgegeben, wenn die in der Anfrage `SecurityGroups` angegebene Größe größer als fünf ist.

HTTP Status Code: 400

SecurityGroupNotFound

Wird zurückgegeben, wenn eine der angegebenen Sicherheitsgruppen nicht in der Virtual Private Cloud (VPC) des Subnetzes vorhanden ist.

HTTP Status Code: 400

SubnetNotFound

Wird zurückgegeben, wenn in der Anfrage kein Subnetz mit ID `SubnetId` angegeben wurde.

HTTP Status Code: 400

UnsupportedAvailabilityZone

Wird zurückgegeben, wenn die angeforderte Amazon EFS-Funktionalität in der angegebenen Availability Zone nicht verfügbar ist.

HTTP Status Code: 400

Beispiele

Fügt einem Dateisystem ein Mount-Ziel hinzu

Die folgende Anfrage erstellt ein Mount-Ziel für ein Dateisystem. Die Anforderung spezifiziert nur Werte für die erforderlichen `SubnetId` Parameter `FileSystemId` und. Die Anfrage stellt die

optionalen SecurityGroups Parameter IPAddress und nicht bereit. Denn IPAddress der Vorgang verwendet eine der verfügbaren IP-Adressen im angegebenen Subnetz. Und der Vorgang verwendet die der VPC zugeordnete Standardsicherheitsgruppe für. SecurityGroups

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

{
  "MountTargetId": "fsmt-55a4413c",
  "NetworkInterfaceId": "eni-01234567",
  "FileSystemId": "fs-01234567",
  "LifecycleState": "available",
  "SubnetId": "subnet-01234567",
  "OwnerId": "231243201240",
  "IpAddress": "172.31.22.183"
}
```

Fügt einem Dateisystem ein Mount-Ziel hinzu

Die folgende Anfrage spezifiziert alle Anforderungsparameter, um ein Mount-Ziel zu erstellen.

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":"fsmnt-9a13661e",
  "FileSystemId":"fs-01234567",
  "SubnetId":"subnet-fd04ff94",
  "LifeCycleState":"available",
  "IpAddress":"10.0.2.42",
  "NetworkInterfaceId":"eni-1bcb7772"
}
```

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS-SDK für .NET](#)
- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWSSDK für JavaScript V3](#)

- [AWS-SDK für PHP V3](#)
- [AWS-SDK für Python](#)
- [AWS-SDK für Ruby V3](#)

CreateReplicationConfiguration

Erstellt eine Replikationskonfiguration, die ein vorhandenes EFS-Dateisystem in ein neues, schreibgeschütztes Dateisystem repliziert. 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. Das Quelldateisystem kann in einer vorhandenen Replikationskonfiguration kein Zieldateisystem sein.
- **AWS-Region** – Die AWS-Region, in der das Zieldateisystem erstellt wird. Die Amazon EFS-Replikation ist in allen AWS-Regionen verfügbar, in denen EFS verfügbar ist. Die Region muss aktiviert sein. Weitere Informationen finden Sie unter [Verwalten von AWS-Regionen](#) in der Allgemeinen Referenz zu AWS.
- **Konfiguration des Zieldateisystems** – Die Konfiguration des Zieldateisystems, in das das Quelldateisystem repliziert wird. In einer Replikationskonfiguration kann es nur ein Zieldateisystem geben.

Zu den Parametern für die Replikationskonfiguration gehören:

- **Dateisystem-ID** – Die ID des Zieldateisystems für die Replikation. Wenn keine ID angegeben wird, erstellt EFS ein neues Dateisystem mit den Standardeinstellungen. Für bestehende Dateisysteme muss der Replikationsschutz des Dateisystems vor Überschreibung deaktiviert werden. Weitere Informationen finden Sie unter [Replizieren in ein vorhandenes Dateisystem](#).
- **Availability Zone** – Wenn Sie möchten, dass das Zieldateisystem One-Zone-Speicher verwendet, müssen Sie die Availability Zone angeben, in der das Dateisystem erstellt werden soll. Weitere Informationen finden Sie unter [EFS-Dateisystemtypen](#) im Amazon EFS-Benutzerhandbuch.
- **Verschlüsselung** – Alle Zieldateisysteme werden mit aktivierter Verschlüsselung im Ruhezustand erstellt. Sie können den Schlüssel AWS Key Management Service (AWS KMS) 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.

Note

Der KMS-Schlüssel kann nicht geändert werden, nachdem das Dateisystem erstellt wurde.

Für neue Zieldateisysteme sind die folgenden Eigenschaften standardmäßig festgelegt:

- **Leistungsmodus** – Der Leistungsmodus des Zielsystems entspricht dem des Quellsystems, es sei denn, das Zielsystem verwendet EFS One-Zone-Speicher. In diesem Fall wird der Allzweck-Leistungsmodus verwendet. Der Leistungsmodus kann nicht geändert werden.
- **Durchsatzmodus** – Der Durchsatzmodus des Zielsystems entspricht dem des Quellsystems. Nachdem das Dateisystem erstellt wurde, können Sie den Durchsatzmodus ändern.
- **Lebenszyklusverwaltung** – Das Lebenszyklusmanagement ist auf dem Zielsystem nicht aktiviert. Nachdem das Zielsystem erstellt wurde, können Sie das Lebenszyklusmanagement aktivieren.
- **Automatische Backups** – Automatische tägliche Backups sind im Zielsystem aktiviert. Diese Einstellung kann nicht geändert werden, nachdem das Dateisystem erstellt wurde.

Weitere Informationen finden Sie unter [Amazon EFS-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"
    }
  ]
}
```

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,
      "Region": "string",
      "Status": "string"
    }
  ],
  "OriginalSourceFileSystemArn": "string",
  "SourceFileSystemArn": "string",
```

```
"SourceFileSystemId": "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})$`

SourceFileSystemRegion

Die AWS-Region, in der 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.

HTTP Status Code: 400

ConflictException

Wird zurückgegeben, wenn das Quelldateisystem in einer Replikation verschlüsselt, das Zieldateisystem jedoch unverschlüsselt ist.

HTTP-Statuscode: 409

FileSystemLimitExceeded

Wird zurückgegeben, wenn das AWS-Konto bereits die maximal zulässige Anzahl an Dateisystemen erstellt hat.

HTTP Status Code: 403

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene `FileSystemId`-Wert nicht im AWS-Konto des Anfragenden existiert.

HTTP Status Code: 404

IncorrectFileSystemLifecycleState

Wird zurückgegeben, wenn der Lebenszyklusstatus des Dateisystems nicht „verfügbar“ ist.

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.

HTTP Status Code: 503

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

HTTP Status Code: 500

ReplicationNotFound

Wird zurückgegeben, wenn das angegebene Dateisystem keine Replikationskonfiguration aufweist.

HTTP Status Code: 404

ThroughputLimitExceeded

Wird zurückgegeben, wenn der Durchsatzmodus oder die Menge des bereitgestellten Durchsatzes nicht geändert werden können, da die Durchsatzgrenze von 1024 MiB/s erreicht wurde.

HTTP Status Code: 400

UnsupportedAvailabilityZone

Wird zurückgegeben, wenn die angeforderte Amazon EFS-Funktion in der angegebenen Availability Zone nicht verfügbar ist.

HTTP Status Code: 400

ValidationException

Wird zurückgegeben, wenn der AWS Backup-Dienst in der AWS-Region, in der die Anfrage gestellt wurde, nicht verfügbar ist.

HTTP Status Code: 400

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS-SDK für .NET](#)
- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS-SDK für PHP V3](#)
- [AWS-SDK für Python](#)
- [AWS-SDK für 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 hinzuzufügenden Tag-Objekten. 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.

HTTP Status Code: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene FileSystemId-Wert nicht im AWS-Konto des Anfragenden existiert.

HTTP Status Code: 404

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

HTTP Status Code: 500

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS-SDK für .NET](#)
- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS-SDK für PHP V3](#)
- [AWS-SDK für Python](#)
- [AWS-SDK für 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 für `AccessPointId` angegebene Wert im AWS-Konto des Anforderers nicht vorhanden ist.

HTTP Status Code: 404

BadRequest

Wird zurückgegeben, wenn die Anforderung falsch formatiert ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

HTTP Status Code: 400

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

HTTP Status Code: 500

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS-SDK für .NET](#)
- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS-SDK für PHP V3](#)
- [AWS-SDK für Python](#)
- [AWS-SDK für 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. Wenn Sie ein Dateisystem mit der AWS-Konsole löschen, wird dieser Schritt für Sie übernommen.

Note

Dateisysteme, die Teil einer EFS-Replikationskonfiguration sind, können nicht gelöscht werden. 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 finden 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.

HTTP Status Code: 400

FileSystemInUse

Wird zurückgegeben, wenn ein Dateisystem Mountingziele hat.

HTTP-Statuscode: 409

FileSystemNotFound

Wird zurückgegeben, wenn der für `FileSystemId` angegebene Wert im AWS-Konto des Anforderers nicht vorhanden ist.

HTTP Status Code: 404

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

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 einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS-SDK für .NET](#)

- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS-SDK für PHP V3](#)
- [AWS-SDK für Python](#)
- [AWS-SDK für 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.

HTTP Status Code: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene FileSystemId-Wert nicht im AWS-Konto des Anfragenden existiert.

HTTP Status Code: 404

IncorrectFileSystemLifecycleState

Wird zurückgegeben, wenn der Lebenszyklusstatus des Dateisystems nicht „verfügbar“ ist.

HTTP-Statuscode: 409

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

HTTP Status Code: 500

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS-SDK für .NET](#)
- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS-SDK für PHP V3](#)
- [AWS-SDK für Python](#)

- [AWS-SDK für 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 der VPC mounten, indem Sie ein anderes Mountingziel 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.

Die Operation erfordert außerdem Berechtigungen für die folgende Amazon EC2-Aktion in der Netzwerkschnittstelle des Mountingsziels:

- `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.

HTTP Status Code: 400

DependencyTimeout

Bei dem Service ist beim Versuch, der Anforderung nachzukommen, eine Zeitüberschreitung aufgetreten, und der Client sollte den Aufruf wiederholen.

HTTP Status Code: 504

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

HTTP Status Code: 500

MountTargetNotFound

Wird zurückgegeben, wenn kein Mountingziel mit der angegebenen ID im AWS-Konto des Aufrufers gefunden wurde.

HTTP Status Code: 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 einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS-SDK für .NET](#)
- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)

- [AWS SDK für JavaScript V3](#)
- [AWS-SDK für PHP V3](#)
- [AWS-SDK für Python](#)
- [AWS-SDK für Ruby V3](#)

DeleteReplicationConfiguration

Löscht eine Replikationskonfiguration. Durch das Löschen einer Replikationskonfiguration wird der Replikationsvorgang beendet. Nach dem Löschen einer Replikationskonfiguration wird das Zielsystem 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 HTTP/1.1
```

URI-Anfrageparameter

Die Anforderung verwendet die folgenden URI-Parameter.

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.

HTTP Status Code: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene `FileSystemId`-Wert nicht im AWS-Konto des Anfragenden existiert.

HTTP Status Code: 404

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

HTTP Status Code: 500

ReplicationNotFound

Wird zurückgegeben, wenn das angegebene Dateisystem keine Replikationskonfiguration aufweist.

HTTP Status Code: 404

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS-SDK für .NET](#)
- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS-SDK für PHP V3](#)

- [AWS-SDK für Python](#)
- [AWS-SDK für 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 Tag- und sonstigen Einschränkungen finden Sie unter [Tag-Einschränkungen](#) im Benutzerhandbuch zu AWS Billing and Cost Management.

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.

HTTP Status Code: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene FileSystemId-Wert nicht im AWS-Konto des Anfragenden existiert.

HTTP Status Code: 404

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

HTTP Status Code: 500

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS-SDK für .NET](#)
- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS-SDK für PHP V3](#)
- [AWS-SDK für Python](#)
- [AWS-SDK für 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 die NextMarker in der nachfolgenden Anforderung 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 für `AccessPointId` angegebene Wert im AWS-Konto des Anforderers nicht vorhanden ist.

HTTP Status Code: 404

BadRequest

Wird zurückgegeben, wenn die Anforderung falsch formatiert ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

HTTP Status Code: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene `FileSystemId`-Wert nicht im AWS-Konto des Anfragenden existiert.

HTTP Status Code: 404

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

HTTP Status Code: 500

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS-SDK für .NET](#)
- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS-SDK für PHP V3](#)

- [AWS-SDK für Python](#)
- [AWS-SDK für 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 VersionAWS-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) Beim Abrufen der Kontoeinstellungen können Sie optional den `MaxItems` Parameter 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 ihn `NextToken` in einer nachfolgenden Anfrage verwenden, um die nächste Seite mit AWS-Konto Einstellungen abzurufen, wenn die Antwort-Payload paginiert wurde.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 1. 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 das NextToken in der nachfolgenden Anfrage verwenden, um die zusätzlichen Beschreibungen abzurufen.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 1. Maximale Länge beträgt 128 Zeichen.

Pattern: . +

[ResourceIdPreference](#)

Beschreibt die Einstellung für die Ressourcen-ID, die dem Benutzer AWS-Konto zugeordnet ist, der die Anfrage stellt, in der aktuellen AWS-Region Version.

Typ: [ResourceIdPreference](#) Objekt

Fehler

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

HTTP Status Code: 500

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS-SDK für .NET](#)
- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWSSDK für JavaScript V3](#)
- [AWS-SDK für PHP V3](#)
- [AWS-SDK für Python](#)
- [AWS-SDK für 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.

HTTP Status Code: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene FileSystemId-Wert nicht im AWS-Konto des Anfragenden existiert.

HTTP Status Code: 404

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

HTTP Status Code: 500

PolicyNotFound

Wird zurückgegeben, wenn die standardmäßige Dateisystemrichtlinie für das angegebene EFS-Dateisystemrichtlinie gilt.

HTTP Status Code: 404

ValidationException

Wird zurückgegeben, wenn der AWS Backup-Dienst in der AWS-Region, in der die Anfrage gestellt wurde, nicht verfügbar ist.

HTTP Status Code: 400

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS-SDK für .NET](#)
- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS-SDK für PHP V3](#)
- [AWS-SDK für Python](#)
- [AWS-SDK für 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.

HTTP Status Code: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene FileSystemId-Wert nicht im AWS-Konto des Anfragenden existiert.

HTTP Status Code: 404

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

HTTP Status Code: 500

PolicyNotFound

Wird zurückgegeben, wenn die standardmäßige Dateisystemrichtlinie für das angegebene EFS-Dateisystemrichtlinie gilt.

HTTP Status Code: 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 einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS-SDK für .NET](#)
- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS-SDK für PHP V3](#)
- [AWS-SDK für Python](#)
- [AWS-SDK für Ruby V3](#)

DescribeFileSystems

Gibt die Beschreibung eines bestimmten Amazon EFS-Dateisystems zurück, wenn entweder das Dateisystem `CreationToken` oder die `FileSystemId` angegeben ist. Andernfalls gibt sie Beschreibungen aller Dateisysteme zurück, die sich im Besitz des AWS-Konto des Aufrufers in der AWS-Region des von Ihnen aufgerufenen Endpunkts befinden.

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.

HTTP Status Code: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene FileSystemId-Wert nicht im AWS-Konto des Anfragenden existiert.

HTTP Status Code: 404

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

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 einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS-SDK für .NET](#)
- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS-SDK für PHP V3](#)
- [AWS-SDK für Python](#)
- [AWS-SDK für Ruby V3](#)

DescribeLifecycleConfiguration

Gibt das aktuelle LifecycleConfiguration-Objekt für das angegebene Amazon 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.

HTTP Status Code: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene FileSystemId-Wert nicht im AWS-Konto des Anfragenden existiert.

HTTP Status Code: 404

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

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 einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS-SDK für .NET](#)
- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS-SDK für PHP V3](#)
- [AWS-SDK für Python](#)
- [AWS-SDK für 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.

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",
      "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 für `AccessPointId` angegebene Wert im AWS-Konto des Anforderers nicht vorhanden ist.

HTTP Status Code: 404

`BadRequest`

Wird zurückgegeben, wenn die Anforderung falsch formatiert ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

HTTP Status Code: 400

`FileSystemNotFound`

Wird zurückgegeben, wenn der angegebene `FileSystemId`-Wert nicht im AWS-Konto des Anfragenden existiert.

HTTP Status Code: 404

`InternalServerError`

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

HTTP Status Code: 500

MountTargetNotFound

Wird zurückgegeben, wenn kein Mountingziel mit der angegebenen ID im AWS-Konto des Aufrufers gefunden wurde.

HTTP Status Code: 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 einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS-SDK für .NET](#)
- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS-SDK für PHP V3](#)
- [AWS-SDK für Python](#)
- [AWS-SDK für 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.

HTTP Status Code: 400

IncorrectMountTargetState

Wird zurückgegeben, wenn das Mountingziel nicht den richtigen Status für die Operation aufweist.

HTTP-Statuscode: 409

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

HTTP Status Code: 500

MountTargetNotFound

Wird zurückgegeben, wenn kein Mountingziel mit der angegebenen ID im AWS-Konto des Aufrufers gefunden wurde.

HTTP Status Code: 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 einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS-SDK für .NET](#)

- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS-SDK für PHP V3](#)
- [AWS-SDK für Python](#)
- [AWS-SDK für Ruby V3](#)

DescribeReplicationConfigurations

Ruft die Replikationskonfiguration für ein bestimmtes Dateisystem ab. Wenn kein Dateisystem angegeben ist, werden alle Replikationskonfigurationen für das AWS-Konto in einer AWS-Region 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.

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,
          "Region": "string",
          "Status": "string"
        }
      ],
      "OriginalSourceFileSystemArn": "string",
      "SourceFileSystemArn": "string",
      "SourceFileSystemId": "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.

HTTP Status Code: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene FileSystemId-Wert nicht im AWS-Konto des Anfragenden existiert.

HTTP Status Code: 404

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

HTTP Status Code: 500

ReplicationNotFound

Wird zurückgegeben, wenn das angegebene Dateisystem keine Replikationskonfiguration aufweist.

HTTP Status Code: 404

ValidationException

Wird zurückgegeben, wenn der AWS Backup-Dienst in der AWS-Region, in der die Anfrage gestellt wurde, nicht verfügbar ist.

HTTP Status Code: 400

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS-SDK für .NET](#)
- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS-SDK für PHP V3](#)
- [AWS-SDK für Python](#)
- [AWS-SDK für 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.

HTTP Status Code: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene `FileSystemId`-Wert nicht im AWS-Konto des Anfragenden existiert.

HTTP Status Code: 404

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

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 einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS-SDK für .NET](#)
- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS-SDK für PHP V3](#)
- [AWS-SDK für Python](#)
- [AWS-SDK für 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 für `AccessPointId` angegebene Wert im AWS-Konto des Anforderers nicht vorhanden ist.

HTTP Status Code: 404

BadRequest

Wird zurückgegeben, wenn die Anforderung falsch formatiert ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

HTTP Status Code: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene `FileSystemId`-Wert nicht im AWS-Konto des Anfragenden existiert.

HTTP Status Code: 404

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

HTTP Status Code: 500

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS-SDK für .NET](#)
- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)

- [AWS SDK für JavaScript V3](#)
- [AWS-SDK für PHP V3](#)
- [AWS-SDK für Python](#)
- [AWS-SDK für 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

Ein Array von bis zu fünf 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.

HTTP Status Code: 400

IncorrectMountTargetState

Wird zurückgegeben, wenn das Mountingziel nicht den richtigen Status für die Operation aufweist.

HTTP-Statuscode: 409

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

HTTP Status Code: 500

MountTargetNotFound

Wird zurückgegeben, wenn kein Mountingziel mit der angegebenen ID im AWS-Konto des Aufrufers gefunden wurde.

HTTP Status Code: 404

SecurityGroupLimitExceeded

Wird zurückgegeben, wenn die in der Anforderung `SecurityGroups` angegebene Größe fünf überschreitet.

HTTP Status Code: 400

SecurityGroupNotFound

Wird zurückgegeben, wenn eine der angegebenen Sicherheitsgruppen nicht in der Virtual Private Cloud (VPC) des Subnetzes vorhanden ist.

HTTP Status Code: 400

Beispiele

Ersetzt die 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 einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS-SDK für .NET](#)
- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS-SDK für PHP V3](#)
- [AWS-SDK für Python](#)
- [AWS-SDK für Ruby V3](#)

PutAccountPreferences

Verwenden Sie diesen Vorgang, um in der aktuellen AWS-Region Kontopräferenz die Verwendung langer 17-stelliger (63 Bit) oder kurzer 8-stelliger (32-Bit) Ressourcen-IDs für neue EFS-Dateisystem- und Mount-Zielressourcen festzulegen. 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 lange Ressourcen-IDs umstellt. Weitere Informationen finden Sie unter [Amazon EFS-Ressourcen-IDs verwalten](#).

Note

Ab Oktober 2021 erhalten Sie eine Fehlermeldung, wenn Sie versuchen, die Kontopräferenz so einzustellen, dass die Ressourcen-ID im kurzen 8-stelligen Format verwendet wird. Wenden Sie sich an den AWS Support, wenn Sie eine Fehlermeldung erhalten und kurze 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 anAWS-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

Ab Oktober 2021 erhalten Sie eine Fehlermeldung, wenn Sie die Kontopräferenz auf `SHORT_ID` setzen. Wenden Sie sich an den AWS Support, wenn Sie eine Fehlermeldung erhalten und kurze 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 falsch formatiert ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

HTTP Status Code: 400

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

HTTP Status Code: 500

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS-SDK für .NET](#)
- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWSSDK für JavaScript V3](#)
- [AWS-SDK für PHP V3](#)
- [AWS-SDK für Python](#)
- [AWS-SDK für 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.

HTTP Status Code: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene FileSystemId-Wert nicht im AWS-Konto des Anfragenden existiert.

HTTP Status Code: 404

IncorrectFileSystemLifecycleState

Wird zurückgegeben, wenn der Lebenszyklusstatus des Dateisystems nicht „verfügbar“ ist.

HTTP-Statuscode: 409

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

HTTP Status Code: 500

ValidationException

Wird zurückgegeben, wenn der AWS Backup-Dienst in der AWS-Region, in der die Anfrage gestellt wurde, nicht verfügbar ist.

HTTP Status Code: 400

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS-SDK für .NET](#)
- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS-SDK für PHP V3](#)
- [AWS-SDK für Python](#)
- [AWS-SDK für Ruby V3](#)

PutFileSystemPolicy

Wendet eine `FileSystemPolicy` auf ein Amazon EFS-Dateisystem an. Eine Dateisystemrichtlinie ist eine auf IAM-Ressourcen basierende Richtlinie und kann mehrere Richtlinienanweisungen 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 [Standard-EFS-Dateisystemrichtlinie](#).

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 eine Dateisystemrichtlinie besteht, 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.

HTTP Status Code: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene FileSystemId-Wert nicht im AWS-Konto des Anfragenden existiert.

HTTP Status Code: 404

IncorrectFileSystemLifecycleState

Wird zurückgegeben, wenn der Lebenszyklusstatus des Dateisystems nicht „verfügbar“ ist.

HTTP-Statuscode: 409

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

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.

HTTP Status Code: 400

Beispiele

Erstellen eines EFS FileSystemPolicy

Mit der folgenden Anforderung wird eine FileSystemPolicy erstellt, mit der alle AWS-Prinzipale das angegebene EFS-Dateisystem mit Lese- und Schreibberechtigungen mounten können.

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 einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS-SDK für .NET](#)
- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)

- [AWS SDK für JavaScript V3](#)
- [AWS-SDK für PHP V3](#)
- [AWS-SDK für Python](#)
- [AWS-SDK für 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 darf entweder nicht festgelegt werden oder muss später als TransitionToIA sein.

Note

Die Speicherklasse Archive ist nur für Dateisysteme verfügbar, die den Elastic-Durchsatzmodus und den Allzweck-Leistungsmodus 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 Lebenszyklusmanagement 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 darf entweder nicht festgelegt werden oder muss später als TransitionToIA sein.

Note

Die Speicherklasse Archive ist nur für Dateisysteme verfügbar, die den Elastic-Durchsatzmodus und den Allzweck-Leistungsmodus 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.

HTTP Status Code: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene FileSystemId-Wert nicht im AWS-Konto des Anfragenden existiert.

HTTP Status Code: 404

IncorrectFileSystemLifecycleState

Wird zurückgegeben, wenn der Lebenszyklusstatus des Dateisystems nicht „verfügbar“ ist.

HTTP-Statuscode: 409

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

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-Durchsatzmodus und den Allzweck-Leistungsmodus 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-Anforderung

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 einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS-SDK für .NET](#)
- [AWS-SDK für C++](#)

- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS-SDK für PHP V3](#)
- [AWS-SDK für Python](#)
- [AWS-SDK für 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 hinzuzufügenden Tag-Objekten. 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 für `AccessPointId` angegebene Wert im AWS-Konto des Anforderers nicht vorhanden ist.

HTTP Status Code: 404

BadRequest

Wird zurückgegeben, wenn die Anforderung falsch formatiert ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

HTTP Status Code: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene `FileSystemId`-Wert nicht im AWS-Konto des Anfragenden existiert.

HTTP Status Code: 404

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

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"
    }
  ]
}
```

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 einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS-SDK für .NET](#)
- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS-SDK für PHP V3](#)
- [AWS-SDK für Python](#)
- [AWS-SDK für 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 für `AccessPointId` angegebene Wert im AWS-Konto des Anforderers nicht vorhanden ist.

HTTP Status Code: 404

BadRequest

Wird zurückgegeben, wenn die Anforderung falsch formatiert ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

HTTP Status Code: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene `FileSystemId`-Wert nicht im AWS-Konto des Anfragenden existiert.

HTTP Status Code: 404

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

HTTP Status Code: 500

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS-SDK für .NET](#)
- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS-SDK für PHP V3](#)
- [AWS-SDK für Python](#)
- [AWS-SDK für 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 Region abhängt. Um dieses Limit zu erhöhen, wenden Sie sich an den AWS 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. Zum Beispiel ist use1-az1 eine Availability Zone ID für die us-east-1 AWS-Region, und sie hat den gleichen Standort in jedem AWS-Konto.

Typ: Zeichenfolge

AvailabilityZoneName

Beschreibt die AWS-Availability-Zone, in der sich das Dateisystem befindet; sie 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: Boolesch

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 AWS KMS key zum Schutz des verschlüsselten Dateisystems.

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

Das AWS-Konto, das 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.

Typ: 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.

HTTP Status Code: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene `FileSystemId`-Wert nicht im AWS-Konto des Anfragenden existiert.

HTTP Status Code: 404

IncorrectFileSystemLifecycleState

Wird zurückgegeben, wenn der Lebenszyklusstatus des Dateisystems nicht „verfügbar“ ist.

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.

HTTP Status Code: 503

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

HTTP Status Code: 500

ThroughputLimitExceeded

Wird zurückgegeben, wenn der Durchsatzmodus oder die Menge des bereitgestellten Durchsatzes nicht geändert werden können, da die Durchsatzgrenze von 1 024 Mbit/s erreicht wurde.

HTTP Status Code: 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.

HTTP-Statuscode: 429

Weitere Informationen finden Sie auch unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS-SDK für .NET](#)
- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS-SDK für PHP V3](#)
- [AWS-SDK für Python](#)
- [AWS-SDK für 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 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**

Fehler

BadRequest

Wird zurückgegeben, wenn die Anfrage fehlerhaft ist oder einen Fehler enthält, z. B. einen ungültigen Parameterwert oder einen fehlenden erforderlichen Parameter.

HTTP Status Code: 400

FileSystemNotFound

Wird zurückgegeben, wenn der angegebene FileSystemId-Wert nicht im AWS-Konto des Anfragenden existiert.

HTTP Status Code: 404

IncorrectFileSystemLifecycleState

Wird zurückgegeben, wenn der Lebenszyklusstatus des Dateisystems nicht „verfügbar“ ist.

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.

HTTP Status Code: 503

InternalServerError

Wird zurückgegeben, wenn auf der Serverseite ein Fehler aufgetreten ist.

HTTP Status Code: 500

ReplicationAlreadyExists

Wird zurückgegeben, wenn das Dateisystem bereits in einer Replikationskonfiguration enthalten ist. >

HTTP-Statuscode: 409

ThroughputLimitExceeded

Wird zurückgegeben, wenn der Durchsatzmodus oder die Menge des bereitgestellten Durchsatzes nicht geändert werden können, da die Durchsatzgrenze von 1 024 Mbit/s erreicht wurde.

HTTP Status Code: 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.

HTTP-Statuscode: 429

Weitere Informationen finden Sie auch unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-Befehlszeilenschnittstelle](#)
- [AWS-SDK für .NET](#)
- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS SDK für JavaScript V3](#)
- [AWS-SDK für PHP V3](#)
- [AWS-SDK für Python](#)

- [AWS-SDK für Ruby V3](#)

Datentypen

Die folgenden Datentypen werden unterstützt:

- [AccessPointDescription](#)
- [BackupPolicy](#)
- [CreationInfo](#)
- [Destination](#)
- [DestinationToCreate](#)
- [FileSystemDescription](#)
- [FileSystemProtectionDescription](#)
- [FileSystemSize](#)
- [LifecyclePolicy](#)
- [MountTargetDescription](#)
- [PosixUser](#)
- [ReplicationConfigurationDescription](#)
- [ResourceIdPreference](#)
- [RootDirectory](#)
- [Tag](#)

AccessPointDescription

Stellt eine Beschreibung eines EFS-Dateisystem-Zugriffspunkts bereit.

Inhalt

AccessPointArn

Der eindeutige Amazon-Ressourcenname (ARN), der dem Access Point zugeordnet ist.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge von 128.

Pattern: `^arn:aws[-a-z]*:elasticfilesystem:[0-9a-z-:]+:access-point/fsap-[0-9a-f]{8,40}$`

Erforderlich: Nein

AccessPointId

Die von Amazon EFS zugewiesene ID des Access Points.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge von 128.

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 von 1. Höchstlänge = 64 Zeichen.

Pattern: `.+`

Erforderlich: Nein

FileSystemId

Die ID des EFS-Dateisystems, auf das der Zugriffspunkt angewendet wird.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge von 128.

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

Identifiziert die Lebenszyklusphase des Access Points.

Typ: Zeichenfolge

Zulässige Werte: `creating | available | updating | deleting | deleted | error`

Erforderlich: Nein

Name

Der Name des Access Points. 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: Die maximale Länge beträgt 14.

Pattern: `^(\\d{12})|(\\d{4}-\\d{4}-\\d{4})$`

Erforderlich: Nein

PosixUser

Die vollständige POSIX-Identität, einschließlich Benutzer-ID, Gruppen-ID und sekundärer Gruppen-IDs auf dem Zugriffspunkt, die für alle Dateioperationen von NFS-Clients verwendet wird, die den Zugriffspunkt verwenden.

Typ: [PosixUser](#) Objekt

Erforderlich: Nein

RootDirectory

Das Verzeichnis im EFS-Dateisystem, das der Access Point als Stammverzeichnis für NFS-Clients bereitstellt, die den Access Point verwenden.

Typ: [RootDirectory](#) Objekt

Erforderlich: Nein

Tags

Die mit dem Access Point verknüpften Tags, dargestellt als Array von Tag-Objekten.

Typ: Array von [Tag](#)-Objekten

Erforderlich: Nein

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS-SDK für Ruby V3](#)

BackupPolicy

Die Backup-Richtlinie für das Dateisystem, das 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 sichert das Dateisystem automatisch.
- **ENABLING**— EFS aktiviert automatische Backups für das Dateisystem.
- **DISABLED**— Automatische Backups für das Dateisystem sind deaktiviert.
- **DISABLING**— EFS deaktiviert automatische Backups für das Dateisystem.

Typ: Zeichenfolge

Zulässige Werte: `ENABLED` | `ENABLING` | `DISABLED` | `DISABLING`

Erforderlich: Ja

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS-SDK für Ruby V3](#)

CreationInfo

Erforderlich, wenn das angegebene `RootDirectory > Path` nicht vorhanden ist. Gibt die POSIX-IDs und Berechtigungen an, die für das `RootDirectory > Path` des Zugriffspunkts 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 Root-Verzeichnis, 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 Zugriffspunkt 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 einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-SDK für C++](#)
- [AWS-SDK for Go](#)
- [AWS-SDK für Java V2](#)
- [AWS SDK für Ruby V3](#)

Destination

Beschreibt das Zielsystem in der Replikationskonfiguration.

Inhalt

FileSystemId

Die ID des Amazon EFS-Zielsystems.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge von 128.

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 von 1. Höchstlänge = 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 des EFS-Zielsystems.

- Der **Paused** Status ist das Ergebnis einer Abmeldung aus der Quell- oder Zielregion, nachdem die Replikationskonfiguration erstellt wurde. Um die Replikation für das Dateisystem fortzusetzen, müssen Sie erneut die AWS-Region Option aktivieren. Weitere Informationen finden Sie AWS-Regionen im AWSAllgemeinen Referenzhandbuch unter [Verwaltung](#).
- Der **Error** Status tritt auf, wenn entweder das Quell- oder das Zielsystem (oder beide) ausgefallen sind und nicht wiederhergestellt werden können. Weitere Informationen finden Sie unter [Überwachung des Replikationsstatus](#) im Amazon EFS-Benutzerhandbuch. Sie müssen die Replikationskonfiguration löschen und dann die letzte Sicherung des

ausgefallenen Dateisystems (entweder die Quelle oder das Ziel) in einem neuen Dateisystem wiederherstellen.

Typ: Zeichenfolge

Zulässige Werte: ENABLED | ENABLING | DELETING | ERROR | PAUSED | PAUSING

Erforderlich: Ja

LastReplicatedTimestamp

Der Zeitpunkt, zu dem die letzte Synchronisierung im Zieldateisystem erfolgreich 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.

Typ: Zeitstempel

Erforderlich: Nein

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS-SDK für Ruby V3](#)

DestinationToCreate

Beschreibt das neue oder vorhandene Zielsystem für die Replikationskonfiguration.

Inhalt

AvailabilityZoneName

Um ein Dateisystem zu erstellen, das One Zone-Speicher verwendet, geben Sie den Namen der Availability Zone an, in der das Zielsystem erstellt werden soll.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 1. Höchstlänge = 64 Zeichen.

Pattern: . +

Erforderlich: Nein

FileSystemId

Die ID des Dateisystems, das für das Ziel verwendet werden soll. Die Replikationsüberschreibreplikation des Dateisystems muss deaktiviert sein. Wenn Sie keine ID angeben, erstellt EFS ein neues Dateisystem für das Replikationsziel.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge von 128.

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 Zielsystems 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 — Zum Beispiel die eindeutige Kennung des Schlüssels `1234abcd-12ab-34cd-56ef-1234567890ab`.

- ARN — Zum Beispiel der Amazon-Ressourcenname (ARN) für den Schlüsselarn: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 — Zum Beispiel der ARN für einen Schlüsselaliasarn:aws:kms:us-west-2:444455556666:alias/projectKey1.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge von 2 048.

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.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 1. Höchstlänge = 64 Zeichen.

Pattern: `^[a-z]{2}-((iso[a-z]{0,1}-)|(gov-)){0,1}[a-z]+-{0,1}[0-9]{0,1}$`

Erforderlich: Nein

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)

- [AWS-SDK für 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

Das AWS-Konto, das 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. Zum Beispiel ist `use1-az1` eine Availability Zone ID für die `us-east-1` AWS-Region, und sie hat den gleichen Standort in jedem AWS-Konto.

Typ: Zeichenfolge

Erforderlich: Nein

AvailabilityZoneName

Beschreibt die AWS-Availability-Zone, in der sich das Dateisystem befindet; sie 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 AWS KMS key zum Schutz des verschlüsselten Dateisystems.

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 einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS-SDK für Ruby V3](#)

FileSystemProtectionDescription

Beschreibt den Schutz eines Dateisystems.

Inhalt

ReplicationOverwriteProtection

Der Status des Replikationsüberschreibschutzes des Dateisystems.

- **ENABLED**— Das Dateisystem kann in einer Replikationskonfiguration nicht als Zieldateisystem verwendet werden. Das Dateisystem ist beschreibbar. Der Schutz vor dem Überschreiben der Replikation ist **ENABLED** standardmäßig aktiviert.
- **DISABLED**— Das Dateisystem kann in einer Replikationskonfiguration als Zieldateisystem verwendet werden. Das Dateisystem ist schreibgeschützt und kann nur durch EFS-Replikation geändert werden.
- **REPLICATING**— Das Dateisystem wird in einer Replikationskonfiguration als Zieldateisystem verwendet. Das Dateisystem ist schreibgeschützt und wird nur durch die EFS-Replikation geändert.

Wenn die Replikationskonfiguration gelöscht wird, wird der Replikationsschutz des Dateisystems erneut aktiviert, sodass das Dateisystem wieder beschreibbar ist.

Typ: Zeichenfolge

Zulässige Werte: **ENABLED** | **DISABLED** | **REPLICATING**

Erforderlich: Nein

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS-SDK für Ruby V3](#)

FileSystemSize

Die letzte bekannte gemessene Größe (in Byte) der im Dateisystem gespeicherten Daten in seinem Value Feld und der Zeitpunkt, zu dem diese Größe in seinem Timestamp Feld bestimmt wurde. Der Wert steht nicht für die Größe eines konsistenten Snapshots des Dateisystems, aber er ist letztlich konsistent, wenn keine Schreibvorgänge in das Dateisystem erfolgen. Das heißt, der Wert stellt nur dann die tatsächliche Größe dar, wenn das Dateisystem über einen Zeitraum von mehr als ein paar Stunden nicht verändert wurde. Andernfalls entspricht der Wert nicht unbedingt der exakten Größe, die das Dateisystem zu einem bestimmten Zeitpunkt hatte.

Inhalt

Value

Die letzte bekannte gemessene Größe (in Byte) der im Dateisystem gespeicherten Daten.

Type: Long

Gültiger Bereich: Mindestwert 0.

Erforderlich: Ja

Timestamp

Der Zeitpunkt, zu dem die Größe der im Value Feld zurückgegebenen Daten bestimmt wurde. Der Wert ist die ganzzahlige Anzahl von Sekunden seit 1970-01-01T 00:00:00 Z.

Typ: Zeitstempel

Erforderlich: Nein

ValueInArchive

Die letzte bekannte gemessene Größe (in Byte) der in der Archivspeicherklasse gespeicherten Daten.

Type: Long

Gültiger Bereich: Mindestwert 0.

Erforderlich: Nein

ValueInIA

Die letzte bekannte gemessene Größe (in Byte) von 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 Byte) von Daten, die in der Speicherklasse Standard gespeichert sind.

Type: Long

Gültiger Bereich: Mindestwert 0.

Erforderlich: Nein

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS-SDK für Ruby V3](#)

LifecyclePolicy

Beschreibt eine Richtlinie, die vom Lebenszyklusmanagement verwendet wird und angibt, wann Dateien in und aus Speicherklassen überführt 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 (der Standardspeicherkategorie), nach denen 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 einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS-SDK für Ruby V3](#)

MountTargetDescription

Stellt eine Beschreibung eines Mount-Ziels bereit.

Inhalt

FileSystemId

Die ID des Dateisystems, für das das Bereitstellungsziel bestimmt ist.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge von 128.

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

Lebenszyklus-Status des Bereitstellungsziels.

Typ: Zeichenfolge

Zulässige Werte: `creating | available | updating | deleting | deleted | error`

Erforderlich: Ja

MountTargetId

Vom System zugewiesene Mount-Ziel-ID.

Typ: Zeichenfolge

Minimale Höchstlänge 45

Pattern: `^fsmt-[0-9a-f]{8,40}$`

Erforderlich: Ja

SubnetId

Die ID des Subnetzes des Bereitstellungsziels.

Typ: Zeichenfolge

Minimale Höchstlänge 47

Pattern: `^subnet-[0-9a-f]{8,40}$`

Erforderlich: Ja

AvailabilityZoneId

Die eindeutige und konsistente Kennung der Availability Zone, in der sich das Mount-Ziel befindet. Beispielsweise `use1-az1` ist es eine AZ-ID für die Region „us-east-1“ und hat in jedem AWS-Konto

Typ: Zeichenfolge

Required: No

AvailabilityZoneName

Der Name der Availability Zone, in der sich das Bereitstellungsziel befindet. Availability Zones werden den einzelnen AWS-Konto Zonen unabhängig voneinander Namen zugeordnet. Beispielsweise ist die Availability Zone `us-east-1a` für Ihre AWS-Konto möglicherweise nicht derselbe Standort wie `us-east-1a` für eine andere AWS-Konto.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 1. Höchstlänge = 64 Zeichen.

Pattern: `.+`

Erforderlich: Nein

IpAddress

Adresse, an der das Dateisystem mithilfe des Mount-Targets gemountet werden kann.

Typ: Zeichenfolge

Minimvon 7. Höchstlänge 15

Pattern: `^[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}$`

Erforderlich: Nein

NetworkInterfaceId

Die ID der Netzwerkschnittstelle, die Amazon EFS bei der Erstellung des Bereitstellungsziels erstellt hat.

Typ: Zeichenfolge

Required: No

OwnerId

AWS-KontoID, die Eigentümer der Ressource ist.

Typ: Zeichenfolge

Höchstgen 14

Pattern: `^(\d{12})|(\d{4}-\d{4}-\d{4})$`

Erforderlich: Nein

VpcId

Die Virtual Private Cloud (VPC) ID, in der das Bereitstellungsziel konfiguriert ist.

Typ: Zeichenfolge

Erforderlich: Nein

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS SDK für Ruby V3](#)

PosixUser

Die vollständige POSIX-Identität, einschließlich der Benutzer-ID, der Gruppen-ID und aller sekundären Gruppen-IDs, auf dem Zugriffspunkt, der für alle Dateisystemoperationen verwendet wird, die von NFS-Clients mit dem Zugriffspunkt ausgeführt werden.

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 1 000.

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 1 000.

Erforderlich: Ja

SecondaryGids

Sekundäre POSIX-Gruppen-IDs, die für alle Dateisystemoperationen verwendet werden, die diesen Zugriffspunkt verwenden.

Typ: Array von 1 000.

Array-Mitglieder: Die Mindestanzahl beträgt 0 Elemente. Die maximale Anzahl beträgt 16 Elemente.

Gültiger Bereich: Mindestwert 0. Maximaler Wert von 1 000.

Required: No

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS SDK für Ruby V3](#)

ReplicationConfigurationDescription

Beschreibt die Replikationskonfiguration für ein bestimmtes Dateisystem.

Inhalt

CreationTime

Beschreibt, wann die Replikationskonfiguration erstellt wurde.

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-Quelldateisystems in der Replikationskonfiguration.

Typ: Zeichenfolge

Erforderlich: Ja

SourceFileSystemArn

Der Amazon-Ressourcenname (ARN) des aktuellen Quelldateisystems in der Replikationskonfiguration.

Typ: Zeichenfolge

Erforderlich: Ja

SourceFileSystemId

Die ID des Amazon EFS-Quelldateisystems, das repliziert wird.

Typ: Zeichenfolge

Längenbeschränkungen: Maximale Länge von 128.

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 von 1. Höchstlänge = 64 Zeichen.

Pattern: `^[a-z]{2}-((iso[a-z]{0,1}-)|(gov-)){0,1}[a-z]+-{0,1}[0-9]{0,1}$`

Erforderlich: Ja

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS-SDK für Ruby V3](#)

ResourceIdPreference

Beschreibt den Ressourcentyp und seine ID-Präferenz für den AWS-Konto Benutzer in der aktuellen VersionAWS-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-Voreinstellung gilt, FILE_SYSTEM undMOUNT_TARGET.

Typ: Zeichenfolge-Array

Zulässige Werte: FILE_SYSTEM | MOUNT_TARGET

Erforderlich: Nein

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS-SDK für 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 die POSIX-IDs und Berechtigungen an, die auf das RootDirectory des Zugriffspunkts angewendet werden sollen. 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 einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-SDK für C++](#)
- [AWS-SDK für Go](#)
- [AWS-SDK für Java V2](#)
- [AWS-SDK für Ruby V3](#)

Tag

Ein Tag ist ein Schlüsselwertpaar. Zulässige 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 von 1. 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 von 256.

Pattern: `^([\\p{L}\\p{Z}\\p{N}_.:/=+\\-@]*)$`

Erforderlich: Ja

Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS-SDKs finden Sie unter:

- [AWS-SDK für C++](#)
- [AWS-SDK for Go](#)
- [AWS-SDK für Java V2](#)
- [AWS SDK für Ruby V3](#)

Zusätzliche Informationen für Amazon EFS

Nachfolgend finden Sie zusätzliche Informationen über Amazon EFS, einschließlich Funktionen, die immer noch unterstützt werden, die aber nicht unbedingt empfohlen werden.

Themen

- [Sichern von Amazon-EFS-Dateisystemen mit AWS Data Pipeline](#)
- [Mounten von Dateisystemen ohne die EFS-Mountinghilfe](#)

Sichern von Amazon-EFS-Dateisystemen mit AWS Data Pipeline

Dieses Thema enthält Informationen zur Verwendung von AWS Data Pipeline, einer Legacy-Sicherungs- und Wiederherstellungslösung für EFS-Dateisysteme.

Note

AWS Backup ist die empfohlene Sicherungs- und Wiederherstellungslösung für EFS-Dateisysteme. Weitere Informationen finden Sie unter [Sichern Ihrer Amazon-EFS-Dateisysteme](#).

Mit erstellen AWS Data Pipeline Sie eine Datenpipeline mithilfe des - AWS Data Pipeline Service. Diese Pipeline kopiert Daten aus Ihrem Amazon-EFS-Dateisystem (dem Produktionsdateisystem) in ein anderes Amazon-EFS-Dateisystem (das Sicherungs-Dateisystem).

AWS Data Pipeline besteht aus Vorlagen, die Folgendes implementieren:

- Automatisierte Backups basierend auf einem von Ihnen definierten Zeitplan (z. B. stündlich, täglich, wöchentlich oder monatlich).
- Automatisierte Rotation der Sicherungen, wobei die älteste Sicherung auf der Grundlage der Anzahl der Sicherungen, die Sie behalten möchten, durch die neueste Sicherung ersetzt wird.
- Schnellere Sicherungen mit rsync, wobei nur Änderungen zwischen einer Sicherung und der nächsten gesichert werden.
- Effiziente Speicherung von Sicherungen mithilfe von harten Links. Ein harter Link ist ein Verzeichniseintrag, der einen Namen mit einer Datei in einem Dateisystem verbindet. Durch Festlegen eines harten Links können Sie eine vollständige Wiederherstellung von Daten aus

jeder Sicherung durchführen, wobei nur das gespeichert wird, was sich zwischen den einzelnen Sicherungen geändert hat.

Nachdem Sie die Sicherungslösung eingerichtet haben, zeigt Ihnen diese Anleitung, wie Sie auf die Sicherungen zugreifen können, um Ihre Daten wiederherzustellen. Diese Backup-Lösung hängt von der Ausführung von Skripts ab, die auf [gehostet werden GitHub](#), und unterliegt daher der GitHub Verfügbarkeit. Wenn Sie diese Abhängigkeit beseitigen und die Skripts stattdessen in einem Amazon-S3-Bucket hosten möchten, finden Sie weitere Informationen unter [Hosting der rsync-Skripts in einem Amazon-S3-Bucket](#).

Important

Diese Lösung erfordert die Verwendung von AWS Data Pipeline in derselben AWS-Region wie Ihr Dateisystem. Da AWS Data Pipeline in USA Ost (Ohio) nicht unterstützt wird, funktioniert diese Lösung in dieser AWS nicht. Wir empfehlen, dass Sie Ihr Dateisystem in einem der anderen AWS-Region unterstützten verwenden, wenn Sie Ihr Dateisystem mit dieser Lösung sichern möchten.

Themen

- [Leistung für Amazon-EFS-Backups mit AWS Data Pipeline](#)
- [Überlegungen zu Amazon-EFS-Sicherungen mithilfe von AWS Data Pipeline](#)
- [Annahmen für Amazon-EFS-Backup mit AWS Data Pipeline](#)
- [So sichern Sie ein Amazon-EFS-Dateisystem mit AWS Data Pipeline](#)
- [Weitere Sicherungsressourcen](#)

Leistung für Amazon-EFS-Backups mit AWS Data Pipeline

Bei der Durchführung von Datensicherungen und -wiederherstellungen unterliegt die Leistung Ihres Dateisystems [Amazon-EFS-Leistung](#), einschließlich der Basisleistung und der Leistung bei der Verarbeitung von Spitzendurchsätzen. Der von Ihrer Sicherungslösung verwendete Durchsatz wird dem Gesamtdurchsatz Ihres Dateisystems angerechnet. In der folgenden Tabelle werden einige Empfehlungen zu den für diese Lösung geeigneten Größen des Amazon-EFS-Dateisystems und der Amazon-EFS-Instance bereitgestellt, wobei von einem 15 Minuten langen Sicherungsfenster ausgegangen wird.

EFS-Größe (durchschnittliche Dateigröße 30 MB)	Volumen der täglichen Änderungen	Verbleibende Spitzenstunden	Minimale Anzahl der Sicherungsagenten
256 GB	Weniger als 25 GB	6.75	1 - m3.medium
512 GB	Weniger als 50 GB	7.75	1 - m3.large
1.0 TB	Weniger als 75 GB	11.75	2 - m3.large*
1.5 TB	Weniger als 125 GB	11.75	2 - m3.xlarge*
2.0 TB	Weniger als 175 GB	11.75	3 - m3.large*
3.0 TB	Weniger als 250 GB	11.75	4 - m3.xlarge*

* Diese Schätzungen basieren auf der Annahme, dass in einem EFS-Dateisystem gespeicherte Daten mit einer Größe von 1 TB oder mehr so organisiert sind, dass sie über mehrere Sicherungsknoten verteilt werden können. Die Beispielskripts mit mehreren Knoten verteilen die Sicherungslast auf Knoten auf der Grundlage der Inhalte des Verzeichnisses der ersten Ebene Ihres EFS-Dateisystems.

Wenn beispielsweise zwei Sicherungsknoten vorhanden sind, sichert ein Knoten alle Dateien und Verzeichnisse mit geraden Zahlen bei der Durchnummerierung im Verzeichnis der ersten Ebene. Der ungerade Knoten tut dies für die ungeraden Dateien und Verzeichnisse. In einem weiteren Beispiel mit sechs Verzeichnissen im Amazon-EFS-Dateisystem und vier Sicherungsknoten sichert der erste Knoten das erste und das fünfte Verzeichnis. Der zweite Knoten sichert das zweite und das sechste Verzeichnis, und der dritte und vierte Knoten sichern das dritte und vierte Verzeichnis.

Überlegungen zu Amazon-EFS-Sicherungen mithilfe von AWS Data Pipeline

Beachten Sie Folgendes, wenn Sie sich entscheiden, ob Sie eine Amazon-EFS-Sicherungslösung mithilfe von AWS Data Pipeline implementieren möchten:

- Dieser Ansatz für das EFS-Backup umfasst eine Reihe von AWS Ressourcen. Für diese Lösung müssen Sie Folgendes erstellen:
 - Ein Produktionsdateisystem und ein Sicherungsdateisystem, das eine vollständige Kopie des Produktionsdateisystems umfasst. Das System enthält auch inkrementelle Änderungen an Ihren Daten während des Sicherungsrotationszeitraums.
 - Amazon EC2-Instances, deren Lebenszyklen von verwaltet werden AWS Data Pipeline, die Wiederherstellungen und geplante Backups durchführen.
 - Eine Regel, die regelmäßig AWS Data Pipeline für die Sicherung von Daten geplant ist.
 - Ein AWS Data Pipeline zum Wiederherstellen von Backups.

Wenn diese Lösung implementiert ist, werden diese Services Ihrem Konto in Rechnung gestellt. Weitere Informationen finden Sie auf den Preisseiten für [Amazon EFS](#), [Amazon EC2](#) und [AWS Data Pipeline](#).

- Bei dieser Lösung handelt es sich nicht um eine Offline-Sicherungslösung. Um eine vollständig konsistente und vollständige Sicherung zu gewährleisten, unterbrechen Sie alle Dateischreibvorgänge im Dateisystem, oder machen Sie das Mounting des Dateisystems rückgängig, während die Sicherung stattfindet. Wir empfehlen, alle Sicherungen während geplanter Ausfallzeiten oder außerhalb der Geschäftszeiten durchzuführen.

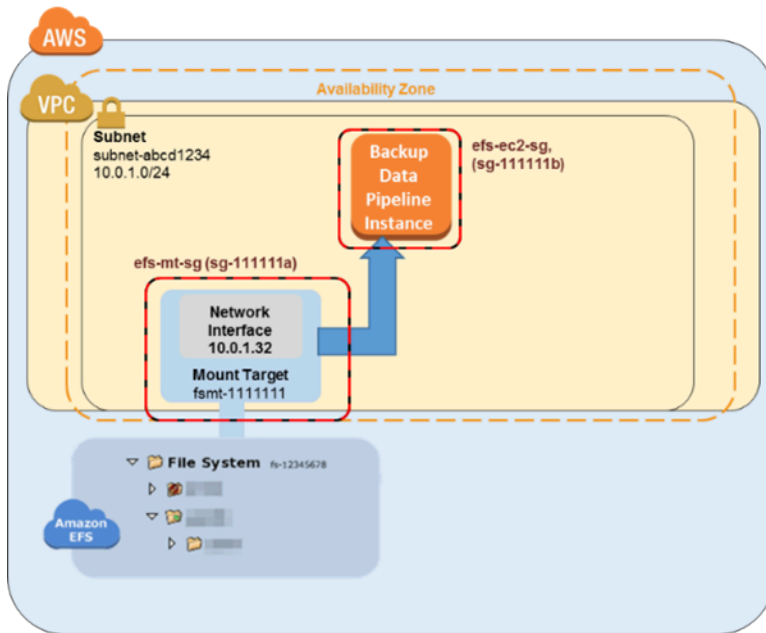
Annahmen für Amazon-EFS-Backup mit AWS Data Pipeline

Dieser exemplarischen Vorgehensweise liegen mehrere Annahmen und die folgenden Beispielwerte zugrunde:

- Bevor Sie beginnen, geht diese exemplarische Vorgehensweise davon aus, dass Sie [Erste Schritte](#) bereits abgeschlossen haben.
- Nachdem Sie die „Erste Schritte“-Übung abgeschlossen haben, haben Sie zwei Sicherheitsgruppen, ein VPC-Subnetz und ein Dateisystem-Mounting-Ziel für das Dateisystem, das Sie sichern möchten. Verwenden Sie für den Rest dieses Beispiels die folgenden Beispielwerte:
 - Die ID des Dateisystems, das Sie in diesem Beispiel sichern, ist fs-12345678.
 - Die Sicherheitsgruppe für das Dateisystem, das mit dem Mounting-Ziel verbunden ist, heißt efs-mt-sg (sg-1111111a).
 - Die Sicherheitsgruppe, die den Amazon-EC2-Instances die Möglichkeit gibt, eine Verbindung zum Produktions-EFS-Mountingpunkt herzustellen, heißt efs-ec2-sg (sg-1111111b).
 - Die VPC-Subnetz hat den ID-Wert subnet-abcd1234.

- Die Mounting-Ziel-IP-Adresse des Quelldateisystems für das Dateisystem, das Sie sichern möchten, ist 10.0.1.32: /.
- Das Beispiel geht davon aus, dass das Produktionsdateisystem ein Content Management-System ist, das Mediendateien mit einer durchschnittlichen Größe von 30 MB bereitstellt.

Die vorangehenden Annahmen und Beispiele sind in der folgenden Grafik für die Ersteinrichtung dargestellt.



So sichern Sie ein Amazon-EFS-Dateisystem mit AWS Data Pipeline

Befolgen Sie die Schritte in diesem Abschnitt, um Ihr Amazon-EFS-Dateisystem mit AWS Data Pipeline zu sichern oder wiederherzustellen.

Themen

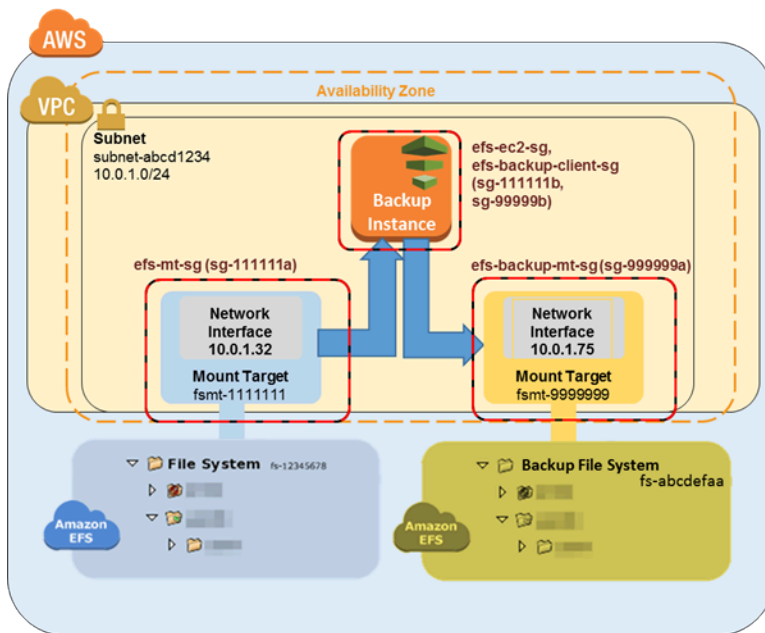
- [Schritt 1: Erstellen einer Sicherungskopie des Amazon-EFS-Dateisystems](#)
- [Schritt 2: Herunterladen der AWS Data Pipeline Vorlage für Backups](#)
- [Schritt 3: Erstellen einer Data Pipeline für die Sicherung](#)
- [Schritt 4: Zugreifen Sie Ihre Amazon-EFS-Sicherungen](#)

Schritt 1: Erstellen einer Sicherungskopie des Amazon-EFS-Dateisystems

Hier erstellen Sie separate Sicherheitsgruppen, Dateisysteme und Mountingpunkte, um Ihre Sicherungen von ihrer Datenquelle zu trennen. In diesem ersten Schritt erstellen Sie diese Ressourcen:

1. Erstellen Sie zuerst zwei neue Sicherheitsgruppen. Die Beispielsicherheitsgruppe für das Sicherungs-Mounting-Ziel ist `efs-backup-mt-sg` (`sg-9999999a`). Die Beispielsicherheitsgruppe für die EC2-Instance für den Zugriff auf das Mounting-Ziel ist `efs-backup-ec2-sg` (`sg-9999999b`). Denken Sie daran, dass Sie diese Sicherheitsgruppen in derselben VPC wie das EFS Volume erstellen, das Sie sichern möchten. In diesem Beispiel ist es die mit dem `subnet-abcd1234`-Subnetz verbundene VPC. Weitere Informationen zum Erstellen von Sicherheitsgruppen finden Sie unter [Erstellen von Sicherheitsgruppen](#).
2. Erstellen Sie dann ein Sicherungs-Amazon-EFS-Dateisystem. In diesem Beispiel ist die Dateisystem-ID `fs-abcdefaa`. Weitere Informationen zum Erstellen von Dateisystemen finden Sie unter [Erstellen eines Amazon-EFS-Dateisystems](#).
3. Erstellen Sie schließlich einen Mountingpunkt für das EFS-Sicherungsdateisystem, und nehmen Sie an, dass dies den Wert `10.0.1.75:/` hat. Weitere Informationen zum Erstellen von Mountingpunkten finden Sie unter [Erstellen und Verwalten von Mount-Zielen und Sicherheitsgruppen](#).

Nach Abschluss dieses ersten Schritts sollte Ihre Einrichtung wie im folgenden Beispieldiagramm aussehen.



Schritt 2: Herunterladen der AWS Data Pipeline Vorlage für Backups

AWS Data Pipeline hilft Ihnen, Daten zuverlässig zu verarbeiten und in bestimmten Intervallen zwischen verschiedenen AWS Datenverarbeitungs- und Speicherservices zu verschieben. Mithilfe der AWS Data Pipeline Konsole können Sie vorkonfigurierte Pipeline-Definitionen erstellen, die als Vorlagen bezeichnet werden. Sie können diese Vorlagen verwenden, um AWS Data Pipeline schnell mit zu beginnen. Bei dieser Anleitung wird eine Vorlage bereitgestellt, um den Vorgang der Einrichtung Ihrer Sicherungspipeline zu vereinfachen.

Nach der Implementierung erstellt diese Vorlage eine Data Pipeline, die eine einzelne Amazon-EC2-Instance nach dem Zeitplan startet, den Sie für die Sicherung von Daten vom Produktionsdateisystem zum Sicherungssystem eingerichtet haben. Diese Vorlage enthält eine Reihe von Platzhalterwerten. Sie geben die übereinstimmenden Werte für diese Platzhalter im Abschnitt Parameter der AWS Data Pipeline Konsole an. Laden Sie die AWS Data Pipeline Vorlage für Backups unter [1-Node-EFSBackupDataPipeline.json](#) von herunter GitHub.

Note

Diese Vorlage verweist auch auf ein Skript und führt dieses aus, um die Sicherungsbefehle durchzuführen. Sie können das Skript vor dem Erstellen der Pipeline herunterladen, um zu prüfen, was es genau tut. Um das Skript zu überprüfen, laden Sie [efs-backup.sh](#) von herunter GitHub. Diese Backup-Lösung hängt von der Ausführung von Skripten ab, die auf gehostet werden GitHub und der GitHub Verfügbarkeit unterliegen. Wenn Sie diese Abhängigkeit

beseitigen und die Skripts stattdessen in einem Amazon-S3-Bucket hosten möchten, finden Sie weitere Informationen unter [Hosting der rsync-Skripts in einem Amazon-S3-Bucket](#).

Schritt 3: Erstellen einer Data Pipeline für die Sicherung

Gehen Sie zum Erstellen ihrer Data Pipeline wie folgt vor.

So erstellen Sie eine Data Pipeline für Amazon-EFS-Sicherungen:

1. Öffnen Sie die - AWS Data Pipeline Konsole unter <https://console.aws.amazon.com/datapipeline/>.

 **Important**

Stellen Sie sicher, dass Sie in derselben AWS-Region wie Ihre Amazon-EFS-Dateisysteme arbeiten.

2. Wählen Sie Create new pipeline (Neue Pipeline erstellen) aus.
3. Geben Sie Werte für Name und optional für Description (Beschreibung) ein.
4. Wählen Sie für Source (Quelle) die Option Import a definition (Definition importieren) und anschließend Load local file (Lokale Datei laden) aus.
5. Navigieren Sie im Datei-Explorer zu der Vorlage, die Sie in [Schritt 2: Herunterladen der AWS Data Pipeline Vorlage für Backups](#) gespeichert haben, und wählen Sie dann Open (Öffnen) aus.
6. Geben Sie unter Parameters (Parameter) die Details für Ihr Sicherungs- und Ihr Produktions-EFS-Dateisystem an.

Parameters

Production EFS mount target IP address.	10.0.1.32:/
Security group that can connect to the Production EFS mount point.	sg-1111111b
Interval for backups.	daily
Security group that can connect to the Backup EFS mount point.	sg-9999999b
Number of backups to retain.	7
Backup EFS mount target IP address.	10.0.1.75:/
VPC subnet for your backup EC2 instance (ideally the same subnet used for the production EFS mount point).	subnet-1234abcd
Instance type for creating backups.	m3.medium
Name for the directory that will contain your backups.	backup-fs-12345678
Shell command to run.	wget https://raw.githubusercontent.com/aws-labs/data-pipeline-

7. Konfigurieren Sie die Optionen in Plan, um Ihren Amazon-EFS-Sicherungszeitplan zu konfigurieren. Die Sicherung in diesem Beispiel wird einmal täglich ausgeführt wird, und die Sicherungen werden eine Woche lang aufbewahrt. Wenn eine Sicherung sieben Tage alt ist, wird sie durch die nächstältere Sicherung ersetzt.

Schedule

Run ☐ once on pipeline activation ☒ on a schedule

Run every 1 day(s)

Starting ☒ on pipeline activation ☐ 2016-06-28 02:46 UTC (Current time is 02:48 UTC)

Ending ☒ never ☐ after 1 occurrence(s) 2016-06-29 02:46 UTC (Current time is 02:48 UTC)

Note

Wir empfehlen, eine Laufzeit anzugeben, die außerhalb Ihrer Spitzenzeiten liegt.

8. (Optional) Geben Sie einen Amazon-S3-Speicherort zum Speichern von Pipeline-Protokollen an, konfigurieren Sie eine benutzerdefinierte IAM-Rolle oder fügen Sie Tags zur Beschreibung Ihrer Pipeline hinzu.
9. Wählen Sie nach der Konfiguration Ihrer Pipeline Activate (Aktivieren) aus.

Damit haben Sie nun Ihre Amazon-EFS-Sicherungs-Data-Pipeline konfiguriert und aktiviert. Weitere Informationen zu AWS Data Pipeline finden Sie im [AWS Data Pipeline -Entwicklerhandbuch](#). Jetzt können Sie die Sicherung als Test durchführen, oder Sie können warten, bis die Sicherung zum geplanten Zeitpunkt durchgeführt wird.

Schritt 4: Zugreifen Sie Ihre Amazon-EFS-Sicherungen

Ihre Amazon-EFS-Sicherung wurde jetzt erstellt und aktiviert und läuft gemäß dem von Ihnen definierten Zeitplan. Dieser Schritt erläutert, wie Sie auf Ihre EFS-Sicherungen zugreifen können. Ihre Sicherungen werden in dem EFS-Sicherungsdateisystem gespeichert, das Sie im folgenden Format erstellt haben.

```
backup-efs-mount-target:/efs-backup-id/[backup interval].[0-backup retention]-->
```

Unter Verwendung der Werte aus dem Beispielszenario befindet sich die Sicherung des Dateisystems unter `10.1.0.75:/fs-12345678/daily.[0-6]`, wobei `daily.0` die jüngste Sicherung und `daily.6` die älteste von sieben rotierenden Sicherungen ist.

Der Zugriff auf Ihre Sicherungen ermöglicht Ihnen die Wiederherstellung von Daten zu Ihrem Produktionsdateisystem. Sie können wählen, ob ein gesamtes Dateisystem oder einzelne Dateien wiederhergestellt werden sollen.

Schritt 4.1: Wiederherstellen einer gesamten Amazon-EFS-Sicherung

Für das Wiederherstellen einer Sicherungskopie eines Amazon-EFS-Dateisystems ist ein weiteres erforderlich AWS Data Pipeline, ähnlich dem, das Sie in konfiguriert haben [Schritt 3: Erstellen einer Data Pipeline für die Sicherung](#). Diese Wiederherstellungspipeline funktioniert jedoch umgekehrt wie die Sicherungspipeline. Diese Wiederherstellungen sind normalerweise nicht zum automatischen Start geplant.

Wie Sicherungen auch können Wiederherstellungen parallel ausgeführt werden, um Ihre Recovery Time Objective (RTO) zu erfüllen. Denken Sie beim Erstellen einer Data Pipeline daran, dass Sie ihre Ausführung planen müssen. Wenn Sie die Ausführung bei Aktivierung wählen, starten Sie den Wiederherstellungsvorgang sofort. Wir empfehlen, dass Sie nur dann eine Wiederherstellungspipeline erstellen, wenn Sie eine Wiederherstellung durchführen müssen, oder wenn Sie an ein bestimmtes Zeitfenster denken.

Die Spitzenkapazität wird von dem Sicherungs-EFS und dem Wiederherstellungs-EFS genutzt. Weitere Informationen zur Leistung finden Sie unter [Amazon-EFS-Leistung](#). Die folgende Vorgehensweise zeigt, wie Sie Ihre Wiederherstellungspipeline erstellen und implementieren können.

So erstellen Sie eine Data Pipeline für die Wiederherstellung von EFS-Daten.

1. Laden Sie die Data Pipeline-Vorlage für die Datenwiederherstellung von Ihrem Sicherungs-EFS-Dateisystem herunter. Diese Vorlage startet eine einzelne Amazon-EC2-Instance, basierend auf der angegebenen Größe. Der Start erfolgt nur, wenn Sie dies angegeben haben. Laden Sie die AWS Data Pipeline Vorlage für Backups unter [1-Node-EFSRestoreDataPipeline.json](#) von herunter GitHub.

 Note

Diese Vorlage verweist auch auf ein Skript und führt dieses aus, um die Wiederherstellungsbefehle durchzuführen. Sie können das Skript vor dem Erstellen der Pipeline herunterladen, um zu prüfen, was es genau tut. Um das Skript zu überprüfen, laden Sie [efs-restore.sh](#) von herunter GitHub.

2. Öffnen Sie die - AWS Data Pipeline Konsole unter <https://console.aws.amazon.com/datapipeline/>.

 Important

Stellen Sie sicher, dass Sie in derselben AWS-Region wie Ihre Amazon-EFS-Dateisysteme und Amazon EC2 arbeiten.

3. Wählen Sie Create new pipeline (Neue Pipeline erstellen) aus.
4. Geben Sie Werte für Name und optional für Description (Beschreibung) ein.
5. Wählen Sie für Source (Quelle) die Option Import a definition (Definition importieren) und anschließend Load local file (Lokale Datei laden) aus.
6. Navigieren Sie im Datei-Explorer zu der Vorlage, die Sie in [Schritt 1: Erstellen einer Sicherungskopie des Amazon-EFS-Dateisystems](#) gespeichert haben, und wählen Sie dann Open (Öffnen) aus.
7. Geben Sie unter Parameters (Parameter) die Details für Ihr Sicherungs- und Ihr Produktions-EFS-Dateisystem an.

Parameters	
Production EFS mount target IP address.	<input type="text" value="10.0.1.32/"/>
Security group that can connect to the Production EFS mount point.	<input type="text" value="sg-1111111b"/>
Instance type for performing the restore.	<input type="text" value="m3.large"/>
Security group that can connect to the Backup EFS mount point.	<input type="text" value="sg-9999999b"/>
Name for the directory that already contains your backups.	<input type="text" value="backup-fs-12345678"/>
Backup number to restore (0 = the most recent backup).	<input type="text" value="0"/>
Backup EFS mount target IP address.	<input type="text" value="10.0.1.75/"/>
Interval that you chose for the backup your going to restore.	<input type="text" value="daily"/>
VPC subnet for your restoration EC2 instance (ideally the same subnet used for the backup EFS mount point).	<input type="text" value="subnet-1234abcd"/>

8. Da Sie normalerweise Wiederherstellungen nur durchführen, wenn Sie sie benötigen, können Sie für die Ausführung der Wiederherstellung die Option Run once on pipeline activation (Einmal bei Aktivierung der Pipeline ausführen) festlegen. Oder Sie planen eine einmalige Wiederherstellung zu einem zukünftigen Zeitpunkt Ihrer Wahl, etwa während eines Zeitfensters außerhalb Ihrer Spitzenzeiten.
9. (Optional) Geben Sie einen Amazon-S3-Speicherort zum Speichern von Pipeline-Protokollen an, konfigurieren Sie eine benutzerdefinierte IAM-Rolle oder fügen Sie Tags zur Beschreibung Ihrer Pipeline hinzu.
10. Wählen Sie nach der Konfiguration Ihrer Pipeline Activate (Aktivieren) aus.

Damit haben Sie nun Ihre Amazon-EFS-Wiederherstellungs-Data-Pipeline konfiguriert und aktiviert. Wenn Sie nun ein Backup in Ihrem EFS-EFS-Dateisystem wiederherstellen müssen, aktivieren Sie es einfach über die AWS Data Pipeline Konsole. Weitere Informationen finden Sie im [AWS Data Pipeline -Entwicklerhandbuch](#).

Schritt 4.2: Wiederherstellen einzelner Dateien aus Ihren Amazon-EFS-Sicherungen

Sie können Dateien aus Ihren Amazon-EFS-Dateisystemsicherungen wiederherstellen, indem Sie eine Amazon-EC2-Instance starten, um das Produktions- und das Sicherungs-EFS-Dateisystem vorübergehend zu mounten. Die EC2-Instance muss Mitglied beider EFS-Client-Sicherheitsgruppen sein (in diesem Beispiel efs-ec2-sg und efs-backup-clients-sg). Beide EFS-Mounting-Ziele können von dieser Wiederherstellungs-Instance gemountet werden. So kann beispielsweise eine Wiederherstellungs-EC2-Instance die folgenden Mountingpunkte erstellen. Hier wird die `-o ro-`

Option für das Mounten des Sicherungs-EFS im schreibgeschützten Zustand verwendet, um zu verhindern, dass die Sicherung versehentlich geändert wird, wenn die Wiederherstellung aus der Sicherung versucht wird.

```
mount -t nfs source-efs-mount-target:/ /mnt/data
```

```
mount -t nfs -o ro backup-efs-mount-target:/fs-12345678/daily.0 /mnt/backup>
```

Nachdem Sie das Mounting der Ziele durchgeführt haben, können Sie Dateien aus der /mnt/backup zum gewünschten Speicherort in /mnt/data im Terminal mit dem Befehl cp -p kopieren. Beispielsweise kann ein gesamtes Startverzeichnis (mit der entsprechenden Dateisystemberechtigung) rekursiv mithilfe des folgenden Befehls kopiert werden.

```
sudo cp -rp /mnt/backup/users/my_home /mnt/data/users/my_home
```

Sie können eine einzelne Datei wiederherstellen, indem Sie den folgenden Befehl ausführen.

```
sudo cp -p /mnt/backup/user/my_home/.profile /mnt/data/users/my_home/.profile
```

Warning

Wenn Sie einzelne Dateien manuell wiederherstellen, achten Sie darauf, dass Sie nicht versehentlich die Sicherung selbst modifizieren. Andernfalls könnte diese dadurch beschädigt werden.

Weitere Sicherungsressourcen

Die in dieser Anleitung vorgestellte Backup-Lösung verwendet -Vorlagen für AWS Data Pipeline. Die in [Schritt 2: Herunterladen der AWS Data Pipeline Vorlage für Backups](#) und [Schritt 4.1: Wiederherstellen einer gesamten Amazon-EFS-Sicherung](#) verwendeten Vorlagen verwenden beide eine einzelne Amazon-EC2-Instance zur Durchführung Ihrer Funktion. Es gibt jedoch keine wirkliche Begrenzung für die Anzahl der parallelen Instances, die Sie ausführen können, um Ihre Daten in Amazon-EFS-Dateisystemen zu sichern oder wiederherzustellen. In diesem Thema finden Sie Links zu anderen AWS Data Pipeline Vorlagen, die für mehrere EC2-Instances konfiguriert sind, die Sie herunterladen und für Ihre Sicherungslösung verwenden können. Außerdem erhalten Sie Anleitungen dazu, wie Sie die Vorlagen so ändern können, dass sie weitere Instances enthalten.

Themen

- [Verwendung zusätzlicher Vorlagen](#)
- [Hinzufügen weiterer Sicherungs-Instances](#)
- [Hinzufügen weiterer Wiederherstellungs-Instances](#)
- [Hosting der rsync-Skripts in einem Amazon-S3-Bucket](#)

Verwendung zusätzlicher Vorlagen

Sie können die folgenden zusätzlichen Vorlagen von herunterladen GitHub:

- [2-Node-EFSBackupPipeline.json](#) – Diese Vorlage startet zwei parallele Amazon EC2-Instances, um Ihr Amazon-EFS-Dateisystem in der Produktion zu sichern.
- [2-Node-EFSRestorePipeline.json](#) – Diese Vorlage startet zwei parallele Amazon EC2-Instances, um ein Backup Ihres Amazon-EFS-Produktionsdateisystems wiederherzustellen.

Hinzufügen weiterer Sicherungs-Instances

Sie können den in dieser Anleitung verwendeten Sicherungsvorlagen weitere Knoten hinzufügen. Um einen Knoten hinzuzufügen, modifizieren Sie die folgenden Abschnitte der Vorlage `2-Node-EFSBackupDataPipeline.json`.

Important

Wenn Sie zusätzliche Knoten verwenden, können Sie keine Leerzeichen in Dateinamen und Verzeichnissen im Top-Level-Verzeichnis verwenden. Wenn Sie dies tun, werden diese Dateien und Verzeichnisse nicht gesichert bzw. wiederhergestellt. Alle Dateien und Unterverzeichnisse, die sich mindestens eine Ebene unter der obersten Ebene befinden, werden wie erwartet gesichert und wiederhergestellt.

- Erstellen Sie eine zusätzliche `EC2Resource` für jeden zusätzlichen Knoten, den Sie erstellen möchten (in diesem Beispiel ist das eine vierte EC2-Instance).

```
{  
  "id": "EC2Resource4",  
  "terminateAfter": "70 Minutes",
```

```

"instanceType": "#{myInstanceType}",
"name": "EC2Resource4",
"type": "Ec2Resource",
"securityGroupIds" : [ "#{mySrcSecGroupID}", "#{myBackupSecGroupID}" ],
"subnetId": "#{mySubnetID}",
"associatePublicIpAddress": "true"
},

```

- Erstellen Sie eine zusätzliche Data Pipeline-Aktivität für jeden weiteren Knoten (in diesem Fall Aktivität BackupPart4); achten Sie darauf, dass Sie die folgenden Abschnitte konfigurieren:
 - Aktualisieren Sie die runsOn-Referenz so, dass Sie auf die vorher erstellte EC2Resource verweist (EC2Resource4 im folgenden Beispiel).
 - Erhöhen Sie die letzten beiden scriptArgument-Werte so, dass sie dem Sicherungsteil, für den jeder Knoten verantwortlich ist, sowie der Gesamtzahl der Knoten entsprechen. Für "2" und "3" im nachfolgenden Beispiel ist der Sicherungsteil "3" für den vierten Knoten, da in diesem Beispiel unsere Moduluslogik die Zählung mit 0 beginnen muss.

```

{
  "id": "BackupPart4",
  "name": "BackupPart4",
  "runsOn": {
    "ref": "EC2Resource4"
  },
  "command": "wget https://raw.githubusercontent.com/aws-labs/data-pipeline-samples/master/samples/EFSBackup/efs-backup-rsync.sh\nchmod a+x efs-backup-rsync.sh\n./efs-backup-rsync.sh $1 $2 $3 $4 $5 $6 $7",
  "scriptArgument": ["#{myEfsSource}", "#{myEfsBackup}", "#{myInterval}", "#{myRetainedBackups}", "#{myEfsID}", "3", "4"],
  "type": "ShellCommandActivity",
  "dependsOn": {
    "ref": "InitBackup"
  },
  "stage": "true"
},

```

- Erhöhen Sie den letzten Wert in allen vorhandenen scriptArgument-Werten auf die Anzahl der Knoten (in diesem Beispiel "4").

```

{
  "id": "BackupPart1",
  ...

```

```

"scriptArgument": ["#{myEfsSource}", "#{myEfsBackup}", "#{myInterval}",
  "#{myRetainedBackups}", "#{myEfsID}", "1", "4"],
...
},
{
  "id": "BackupPart2",
  ...
  "scriptArgument": ["#{myEfsSource}", "#{myEfsBackup}", "#{myInterval}",
    "#{myRetainedBackups}", "#{myEfsID}", "2", "4"],
  ...
},
{
  "id": "BackupPart3",
  ...
  "scriptArgument": ["#{myEfsSource}", "#{myEfsBackup}", "#{myInterval}",
    "#{myRetainedBackups}", "#{myEfsID}", "0", "4"],
  ...
},

```

- Aktualisieren Sie die `FinalizeBackup`-Aktivität, und fügen Sie die neue Sicherungsaktivität der Liste `dependsOn` hinzu (BackupPart4 in diesem Fall).

```

{
  "id": "FinalizeBackup", "name": "FinalizeBackup", "runsOn": { "ref":
    "EC2Resource1" }, "command": "wget
    https://raw.githubusercontent.com/awslabs/data-pipeline-samples/master/samples/
    EFSBackup/efs-backup-end.sh\nchmod a+x
    efs-backup-end.sh\n./efs-backup-end.sh $1 $2", "scriptArgument": ["#{myInterval}",
    "#{myEfsID}"], "type": "ShellCommandActivity", "dependsOn": [ { "ref":
      "BackupPart1" },
    { "ref": "BackupPart2" }, { "ref": "BackupPart3" }, { "ref": "BackupPart4" } ],
    "stage":
    "true"

```

Hinzufügen weiterer Wiederherstellungs-Instances

Sie können den in dieser Anleitung verwendeten Wiederherstellungsvorlagen weitere Knoten hinzufügen. Um einen Knoten hinzuzufügen, modifizieren Sie die folgenden Abschnitte der Vorlage `2-Node-EFSRestorePipeline.json`.

- Erstellen Sie eine zusätzliche EC2Resource für jeden zusätzlichen Knoten, den Sie erstellen möchten (in diesem Fall eine dritte EC2-Instance mit dem Namen EC2Resource3).

```
{
  "id": "EC2Resource3",
  "terminateAfter": "70 Minutes",
  "instanceType": "#{myInstanceType}",
  "name": "EC2Resource3",
  "type": "Ec2Resource",
  "securityGroupIds" : [ "#{mySrcSecGroupID}", "#{myBackupSecGroupID}" ],
  "subnetId": "#{mySubnetID}",
  "associatePublicIpAddress": "true"
},
```

- Legen Sie für jeden weiteren Knoten eine zusätzliche Datenpipeline-Aktivität an (in diesem Fall Aktivität RestorePart3). Achten Sie darauf, die folgenden Abschnitte zu konfigurieren:
 - Aktualisieren Sie die runsOn-Referenz so, dass sie auf die zuvor erstellte EC2Resource verweist (in diesem Beispiel EC2Resource3).
 - Erhöhen Sie die letzten beiden scriptArgument-Werte so, dass sie dem Sicherungsteil, für den jeder Knoten verantwortlich ist, sowie der Gesamtzahl der Knoten entsprechen. Für "2" und "3" im nachfolgenden Beispiel ist der Sicherungsteil "3" für den vierten Knoten, da in diesem Beispiel unsere Moduluslogik die Zählung mit 0 beginnen muss.

```
{
  "id": "RestorePart3",
  "name": "RestorePart3",
  "runsOn": {
    "ref": "EC2Resource3"
  },
  "command": "wget https://raw.githubusercontent.com/aws-labs/data-pipeline-samples/master/samples/EFSBackup/efs-restore-rsync.sh\nchmod a+x efs-restore-rsync.sh\n./efs-backup-rsync.sh $1 $2 $3 $4 $5 $6 $7",
  "scriptArgument": ["#{myEfsSource}", "#{myEfsBackup}", "#{myInterval}", "#{myBackup}", "#{myEfsID}", "2", "3"],
  "type": "ShellCommandActivity",
  "dependsOn": {
    "ref": "InitBackup"
  },
  "stage": "true"
},
```

- Erhöhen Sie den letzten Wert in allen vorhandenen `scriptArgument`-Werten auf die Anzahl der Knoten (in diesem Beispiel "3").

```
{
  "id": "RestorePart1",
  ...
  "scriptArgument": ["#{myEfsSource}", "#{myEfsBackup}", "#{myInterval}",
    "#{myBackup}", "#{myEfsID}", "1", "3"],
  ...
},
{
  "id": "RestorePart2",
  ...
  "scriptArgument": ["#{myEfsSource}", "#{myEfsBackup}", "#{myInterval}",
    "#{myBackup}", "#{myEfsID}", "0", "3"],
  ...
},
```

Hosting der rsync-Skripts in einem Amazon-S3-Bucket

Diese Backup-Lösung hängt von der Ausführung von rsync-Skripten ab, die in einem GitHub Repository im Internet gehostet werden. Daher unterliegt diese Backup-Lösung dem GitHub verfügbaren Repository. Diese Anforderung bedeutet, dass, wenn das GitHub Repository diese Skripts entfernt oder wenn die GitHub Website offline geht, die oben implementierte Backup-Lösung nicht funktioniert.

Wenn Sie diese GitHub Abhängigkeit beseitigen möchten, können Sie stattdessen die Skripts in einem Amazon S3-Bucket hosten, den Sie besitzen. Nachfolgend werden die erforderlichen Schritte zum Hosten der Skripts erläutert.

So hosten Sie die rsync-Skripts in Ihrem eigenen Amazon-S3-Bucket:

1. Registrieren für AWS und Erstellen eines Administratorbenutzers – Wenn Sie bereits über ein verfügbares AWS-Konto verfügen, fahren Sie mit dem nächsten Schritt fort. Andernfalls lesen Sie unter [Einrichten](#) weiter.
2. Erstellen Sie einen Amazon-S3-Bucket – Wenn Sie bereits einen Bucket zum Hosting der rsync-Skripts haben, fahren Sie mit dem nächsten Schritt fort. Ansonsten finden Sie weitere Anleitungen unter [Erstellen eines Buckets](#) im Benutzerhandbuch für Amazon Simple Storage Service.

3. Herunterladen der rsync-Skripts und -Vorlagen – Laden Sie alle rsync-Skripts und -Vorlagen im [Ordner EFSBackup](#) von herunter GitHub. Notieren Sie sich den Speicherort auf Ihrem Computer, zu dem Sie diese Dateien herunterladen.
4. Laden Sie das rsync-Skript in Ihren S3-Bucket [Für eine Anleitung zum Laden von Objekten in Ihren S3-Bucket siehe](#) Hinzufügen eines Objekts zu einem Bucket im Benutzerhandbuch zu Amazon Simple Storage Service.
5. Ändern Sie die Berechtigungen für die hochgeladenen rsync-Skripts, so, dass Everyone (Jeder) die Erlaubnis zum Open/Download (Öffnen/Herunterladen) hat. Für eine Anleitung zum Ändern der Berechtigungen für ein Objekt in Ihrem S3-Bucket siehe [Bearbeiten von Objektberechtigungen](#) im Benutzerhandbuch zu Amazon Simple Storage Service.

The screenshot shows the 'Object: efs-backup.sh' details in the AWS S3 console. The 'Permissions' section is expanded, showing a table of grants. The first grant is for 'Everyone' with 'Open/Download' checked. The second grant is for a user (name redacted) with 'Open/Download', 'View Permissions', and 'Edit Permissions' all checked. There is an 'Add more permissions' button at the bottom left and 'Save' and 'Cancel' buttons at the bottom right.

Grantee	Open/Download	View Permissions	Edit Permissions
Everyone	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
[Redacted]	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

6. Aktualisieren Sie Ihre Vorlagen – Modifizieren Sie die wget-Anweisung im shellCmd-Parameter so, dass Sie auf den Amazon-S3-Bucket verweist, in den Sie das Startup-Skript gelegt haben. Speichern Sie die aktualisierte Vorlage, und verwenden Sie diese Vorlage, wenn Sie die Anleitung unter [Schritt 3: Erstellen einer Data Pipeline für die Sicherung](#) befolgen.

 Note

Wir empfehlen Ihnen, den Zugriff auf Ihren Amazon S3-Bucket auf das IAM-Konto zu beschränken, das die AWS Data Pipeline für diese Sicherungslösung aktiviert. Weitere Informationen finden Sie unter [Ändern von Bucket-Berechtigungen](#) im Benutzerhandbuch zu Amazon Simple Storage Service.

Sie hosten jetzt die rsync-Skripte für diese Backup-Lösung und Ihre Backups hängen nicht mehr von der GitHub Verfügbarkeit ab.

Mounten von Dateisystemen ohne die EFS-Mountinghilfe

 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. Dazu empfehlen wir die Verwendung des `- amazon-efs-utils` Pakets. Weitere Informationen finden Sie unter [Verwenden der amazon-efs-utils Tools](#).

Nachfolgend erfahren Sie, wie Sie den Network File System (NFS)-Client installieren und Ihr Amazon-EFS-Dateisystem auf einer Amazon-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

Bevor Sie ein Dateisystem mounten können, müssen Sie Ihre zugehörigen AWS -Ressourcen erstellen, konfigurieren und starten. Detaillierte Anweisungen finden Sie unter [Erste Schritte mit Amazon Elastic File System](#).

 Note

Bevor Sie Ihr Dateisystem mounten, müssen Sie VPC-Sicherheitsgruppen für Ihre Amazon-EC2-Instances erstellen und Ziele mit dem erforderlichen eingehenden und ausgehenden Zugriff mounten. Weitere Informationen finden Sie unter [Verwendung von VPC-Sicherheitsgruppen für Amazon EC2-Instance und Mounting-Ziele](#).

Themen

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

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 z. B. CentOS), Version 7 und höher
- 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.

 Note

RHEL 6.9 könnte für bestimmte Workloads suboptimal sein aufgrund von [Schlechte Leistung beim Öffnen vieler Dateien gleichzeitig](#).

 Note

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

Fehlerbehebung für AMI- und Kernel-Versionen

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](#).

Installieren des NFS-Clients

Zum Mounten Ihres Amazon-EFS-Dateisystems auf Ihrer Amazon-EC2-Instance müssen Sie zuerst 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. Beachten Sie Folgendes im Zusammenhang mit der Herstellung einer Verbindung mit der Instance:
 - Geben Sie für die Herstellung einer Verbindung mit Ihrer Instance von einem Computer mit macOS oder Linux aus die PEM-Datei für Ihren Secure Shell (SSH)-Client mit der `-i`-Option und dem Pfad zu Ihrem privaten Schlüssel an.
 - Um von einem Computer, auf dem Windows ausgeführt wird, eine Verbindung zu Ihrer Instance herzustellen, können Sie entweder MindTerm oder PuTTY verwenden. Wenn Sie PuTTY verwenden möchten, müssen Sie dies installieren und die `.pem`-Datei auf die folgende Weise zu einer `.ppk`-Datei konvertieren.

Weitere Informationen finden Sie in den folgenden Themen im Amazon EC2-Benutzerhandbuch für Linux-Instances:

- [Herstellung einer Verbindung zu Ihrer Linux-Instance von Windows mit PuTTY](#)
- [Herstellen einer Verbindung mit Ihrer Linux-Instance per SSH](#)

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 Schlüsselpaars](#).

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 den folgenden Befehlen. Für RHEL 7:

```
$ sudo service nfs start
```

Für RHEL 8:

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

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

```
$ sudo service nfs 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 NFSv4.1-Client-Kernel-Modul und die richtige NFS4-Userspace-Mountinghilfe einschließen.

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 Mounting des Amazon EFS-Dateisystems](#)

Empfohlene NFS-Mounting-Optionen

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).

- `wsz=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-Mounting-Ziels 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 Informationen und ausführliche Erklärungen der Standardwerte finden Sie auf den Seiten [man fstab](#) und [man nfs](#) 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 von der EC2-Sicherheitsgruppe aus ermöglicht. Weitere Informationen finden Sie unter [Verwendung von VPC-Sicherheitsgruppen für Amazon EC2-Instance und Mounting-Ziele](#).

- 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 Mounting-Ziels 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,rsiz=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,rsiz=65536,wsiz=65536,hard,timeo=600,retrans=2,noresvport,mountport=2049 fil  
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 Mounting-Ziels ist wie folgt.

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

Note

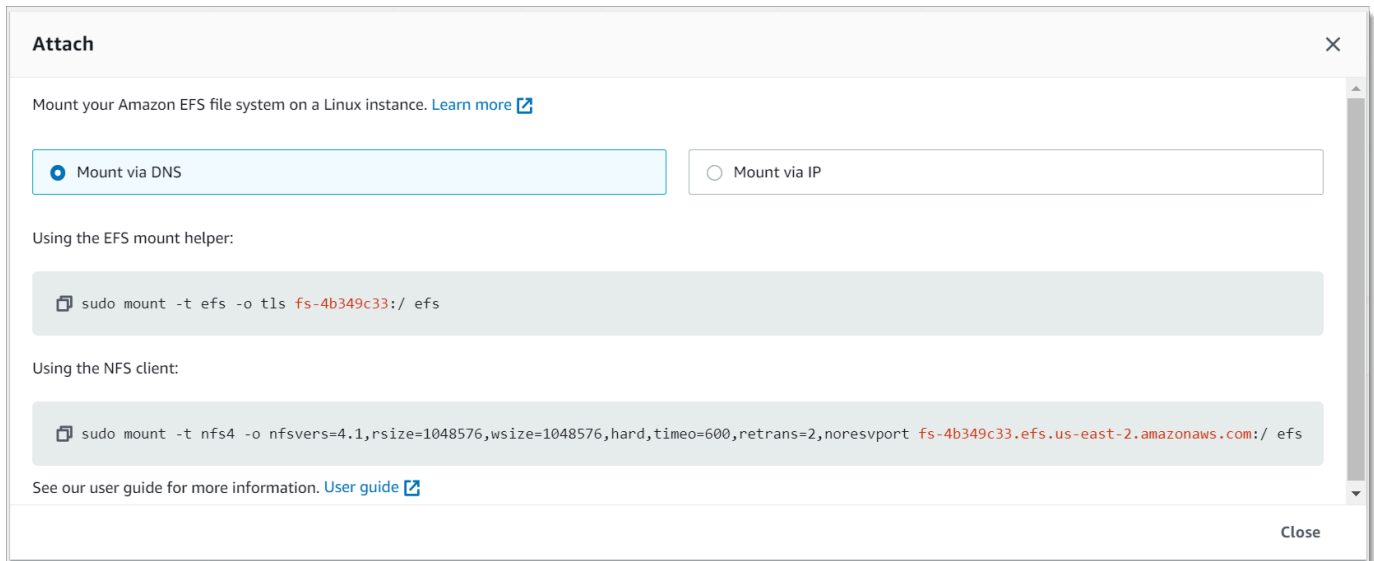
Die Mounten der Ziel-DNS-Namensauflösung in allen Availability Zones wird unterstützt.

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

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](#) im 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-Optionsgruppen](#) 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 [Anzeige von DNS-Hostnamen für die EC2-Instance](#) 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 [Exemplarische Vorgehensweise: Erstellen und Bereitstellen eines lokalen Dateisystems mit VPN AWS Direct Connect](#) und [Exemplarische Vorgehensweise: Mounten eines Dateisystems aus einer anderen VPC](#).

Note

Wir empfehlen, dass Sie nach dem Erstellen eines Mounting-Ziels 90 Sekunden warten, bevor Sie das Dateisystem mounten. Durch diese Wartezeit können die DNS-Datensätze vollständig in der weitergegeben werden AWS-Region , in der 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 Mounting-Ziels mounten. Das Mounten nach IP-Adresse funktioniert in Umgebungen, in denen DNS deaktiviert ist, wie etwa in VPCs mit deaktivierten DNS-Hostnamen.

Sie können das Mounten eines Dateisystems auch mithilfe der IP-Adresse des Mounting-Ziels 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 Mounting-Ziels sollten EC2-Instances das Mounting unter Verwendung der IP-Adresse des Mounting-Ziels in derselben Availability Zone wie die verbindende Instance durchführen.

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

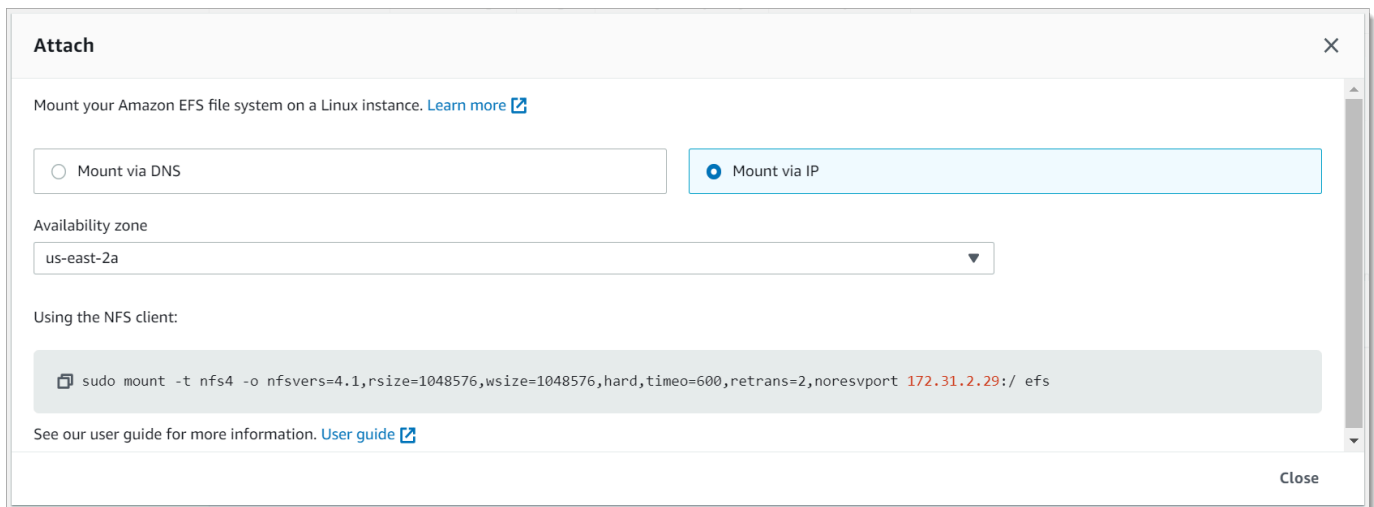
Note

Bevor Sie Ihr Dateisystem mounten, müssen Sie der Mount-Zielsicherheitsgruppe eine Regel hinzufügen, die eingehenden NFS-Zugriff von der EC2-Sicherheitsgruppe aus ermöglicht.

Weitere Informationen finden Sie unter [Verwendung von VPC-Sicherheitsgruppen für Amazon EC2-Instance und Mounting-Ziele](#).

Gehen Sie folgendes 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.



Attach

Mount your Amazon EFS file system on a Linux instance. [Learn more](#)

☐ Mount via DNS ☒ Mount via IP

Availability zone
us-east-2a

Using the NFS client:

```
sudo mount -t nfs4 -o nfsvers=4.1,rsize=1048576,wsz=1048576,hard,timeo=600,retrans=2,noresvport 172.31.2.29:/ efs
```

See our user guide for more information. [User guide](#)

Close

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,rsize=1048576,wsz=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  
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.

Mounting mit einer IP-Adresse in AWS CloudFormation

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

Dokumentverlauf

- API-Version: 2015-02-01
- Letzte Aktualisierung der Dokumentation: 13. März 2024

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
Das Limit für den elastischen Durchsatz wurde erhöht	Das Limit für den elastischen Durchsatz wurde für bestimmte erhöht AWS-Regionen. Weitere Informationen finden Sie unter Gesamter elastischer Standarddurchsatz für alle verbundenen Clients in jeder AWS-Region .	13. März 2024
Mehr IOPS	Dateisysteme, die den Elastic-Durchsatz verwenden, können maximal 90.000 Lesevorgänge für Daten ausführen, auf die selten zugegriffen wird. Weitere Informationen zur Leistung finden Sie unter Leistungsübersicht .	22. Januar 2024
Vorhandene AWS verwaltete Richtlinie aktualisiert	Der vorhandenen AmazonElasticFileSystemFullAccess Richtlinie wurde eine Berechtigung <code>elasticfilesystem:UpdateFileSystemProtection</code> hinzugefügt, die es Prinzipalen ermöglicht, den	8. November 2023

Schutz auf einem Dateisystem zu aktualisieren. Weitere Informationen finden Sie unter [Amazon-EFS-Updates für - AWS verwaltete Richtlinien](#).

[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 [Zielfilesystem](#).

8. 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 elastischem Durchsatz 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

Erhöhung des Limits für den bereitgestellten Durchsatz

Das Limit für den bereitgestellten Durchsatz wurde für bestimmte erhöht AWS-Regionen. Weitere Informationen finden Sie unter [Gesamter bereitgestellter Standarddurchsatz für alle verbundenen Clients in jeder AWS-Region](#).

21. Juni 2023

Erweiterte Regionsunterstützung für EFS-Replikation

Die EFS-Replikation ist jetzt in allen verfügbar, AWS-Regionen in denen EFS verfügbar ist. Weitere Informationen finden Sie unter [Amazon-EFS-Replikation](#).

28. April 2023

Erhöhung des elastischen Durchsatzlimits

Das Limit für den elastischen Durchsatz wurde für bestimmte erhöht AWS-Regionen. Weitere Informationen finden Sie in der Tabelle [Gesamter elastischer Standarddurchsatz für alle verbundenen Clients in jeder AWS-Region](#).

17. April 2023

Elastic ersetzt Bursting als Standarddurchsatzmodus

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 der 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 wird 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 vorheriges Limit entfernt, bei dem die EFS-Replikation keine Sockets und benannten Pipes oder FIFOs unterstützt.

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 vorheriges Limit entfernt, bei dem maximal 256 Prozesse auf einer einzelnen Instance gleichzeitig Dateisperren verwenden können. 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 auf einem EFS-Dateisystem in ein anderes EFS-Dateisystem in der AWS-Region Ihrer Wahl zu replizieren. 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-stellige IDs zugewiesen. Weitere Informationen finden Sie unter [Ressourcen-IDs](#).

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-stelligen IDs auf 17-stellige IDs für Dateisysteme und Mount-Ziele um. Während dieses Übergangs können Sie sich anmelden und 17 Zeichen Ressourcen-IDs pro AWS-Region und verwenden. Weitere Informationen finden Sie unter [Ressourcen-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

[Neue Amazon-EFS-Konsole ist in AWS GovCloud \(US\) der Region verfügbar](#)

Die neue Amazon-EFS-Konsole ist jetzt in der verfügbar AWS GovCloud (US) AWS-Region.

10. Februar 2021

[Unterstützung für neue Amazon-EFS CloudWatch-Metriken hinzugefügt](#)
[MeteredIOBytes](#)

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 [Amazon CloudWatch-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

[Unterstützung für neue Amazon-EFS CloudWatch-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 [Amazon- CloudWatch Metriken für Amazon EFS.](#)

11. Januar 2021

[Verwenden von AWS Transfer Family für den Zugriff auf Amazon-EFS-Dateisysteme](#)

Sie können verwenden AWS Transfer Family , um Dateien in und aus Ihren Amazon-EFS-Dateisystemen zu übertragen. Weitere Informationen finden Sie unter [Verwenden von AWS Transfer Family für den Zugriff auf Dateien in Ihrem EFS-Dateisystem.](#)

06. Januar 2021

[Verwenden von AWS Systems Manager zur Verwaltung des Amazon-EFS-Clients \(amazon-efs-utils \)](#)

Sie können verwenden AWS Systems Manager , um die Amazon-EFS-Clients (amazon-efs-utils) automatisch auf Ihren EC2-Instances zu installieren oder zu aktualisieren. Weitere Informationen finden Sie unter [Verwenden von AWS Systems Manager zum automatischen Installieren oder Aktualisieren von Amazon-EFS-Clients.](#)

29. September 2020

[Erzwingen der Erstellung verschlüsselter EFS-Dateisysteme](#)

Sie können den elasticfilesystem:Encrypted AWS Identity and Access Management (IAM)-Bedingungsschlüssel 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 von bis zu 500 MB/s 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 von 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 [Create Your Amazon-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 den 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 AWS-Region Unterstützung hinzugefügt](#)

Amazon EFS ist jetzt für alle Benutzer in den Peking und Ningxia verfügbar AWS-Regionen.

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

[Unterstützung für 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 von 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 der Europa (Stockholm) verfügbar AWS-Region.

20. November 2019

[Zusätzliche AWS-Region
Unterstützung hinzugefügt](#)

Amazon EFS ist jetzt für alle Benutzer in der Asien-Pazifik (Hongkong) verfügbar AWS-Region.

20. November 2019

[Zusätzliche AWS-Region
Unterstützung hinzugefügt](#)

Amazon EFS ist jetzt für alle Benutzer in der 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 der Europa (Paris) verfügbar AWS-Region.

12. Juni 2019

[Zusätzliche AWS-Region Unterstützung hinzugefügt](#)

Amazon EFS ist jetzt für alle Benutzer in der Asien-Pazifik (Mumbai) verfügbar AWS-Region.

5. Juni 2019

[Zusätzliche AWS-Region Unterstützung hinzugefügt](#)

Amazon EFS ist jetzt für alle Benutzer in Kanada (Zentral) verfügbar AWS-Region.

1. Mai 2019

[API-Update: Tags sind jetzt Teil der Nutzlast der CreateFileSystem Operation](#)

Sie können jetzt Tags einschließen, wenn Sie die AWS API- und CLI- CreateFileSystem 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 der Europa (London) verfügbar AWS-Region.

23. Januar 2019

[AWS Backup Service-Integration mit Amazon EFS](#)

Amazon-EFS-Dateisysteme können mit gesichert werden AWS Backup, einem vollständig verwalteten, zentralisierten, automatisierten Backup-Service zum Sichern von Daten über - AWS Services in der Cloud und On-Premises. 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 Service.](#)

AWS DataSync ist ein verwalteter Datenübertragungsservice, der die Synchronisierung großer Datenmengen zwischen On-Premises-Speichersystemen und - AWS Speicherservices vereinfacht. Weitere Informationen finden Sie unter [Übertragen von Dateien von On-Premises-Dateisystemen zu Amazon EFS mit 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 On-Premises-Dateisystemen zu Amazon EFS mit 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 VPC</u> 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.

Änderung	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 Metriken hinzugefügt	Mit Metrikberechnungen können Sie mehrere CloudWatch Metriken abfragen und mathematische Ausdrücke verwenden, um neue Zeitreihen basierend auf diesen Metriken zu erstellen. Weitere Informationen finden Sie unter Verwenden von Metrikberechnungen mit Amazon EFS .	4. April 2018
Die amazon-efs-utils - Reihe von Open-Source-Tools und Verschlüsselung bei der Übertragung hinzugefügt	<p>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 von fallen keine zusätzlichen Kosten an amazon-efs-utils . Sie können diese Tools von herunterladen GitHub. Weitere Informationen finden Sie unter Verwenden der amazon-efs-utils Tools.</p> <p>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.</p>	4. April 2018
Aktualisierte Dateisystemlimits 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 in USA West (Nordkalifornien) verfügbar AWS-Region.	14. März 2018

Änderung	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 Markieren der Amazon-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.