User Guide

# AWS Transfer Family

# AWS Transfer Family: User Guide

# Table of Contents

Was ist AWS Transfer Family? ..... 1
Wie AWS Transfer Family funktioniert ..... 4
Blogbeiträge relevant für Transfer Family ..... 5
Voraussetzungen ..... 8
Regionen, Endpunkte und Kontingente ..... 8
Melde dich an für AWS ..... 8
Speicher konfigurieren ..... 9
Einen Amazon S3 S3-Bucket konfigurieren ..... 10
Ein Amazon EFS-Dateisystem konfigurieren ..... 15
Erstellen Sie eine IAM-Rolle und -Richtlinie ..... 18
Eine Benutzerrolle erstellen ..... 19
So funktionieren Sitzungsrichtlinien ..... 23
Beispiel für eine Lese-/Schreibzugriffsrichtlinie ..... 26
Tutorials zur Transfer Family ..... 30
Erste Schritte mit Serverendpunkten ..... 31
Voraussetzungen ..... 31
Anmelden bei der -Konsole ..... 32
Erstellen Sie einen SFTP-fähigen Server ..... 32
Fügen Sie einen vom Dienst verwalteten Benutzer hinzu ..... 34
Eine Datei mit einem Client übertragen ..... 35
Erstellen Sie einen Entschlüsselungs-Workflow ..... 37
Schritt 1: Konfigurieren Sie eine Ausführungsrolle ..... 38
Schritt 2: Erstellen Sie einen verwalteten Workflow ..... 39
Schritt 3: Fügen Sie den Workflow einem Server hinzu und erstellen Sie einen Benutzer ..... 40
Schritt 4: Erstellen Sie ein PGP-Schlüsselpaar ..... 42
Schritt 5: Speichern Sie den privaten PGP-Schlüssel in AWS Secrets Manager ..... 43
Schritt 6: Verschlüsseln Sie eine Datei ..... 44
Schritt 7: Führen Sie den Workflow aus und sehen Sie sich die Ergebnisse an ..... 45
SFTP-Konnektoren erstellen und verwenden ..... 46
Schritt 1: Erstellen Sie die erforderlichen unterstützenden Ressourcen ..... 47
Schritt 2: Erstellen und testen Sie einen SFTP-Connector ..... 52
Schritt 3: Senden und Abrufen von Dateien mithilfe des SFTP-Connectors ..... 57
Verfahren zum Erstellen eines Transfer Family Family-Servers, der als Remote-SFTP-Server verwendet werden kann ..... 60

Verwenden Sie einen benutzerdefinierten Identitätsanbieter ..... 63
Voraussetzungen ..... 63
Schritt 1: Erstellen Sie einen CloudFormation Stack ..... 64
Schritt 2: Überprüfen Sie die Konfiguration der API Gateway Gateway-Methode für Ihren Server ..... 65
Schritt 3: Die Transfer Family Family-Serverdetails anzeigen ..... 66
Schritt 4: Testen Sie, ob Ihr Benutzer eine Verbindung zum Server herstellen kann ..... 68
Schritt 5: Testen Sie die SFTP-Verbindung und die Dateiübertragung ..... 68
Schritt 6: Beschränken Sie den Zugriff auf den Bucket ..... 69
Lambda aktualisieren, wenn Sie Amazon EFS verwenden ..... 72
Richten Sie eine AS2-Konfiguration ein ..... 72
Schritt 1: Zertifikate für AS2 erstellen ..... 74
Schritt 2: Erstellen Sie einen Transfer Family Family-Server, der das AS2-Protokoll verwendet ..... 78
Schritt 3: Zertifikate als Transfer Family Family-Zertifikatsressourcen importieren ..... 82
Schritt 4: Erstellen Sie Profile für Sie und Ihren Handelspartner ..... 83
Schritt 5: Erstellen Sie eine Vereinbarung zwischen Ihnen und Ihrem Partner ..... 84
Schritt 6: Stellen Sie eine Verbindung zwischen Ihnen und Ihrem Partner her ..... 85
Schritt 7: Testen Sie den Austausch von Dateien über AS2 mithilfe von Transfer Family ..... 86
Transfer Family für SFTP, FTPS, FTP ..... 89
Optionen für den Identitätsanbieter ..... 89
AWS Transfer Family Endpunkt-Typmatrix ..... 91
Konfiguration eines Transfer Family Family-Serverendpunkts ..... 95
Erstellen Sie einen SFTP-fähigen Server ..... 97
Erstellen Sie einen FTPS-fähigen Server ..... 106
Erstellen Sie einen FTP-fähigen Server ..... 115
Einen Server in einer VPC erstellen ..... 124
Arbeiten mit benutzerdefinierten Hostnamen ..... 148
Dateien über den Serverendpunkt übertragen ..... 152
Verfügbare SFTP/FTPS/FTP-Befehle ..... 154
Finden Sie Ihren Amazon VPC-Endpunkt ..... 156
Vermeiden Sie Fehler `setstat` ..... 157
OpenSSH verwenden ..... 36
Verwenden Sie WinSCP ..... 159
Benutze Cyberduck ..... 35
Benutzen FileZilla ..... 163

Verwenden Sie einen Perl-Client ... 164
Verarbeitung nach dem Upload ... 165
Benutzer verwalten ... 166
Serviceverwaltete Benutzer ... 168
Benutzer von Verzeichnisdiensten ... 178
Benutzer von benutzerdefinierten Identitätsanbietern ... 196
Verwenden Sie logische Verzeichnisse ... 228
Regeln für die Verwendung logischer Verzeichnisse ... 230
Implementierung logischer Verzeichnisse und `chroot` ... 232
Beispiel für die Konfiguration logischer Verzeichnisse ... 234
Logische Verzeichnisse für Amazon EFS konfigurieren ... 235
Benutzerdefinierte AWS Lambda Antwort ... 236
SFTP-Anschlüsse ... 237
Konfigurieren Sie SFTP-Anschlüsse ... 237
Erstellen Sie einen SFTP-Connector ... 238
Speichern Sie ein Geheimnis zur Verwendung mit einem SFTP-Connector ... 247
Generieren und formatieren Sie den privaten Schlüssel des SFTP-Connectors ... 249
Testen Sie einen SFTP-Connector ... 252
Übertragen Sie Dateien mit SFTP-Anschlüssen ... 254
Listet den Inhalt eines Remote-Verzeichnisses auf ... 256
SFTP-Konnektoren verwalten ... 258
Aktualisieren Sie die SFTP-Konnektoren ... 258
Details zum SFTP-Connector anzeigen ... 259
Kontingente für SFTP-Konnektoren ... 261
Familie für AS2 übertragen ... 263
AS2-Anwendungsfälle ... 264
AS2 konfigurieren ... 269
Erstellen Sie einen AS2-Server mit der Transfer Family Family-Konsole ... 270
Erstellen Sie einen AS2-Server mithilfe einer Vorlage ... 273
AS2-Konfigurationen ... 277
AS2-Funktionen und -Fähigkeiten ... 283
AS2-Konnektoren konfigurieren ... 285
Erstellen Sie einen AS2-Connector ... 285
Algorithmen für AS2-Konnektoren ... 289
Standardauthentifizierung für AS2-Konnektoren ... 290
Aktivieren Sie die Standardauthentifizierung für AS2-Konnektoren ... 292

Konnektordetails anzeigen ... 296
AS2-Partner verwalten ... 298
AS2-Zertifikate importieren ... 298
Rotation der AS2-Zertifikate ... 300
Erstellen Sie AS2-Profile ... 302
Erstellen Sie AS2-Vereinbarungen ... 302
AS2-Nachrichten übertragen ... 304
AS2-Nachrichten senden ... 305
Empfangen Sie AS2-Nachrichten ... 306
Konfigurieren Sie HTTPS für AS2 ... 307
Dateien mit AS2-Anschlüssen übertragen ... 314
Dateinamen und Speicherorte ... 315
Statuscodes ... 317
JSON-Beispieldateien ... 318
Überwachen von AS2 ... 320
AS2-Statuscodes ... 322
AS2-Fehlercodes ... 323
Verwaltung von Workflows zur Dateiverarbeitung ... 338
Erstellen Sie einen Workflow ... 340
Konfigurieren Sie einen Workflow und führen Sie ihn aus ... 342
Workflow-Details anzeigen ... 344
Verwenden Sie vordefinierte Schritte ... 347
Datei kopieren ... 347
Datei entschlüsseln ... 352
Datei kennzeichnen ... 358
Datei löschen ... 359
Benannte Variablen für Workflows ... 360
Beispiel für einen Arbeitsablauf zum Markieren und Löschen ... 360
Verwenden Sie benutzerdefinierte Schritte zur Dateiverarbeitung ... 365
Mehrere Lambda-Funktionen nacheinander verwenden ... 367
Zugreifen auf eine Datei nach der benutzerdefinierten Verarbeitung ... 367
Beispiele für Ereignisse, an die AWS Lambda beim Hochladen einer Datei gesendet werden ... 368
Beispiel für eine Lambda-Funktion für einen benutzerdefinierten Workflow-Schritt ... 370
IAM-Berechtigungen für einen benutzerdefinierten Schritt ... 370
IAM-Richtlinien für Workflows ... 371

Vertrauensbeziehungen im Arbeitsablauf ... 373
Beispiel für eine Ausführungsrolle: Entschlüsseln, Kopieren und Markieren ... 373
Beispiel für eine Ausführungsrolle: Funktion ausführen und löschen ... 375
Ausnahmebehandlung für einen Workflow ... 376
Überwachen Sie die Workflow-Ausführung ... 377
CloudWatch Protokollierung für einen Workflow ... 377
CloudWatch Metriken für Workflows ... 380
Erstellen Sie einen Workflow aus einer Vorlage ... 380
Einen Workflow von einem Transfer Family Family-Server entfernen ... 384
Einschränkungen und Beschränkungen ... 386
Server verwalten ... 388
Eine Liste von Servern anzeigen ... 388
Löschen Sie einen Server ... 389
SFTP-Serverdetails anzeigen ... 390
AS2-Serverdetails anzeigen ... 392
Serverdetails bearbeiten ... 393
Bearbeiten Sie die Dateiübertragungsprotokolle ... 397
Bearbeiten Sie benutzerdefinierte Identitätsanbieter-Parameter ... 399
Bearbeiten Sie den Serverendpunkt ... 402
Bearbeiten Sie die Protokollierung ... 403
Bearbeiten Sie die Sicherheitsrichtlinie ... 404
Ändern Sie den verwalteten Workflow ... 405
Ändern Sie die Display-Banner für Ihren Server ... 406
Stellen Sie Ihren Server online oder offline ... 407
Server-Hostschlüssel verwalten ... 408
Fügen Sie einen zusätzlichen Server-Host-Schlüssel hinzu ... 409
Löscht einen Server-Hostschlüssel ... 410
Rotiert die Server-Hostschlüssel ... 411
Zusätzliche Schlüsselinformationen zum Server-Host ... 413
Überwachen Sie die Nutzung innerhalb der Konsole ... 414
Verwaltung der Zugriffskontrollen ... 417
Eine S3-Bucket-Zugriffsrichtlinie erstellen ... 418
Eine Sitzungsrichtlinie erstellen ... 419
Benutzer daran hindern, `mkdir` in einem S3-Bucket zu laufen ... 423
Protokollierung ... 424
CloudTrail Protokollierung ... 424

Aktivieren der CloudTrail Protokollierung ... 426
Beispielprotokolleintrag zum Erstellen eines Servers ... 426
CloudWatch Protokollierung ... 428
Arten der CloudWatch Protokollierung für Transfer Family ... 428
Protokollierung für Server erstellen ... 431
Verwaltung der Protokollierung für Workflows ... 439
Konfiguration einer Rolle für CloudWatch ... 442
Protokollstreams von Transfer Family anzeigen ... 444
CloudWatch Amazon-Alarme erstellen ... 448
Protokollierung von S3-API-Aufrufen in S3-Zugriffsprotokollen ... 448
Beispiele zur Begrenzung des Problems mit verwirrtem Stellvertreter ... 449
CloudWatch -Protokollstruktur für Transfer Family ... 451
Beispiel für CloudWatch Protokolleinträge ... 456
CloudWatch Metriken verwenden ... 461
Benutzerbenachrichtigungen ... 463
CloudWatch Abfragen ... 464
Ereignisse verwalten mit EventBridge ... 467
Transfer Family Ereignisse ... 468
SFTP-, FTPS- und FTP-Serverereignisse ... 468
Ereignisse des SFTP-Connectors ... 469
A2S-Ereignisse ... 470
Senden von Transfer Family Ereignissen ... 471
Erstellen von Ereignismustern ... 471
Testen von Ereignismustern für Transfer Family Ereignisse ... 473
Berechtigungen ... 473
Weitere Ressourcen ... 473
Detailreferenz zu Ereignissen ... 474
Serverereignisse ... 474
Connector-Ereignisse ... 479
AS2-Ereignisse ... 485
Sicherheit ... 492
Sicherheitsrichtlinien für Server ... 494
Kryptografische Algorithmen ... 495
TransferSecurityRichtlinie 2024-01 ... 504
TransferSecurityRichtlinie 2023-05 ... 505
TransferSecurityRichtlinie — 2022 — 03 ... 506

TransferSecurityPolitik — 2020-06 ..... 507
TransferSecurityRichtlinie 2018-11 ..... 508
TransferSecurityRichtlinie-FIPS-2024-01/ Richtlinie-FIPS-2024-05 TransferSecurity ..... 509
TransferSecurityRichtlinie-FIPS-2023-05 ..... 511
TransferSecurityRichtlinie-FIPS-2020-06 ..... 512
Sicherheitsrichtlinien nach Quantum ..... 513
Sicherheitsrichtlinien für SFTP-Konnektoren ..... 518
Sicherheitsrichtlinien für die Zeit nach Quantum ..... 520
Informationen zum hybriden Schlüsselaustausch in SSH nach der Quantenzeit ..... 522
Verwendung ..... 522
Testen ..... 524
Datenschutz ..... 527
Datenverschlüsselung ..... 529
Schlüsselmanagement in Transfer Family ..... 530
Identity and Access Management ..... 548
Zielgruppe ..... 548
Authentifizierung mit Identitäten ..... 549
Verwalten des Zugriffs mit Richtlinien ..... 553
Wie AWS Transfer Family funktioniert mit IAM ..... 556
Beispiele für identitätsbasierte Richtlinien ..... 561
Beispiel für eine tagbasierte Richtlinie ..... 564
Fehlerbehebung für -Identität und -Zugriff ..... 568
Compliance-Validierung ..... 570
Ausfallsicherheit ..... 571
Sicherheit der Infrastruktur ..... 572
Firewall für Webanwendungen ..... 573
Serviceübergreifende Confused-Deputy-Prävention ..... 574
Familienbenutzerrollen übertragen ..... 576
Workflow-Rollen für Transfer Family ..... 578
Rollen für die Protokollierung und den Aufruf von Transfer Family ..... 579
AWS verwaltete Richtlinien ..... 581
AWSTransferConsoleFullAccess ..... 581
AWSTransferFullAccess ..... 583
AWSTransferLoggingAccess ..... 585
AWSTransferReadOnlyAccess ..... 585
Richtlinienaktualisierungen ..... 586

Problembehebung bei Transfer Family ..... 588
Problembehandlung für vom Service verwaltete Benutzer ..... 588
Problembehandlung für vom Service verwaltete Amazon EFS-Benutzer ..... 589
Die Problembehandlung für den öffentlichen Schlüsseltext dauert zu lange ..... 589
Fehler beim Hinzufügen des öffentlichen SSH-Schlüssels zur Fehlerbehebung ..... 590
Probleme mit Amazon API Gateway beheben ..... 590
Zu viele Authentifizierungsfehler ..... 590
Die Verbindung wurde geschlossen ..... 592
Richtlinien zur Fehlerbehebung für verschlüsselte Amazon S3 S3-Buckets ..... 592
Beheben Sie Probleme mit der Authentifizierung ..... 593
Authentifizierungsfehler — SSH/SFTP ..... 593
Problem mit nicht übereinstimmenden verwalteten AD-Bereichen ..... 594
Verschiedene Probleme bei der Authentifizierung ..... 594
Beheben Sie Probleme mit verwalteten Workflows ..... 595
Workflow-bezogene Fehler mithilfe von Amazon beheben CloudWatch ..... 595
Beheben Sie Fehler beim Kopieren von Workflows ..... 597
Beheben Sie Probleme mit der Workflow-Entschlüsselung ..... 598
Beheben Sie den Fehler für die signierte Verschlüsselungsdatei ..... 598
Beheben Sie einen Fehler für einen FIPS-Algorithmus ..... 599
Probleme mit Amazon EFS beheben ..... 601
Beheben Sie das fehlende POSIX-Profil ..... 601
Problembehandlung bei logischen Verzeichnissen mit Amazon EFS ..... 602
Beheben Sie Fehler beim Testen Ihres Identitätsanbieters ..... 603
Beheben Sie Probleme beim Hinzufügen vertrauenswürdiger Hostschlüssel für Ihren SFTP-Connector ..... 603
Beheben Sie Probleme beim Hochladen von Dateien ..... 604
Fehler beim Hochladen von Amazon S3 S3-Dateien beheben ..... 604
Beheben Sie Probleme mit unlesbaren Dateinamen ..... 605
Beheben Sie die Ausnahme `ResourceNotFound` ..... 605
Beheben Sie Probleme mit dem SFTP-Connector ..... 606
Die Schlüsselverhandlung schlägt fehl ..... 606
Verschiedene Probleme mit dem SFTP-Connector ..... 607
Beheben Sie AS2-Probleme ..... 607
API-Referenz ..... 608
Willkommen ..... 608
Aktionen ..... 611

CreateAccess ... 614
CreateAgreement ... 622
CreateConnector ... 628
CreateProfile ... 636
CreateServer ... 641
CreateUser ... 655
CreateWorkflow ... 664
DeleteAccess ... 673
DeleteAgreement ... 676
DeleteCertificate ... 679
DeleteConnector ... 681
DeleteHostKey ... 683
DeleteProfile ... 686
DeleteServer ... 688
DeleteSshPublicKey ... 691
DeleteUser ... 695
DeleteWorkflow ... 698
DescribeAccess ... 701
DescribeAgreement ... 705
DescribeCertificate ... 708
DescribeConnector ... 711
DescribeExecution ... 714
DescribeHostKey ... 719
DescribeProfile ... 722
DescribeSecurityPolicy ... 725
DescribeServer ... 729
DescribeUser ... 734
DescribeWorkflow ... 739
ImportCertificate ... 744
ImportHostKey ... 750
ImportSshPublicKey ... 754
ListAccesses ... 759
ListAgreements ... 763
ListCertificates ... 767
ListConnectors ... 771
ListExecutions ... 774

ListHostKeys ... 779
ListProfiles ... 783
ListSecurityPolicies ... 787
ListServers ... 791
ListTagsForResource ... 795
ListUsers ... 800
ListWorkflows ... 805
SendWorkflowStepState ... 808
StartDirectoryListing ... 812
StartFileTransfer ... 818
StartServer ... 825
StopServer ... 828
TagResource ... 831
TestConnection ... 835
TestIdentityProvider ... 839
UntagResource ... 846
UpdateAccess ... 850
UpdateAgreement ... 857
UpdateCertificate ... 863
UpdateConnector ... 867
UpdateHostKey ... 873
UpdateProfile ... 877
UpdateServer ... 880
UpdateUser ... 893
Datentypen ... 900
As2ConnectorConfig ... 903
CopyStepDetails ... 907
CustomStepDetails ... 910
DecryptStepDetails ... 912
DeleteStepDetails ... 915
DescribedAccess ... 917
DescribedAgreement ... 921
DescribedCertificate ... 925
DescribedConnector ... 929
DescribedExecution ... 933
DescribedHostKey ... 936

DescribedProfile ..... 939
DescribedSecurityPolicy ..... 942
DescribedServer ..... 945
DescribedUser ..... 954
DescribedWorkflow ..... 959
EfsFileLocation ..... 961
EndpointDetails ..... 963
ExecutionError ..... 967
ExecutionResults ..... 969
ExecutionStepResult ..... 970
FileLocation ..... 972
HomeDirectoryMapEntry ..... 973
IdentityProviderDetails ..... 975
InputFileLocation ..... 978
ListedAccess ..... 979
ListedAgreement ..... 982
ListedCertificate ..... 985
ListedConnector ..... 988
ListedExecution ..... 990
ListedHostKey ..... 992
ListedProfile ..... 994
ListedServer ..... 996
ListedUser ..... 1000
ListedWorkflow ..... 1003
LoggingConfiguration ..... 1005
PosixProfile ..... 1007
ProtocolDetails ..... 1009
S3FileLocation ..... 1013
S3InputFileLocation ..... 1015
S3StorageOptions ..... 1017
S3Tag ..... 1018
ServiceMetadata ..... 1019
SftpConnectorConfig ..... 1020
SshPublicKey ..... 1022
Tag ..... 1024
TagStepDetails ..... 1026

UserDetails ..... 1028
WorkflowDetail ..... 1030
WorkflowDetails ..... 1032
WorkflowStep ..... 1034
API-Anfragen stellen ..... 1036
Erforderliche Anforderungsheader für Transfer Family ..... 1036
Eingaben und Unterschreiben von Familienanfragen übertragen ..... 1038
Fehlermeldungen ..... 1039
Verfügbare Bibliotheken ..... 1041
Geläufige Parameter ..... 1042
Häufige Fehler ..... 1044
Dokumentverlauf ..... 1047
AWS-Glossar ..... 1061
..... mlxii

# Was ist AWS Transfer Family?

AWS Transfer Family ist ein sicherer Übertragungsdienst, mit dem Sie Dateien in und aus AWS Speicherdiensten übertragen können. Transfer Family ist Teil der AWS Cloud Plattform. AWS Transfer Family bietet vollständig verwaltete Unterstützung für die Übertragung von Dateien über SFTP, AS2, FTPS und FTP direkt in und aus Amazon S3 oder Amazon EFS. Sie können Ihre Workflows für die Dateiübertragung nahtlos migrieren, automatisieren und überwachen, indem Sie bestehende clientseitige Konfigurationen für Authentifizierung, Zugriff und Firewalls beibehalten. Somit ändert sich nichts für Ihre Kunden, Partner und internen Teams oder deren Anwendungen.

Weitere Informationen und Informationen AWS zum Erstellen von Cloud-Anwendungen mit Amazon Web Services finden Sie unter Erste Schritte mit.

AWS Transfer Family unterstützt die Übertragung von Daten von oder zu den folgenden AWS Speicherdiensten.

- Speicher im Amazon Simple Storage Service (Amazon S3). Informationen zu Amazon S3 finden Sie unter Erste Schritte mit Amazon Simple Storage Service.
- Dateisysteme des Amazon Elastic File System (Amazon EFS) Network File System (NFS). Informationen zu Amazon EFS finden Sie unter Was ist Amazon Elastic File System?

AWS Transfer Family unterstützt die Übertragung von Daten über die folgenden Protokolle:

- Secure File Transfer Protocol (SFTP): Version 3

  Das offizielle IETF-Dokument ist hier: SSH File Transfer Protocol -02.txt. draft-ietf-secsh-filexfer
- Sicheres Dateiübertragungsprotokoll (FTPS)
- Dateiübertragungsprotokoll (FTP)
- Erklärung zur Anwendbarkeit 2 (AS2)

> **Note**
>
> Für FTP- und FTPS-Datenverbindungen liegt der Portbereich, den Transfer Family zum Einrichten des Datenkanals verwendet, zwischen 8192 und 8200.

Dateiübertragungsprotokolle werden in Datenaustausch-Workflows in verschiedenen Branchen verwendet, z. B. in den Bereichen Finanzdienstleistungen, Gesundheitswesen, Werbung und Einzelhandel. Transfer Family vereinfacht die Migration von Dateiübertragungsworkflows zu AWS.

Im Folgenden sind einige häufige Anwendungsfälle für die Verwendung von Transfer Family mit Amazon S3 aufgeführt:

- Daten werden AWS für Uploads von Drittanbietern wie Anbietern und Partnern genutzt.
- Abonnement-basierte Datenverteilung an Kunden
- Unternehmensinterne Übertragungen

Im Folgenden sind einige häufige Anwendungsfälle für die Verwendung von Transfer Family mit Amazon EFS aufgeführt:

- Datenverteilung
- Lieferkette
- Verwaltung von Inhalten
- Web-Server-Anwendungen

Im Folgenden sind einige häufige Anwendungsfälle für die Verwendung von Transfer Family mit AS2 aufgeführt:

- Workflows mit Compliance-Anforderungen, die darauf beruhen, dass Datenschutz- und Sicherheitsfunktionen in das Protokoll integriert sind
- Logistik in der Lieferkette
- Arbeitsabläufe im Zahlungsverkehr
- B usiness-to-business (B2B) -Transaktionen
- Integrationen mit Systemen für Unternehmensressourcenplanung (ERP) und Kundenbeziehungsmanagement (CRM)

Mit Transfer Family erhalten Sie Zugriff auf einen Server, der das Dateiübertragungsprotokoll unterstützt, AWS ohne dass Sie eine Serverinfrastruktur betreiben müssen. Sie können diesen Service verwenden, um Ihre auf Dateiübertragung basierenden Workflows zu migrieren und AWS gleichzeitig die Clients und Konfigurationen Ihrer Endbenutzer unverändert beizubehalten. Sie verknüpfen Ihren Hostnamen zunächst mit dem Serverendpunkt, fügen dann Ihre Benutzer

hinzu und gewähren ihnen die richtige Zugriffsebene. Nachdem Sie dies getan haben, werden die Übertragungsanfragen Ihrer Benutzer direkt von Ihrem Transfer Family Family-Serverendpunkt aus bearbeitet.

Transfer Family bietet die folgenden Vorteile:

- Ein vollständig verwalteter Service, der sich in Echtzeit an die Anforderungen anpasst.
- Sie müssen Ihre Anwendungen nicht ändern oder eine Infrastruktur für Dateiübertragungsprotokolle ausführen.
- Da sich Ihre Daten im dauerhaften Amazon S3 S3-Speicher befinden, können Sie native Funktionen AWS-Services für Verarbeitung, Analyse, Berichterstattung, Prüfung und Archivierung verwenden.
- Mit Amazon EFS als Datenspeicher erhalten Sie ein vollständig verwaltetes elastisches Dateisystem zur Verwendung mit AWS Cloud Services und lokalen Ressourcen. Amazon EFS ist so konzipiert, dass es bei Bedarf auf Petabytes skaliert werden kann, ohne Anwendungen zu unterbrechen. Es wächst und schrumpft automatisch, wenn Sie Dateien hinzufügen oder entfernen. Dadurch entfällt die Notwendigkeit, Kapazitäten bereitzustellen und zu verwalten, um dem Wachstum Rechnung zu tragen.
- Ein vollständig verwalteter, serverloser File Transfer Workflow-Dienst, der das Einrichten, Ausführen, Automatisieren und Überwachen der Verarbeitung von Dateien, die mit AWS Transfer Family hochgeladen wurden, vereinfacht.
- Es sind keine Investitionen erforderlich und die Kosten für den Service sind nutzungsbezogen.

In den folgenden Abschnitten finden Sie eine Beschreibung der verschiedenen Funktionen von Transfer Family, ein Tutorial für die ersten Schritte, detaillierte Anweisungen zur Einrichtung der verschiedenen protokollfähigen Server, zur Verwendung verschiedener Arten von Identitätsanbietern und die API-Referenz des Dienstes.

Informationen zu den ersten Schritten mit Transfer Family finden Sie im Folgenden:

- Wie AWS Transfer Family funktioniert
- Voraussetzungen
- Erste Schritte mit AWS Transfer Family Serverendpunkten

# Wie AWS Transfer Family funktioniert

AWS Transfer Family ist ein vollständig verwalteter AWS Service, mit dem Sie Dateien über die folgenden Protokolle in und aus dem Amazon Simple Storage Service (Amazon S3) -Speicher oder Amazon Elastic File System (Amazon EFS) -Dateisysteme übertragen können:

- Secure File Transfer Protocol (SFTP): Version 3

  Das offizielle IETF-Dokument ist hier: SSH File Transfer Protocol -02.txt. draft-ietf-secsh-filexfer
- Sicheres Dateiübertragungsprotokoll (FTPS)
- Dateiübertragungsprotokoll (FTP)
- Erklärung zur Anwendbarkeit 2 (AS2)

AWS Transfer Family unterstützt bis zu 3 Availability Zones und wird durch eine automatisch skalierbare, redundante Flotte für Ihre Verbindungs- und Übertragungsanfragen unterstützt. Ein Beispiel dafür, wie Sie mithilfe von latenzbasiertem Routing eine höhere Redundanz erreichen und die Netzwerklatenz minimieren können, finden Sie im Blogbeitrag Minimiere die Netzwerklatenz bei der AWS Übertragung für SFTP-Server.

Transfer Family Managed File Transfer Workflows (MFTW) ist ein vollständig verwalteter, serverloser File Transfer Workflow-Dienst, der es einfach macht, die Verarbeitung von Dateien einzurichten, auszuführen, zu automatisieren und zu überwachen, die mit AWS Transfer Family Kunden können MFTW verwenden, um verschiedene Verarbeitungsschritte wie Kopieren, Markieren, Scannen, Filtern, Komprimieren/Dekomprimieren und Verschlüsseln/Entschlüsseln der mit Transfer Family übertragenen Daten zu automatisieren. Dies bietet umfassende Transparenz für Nachverfolgung und Überprüfbarkeit. Weitere Details finden Sie unter AWS Transfer Family verwaltete Workflows.

AWS Transfer Family unterstützt jeden Standard-Client für das Dateiübertragungsprotokoll. Einige häufig verwendete Clients sind die folgenden:

- OpenSSH — Ein Befehlszeilenprogramm für Macintosh und Linux.
- WinSCP — Ein grafischer Client nur für Windows.
- Cyberduck — Ein grafischer Linux-, Macintosh- und Microsoft Windows-Client.
- FileZilla— Ein grafischer Linux-, Macintosh- und Windows-Client.

AWS bietet die folgenden Transfer Family Family-Workshops an.

- Entwickeln Sie eine Dateiübertragungslösung, die verwaltete SFTP/FTPS-Endpunkte sowie Amazon Cognito und DynamoDB AWS Transfer Family für die Benutzerverwaltung nutzt. Die Einzelheiten zu diesem Workshop finden Sie hier.
- Erstellen Sie einen Transfer Family-Endpunkt mit aktiviertem AS2 und einen Transfer Family AS2-Connector. Die Details für diesen Workshop finden Sie hier.
- Entwickeln Sie eine Lösung, die präskriptive Anleitungen und ein praktisches Lab bietet, auf der Sie erfahren, wie Sie eine skalierbare und sichere Dateiübertragungsarchitektur aufbauen können, AWS ohne bestehende Anwendungen ändern oder die Serverinfrastruktur verwalten zu müssen. Die Einzelheiten zu diesem Workshop finden Sie hier.

## Blogbeiträge relevant für Transfer Family

In der folgenden Tabelle sind die Blogbeiträge aufgeführt, die nützliche Informationen für Kunden von Transfer Family enthalten. Die Tabelle ist in umgekehrter chronologischer Reihenfolge angeordnet, sodass die neuesten Beiträge am Anfang der Tabelle stehen.

| Titel und Link des Blogbeitrags | Datum |
| --- | --- |
| Architektur sicherer und richtlinienkonformer verwalteter Dateiübertragungen mit AWS Transfer Family SFTP-Konnektoren und PGP-Verschlüsselung | 16. Mai 2024 |
| Verwenden von Amazon Cognito als Identitätsanbieter mit AWS Transfer Family Amazon S3 | 14. Mai 2024 |
| Wie Transfer Family Ihnen helfen kann, eine sichere, konforme verwaltete Dateiübertragungslösung zu entwickeln | 3. Januar 2024 |
| Erkennen Sie Malware-Bedrohungen mit AWS Transfer Family | 20. Juli 2023 |
| Erweiterung der SAP-Workloads mit AWS Transfer Family | 13. Juli 2023 |

| Titel und Link des Blogbeitrags | Datum |
| --- | --- |
| Verschlüsseln und entschlüsseln Sie Dateien mit PGP und AWS Transfer Family | 21. Juni 2023 |
| Authentifizierung AWS Transfer Family mit Azure Active Directory und AWS Lambda | 15. Dezember 2022 |
| Passen Sie Benachrichtigungen zur Dateizustellung mithilfe AWS Transfer Family verwalteter Workflows an | 14. Oktober 2022 |
| Aufbau einer Cloud-nativen Dateiübertragungsplattform mithilfe von Workflows AWS Transfer Family | 5. Januar 2022 |
| Aktivierung der Self-Service-Schlüsselverwaltung für Benutzer mit A AWS Transfer Family und. AWS Lambda | 17. Dezember 2021 |
| Verbessern Sie die Datenzugriffskontrolle mit AWS Transfer Family Amazon S3 | 5. Oktober 2021 |
| Verbessern Sie den Durchsatz für Dateiübertragungen mit Internetzugriff mithilfe von AWS Global AcceleratorAWS Transfer Family Services | 7. Juni 2021 |
| Sicherung AWS Transfer Family mit AWS Web Application Firewall und Amazon API Gateway | 5. Mai 2021 |
| Sicherung AWS Transfer Family mit AWS Web Application Firewall und Amazon API Gateway | 15. Januar 2021 |
| AWS Transfer Family Unterstützung für Amazon Elastic File System | 7. Januar 2021 |

| Titel und Link des Blogbeitrags | Datum |
| --- | --- |
| Aktivieren Sie die Passwortauthentifizierung für die AWS Transfer Family Verwendung AWS Secrets Manager | 5. November 2020 |
| Zentralisieren Sie den Datenzugriff mit AWS Transfer Family und AWS Storage Gateway | 22. Juni 2020 |
| Verwenden von Amazon EFS für AWS Lambda in Ihren serverlosen Anwendungen | 18. Juni 2020 |
| Verwenden Sie die IP-Zulassungsliste, um Ihre AWS Transfer Family Server zu sichern | 8. April 2020 |
| Minimiere die Netzwerklatenz mit deiner AWS Übertragung für SFTP-Server | 19. Februar 2020 |
| Lift-and-Shift-Migration von SFTP-Servern zu AWS | 12. Februar 2020 |
| Vereinfachen Sie Ihre AWS SFTP-Struktur mit Chroot und logischen Verzeichnissen | 26. September 2019 |
| Verwenden Sie Okta als Identitätsanbieter mit AWS Transfer Family | 30. Mai 2019 |

# Voraussetzungen

In den folgenden Abschnitten werden die Voraussetzungen beschrieben, die für die Nutzung des AWS Transfer Family Dienstes erforderlich sind. Sie müssen mindestens einen Amazon Simple Storage Service (Amazon S3) -Bucket erstellen und über eine AWS Identity and Access Management (IAM) -Rolle Zugriff auf diesen Bucket gewähren. Die Rolle muss zudem eine Vertrauensstellung einrichten. Diese Vertrauensstellung ermöglicht es Transfer Family, die IAM-Rolle für den Zugriff auf Ihren Bucket zu übernehmen, sodass er die Dateiübertragungsanfragen Ihrer Benutzer bearbeiten kann.

Themen


- Unterstützte AWS Regionen, Endpunkte und Kontingente
- Melden Sie sich an für AWS
- Konfigurieren Sie den Speicher für die Verwendung mit AWS Transfer Family
- Erstellen Sie eine IAM-Rolle und -Richtlinie


## Unterstützte AWS Regionen, Endpunkte und Kontingente

Um programmgesteuert eine Verbindung zu einem AWS Dienst herzustellen, verwenden Sie einen Endpunkt. Der Endpunkt für Kunden in der Region USA Ost (Ohio) (`us-east-2`) ist beispielsweise. `transfer.us-east-2.amazonaws.com` Service Quotas, auch als Limits bezeichnet, sind die maximale Anzahl von Serviceressourcen oder -vorgängen für Ihr AWS-Konto. In diesem Handbuch finden Sie Kontingente in AS2-Kontingente undKontingente für SFTP-Konnektoren.

Weitere Informationen zu unterstützten AWS Regionen, Endpunkten und Servicekontingenten finden Sie unter AWS Transfer Family Endpunkte und Kontingente in der. Allgemeine Amazon Web Services-Referenz

## Melden Sie sich an für AWS

Wenn Sie sich für Amazon Web Services (AWS) registrieren, wird Ihr AWS Konto automatisch für alle Dienste in angemeldet AWS, einschließlich AWS Transfer Family. Berechnet werden Ihnen aber nur die Services, die Sie nutzen.

Wenn Sie bereits ein AWS Konto haben, fahren Sie mit der nächsten Aufgabe fort. Wenn Sie kein AWS -Konto haben, führen Sie die folgenden Schritte zum Erstellen eines Kontos aus.

Wenn Sie noch kein Konto haben AWS-Konto, führen Sie die folgenden Schritte aus, um eines zu erstellen.

Um sich für eine anzumelden 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 eine anmelden AWS-Konto, Root-Benutzer des AWS-Kontoswird eine erstellt. Der Root-Benutzer hat Zugriff auf alle AWS-Services und Ressourcen des Kontos. Aus Sicherheitsgründen sollten Sie einem Benutzer Administratorzugriff zuweisen und nur den Root-Benutzer verwenden, um Aufgaben auszuführen, für die Root-Benutzerzugriff erforderlich ist.

Informationen zur Preisgestaltung und zur Schätzung der Kosten für die Nutzung von Transfer Family finden Sie unter AWS Transfer Family Preise. AWS Pricing Calculator

Informationen zur AWS regionalen Verfügbarkeit finden Sie unter den AWS Transfer Family Endpunkten und Kontingenten in der Allgemeine AWS-Referenz.

# Konfigurieren Sie den Speicher für die Verwendung mit AWS Transfer Family

In diesem Thema werden die Speicheroptionen beschrieben, die Sie zusammen verwenden können AWS Transfer Family. Sie können entweder Amazon S3 oder Amazon EFS als Speicher für Ihre Transfer Family Family-Server verwenden.

Inhalt


- Einen Amazon S3 S3-Bucket konfigurieren
  - Amazon S3 Access Points
  - HeadObject Verhalten von Amazon S3
    - Gewähren Sie die Möglichkeit, nur Dateien zu schreiben und aufzulisten
    - Große Anzahl von Null-Byte-Objekten verursacht Latenzprobleme
- Ein Amazon EFS-Dateisystem konfigurieren

- Besitz von Amazon EFS-Dateien
- Amazon EFS-Benutzer für Transfer Family einrichten
  - Transfer Family Family-Benutzer auf Amazon EFS konfigurieren
  - Erstellen Sie einen Amazon EFS-Root-Benutzer
- Unterstützte Amazon EFS-Befehle

## Einen Amazon S3 S3-Bucket konfigurieren

AWS Transfer Family greift auf Ihren Amazon S3 S3-Bucket zu, um die Übertragungsanfragen Ihrer Benutzer zu bearbeiten. Daher müssen Sie im Rahmen der Einrichtung Ihres File-Transfer-Protokoll-fähigen Servers einen Amazon S3 S3-Bucket bereitstellen. Sie können einen vorhandenen Bucket verwenden oder einen neuen Bucket erstellen.

> **Note**
>
> Sie müssen keinen Server und keinen Amazon S3 S3-Bucket verwenden, die sich in derselben AWS Region befinden, aber wir empfehlen dies als bewährte Methode.

Wenn Sie Ihre Benutzer einrichten, weisen Sie ihnen jeweils eine IAM-Rolle zu. Diese Rolle bestimmt die Zugriffsebene, die sie auf Ihren Amazon S3 S3-Bucket haben.

Informationen zum Erstellen eines neuen Buckets finden Sie unter Wie erstelle ich einen S3-Bucket? im Amazon Simple Storage Service-Benutzerhandbuch.

> **Note**
>
> Sie können Amazon S3 Object Lock verwenden, um zu verhindern, dass Objekte für einen bestimmten Zeitraum oder auf unbestimmte Zeit überschrieben werden. Dies funktioniert bei Transfer Family genauso wie bei anderen Diensten. Wenn ein Objekt existiert und geschützt ist, ist es nicht erlaubt, in diese Datei zu schreiben oder sie zu löschen. Weitere Informationen zu Amazon S3 Object Lock finden Sie unter Verwenden von Amazon S3 Object Lock im Amazon Simple Storage Service-Benutzerhandbuch.

## Amazon S3 Access Points

AWS Transfer Family unterstützt Amazon S3 Access Points, eine Funktion von Amazon S3, mit der Sie den detaillierten Zugriff auf gemeinsam genutzte Datensätze einfach verwalten können. Sie können S3 Access Point-Aliase überall verwenden, wo Sie einen S3-Bucket-Namen verwenden. Sie können in Amazon S3 Hunderte von Access Points für Benutzer erstellen, die über unterschiedliche Berechtigungen für den Zugriff auf gemeinsam genutzte Daten in einem Amazon S3 S3-Bucket verfügen.

Beispielsweise können Sie Access Points verwenden, um drei verschiedenen Teams den Zugriff auf denselben gemeinsamen Datensatz zu ermöglichen, wobei ein Team Daten aus S3 lesen kann, ein zweites Team Daten in S3 schreiben kann und das dritte Team Daten aus S3 lesen, schreiben und löschen kann. Um eine detaillierte Zugriffskontrolle wie oben erwähnt zu implementieren, können Sie einen S3-Zugriffspunkt erstellen, der eine Richtlinie enthält, die verschiedenen Teams asymmetrischen Zugriff gewährt. Sie können S3-Zugriffspunkte mit Ihrem Transfer Family Family-Server verwenden, um eine differenzierte Zugriffskontrolle zu erreichen, ohne eine komplexe S3-Bucket-Richtlinie zu erstellen, die Hunderte von Anwendungsfällen umfasst. Weitere Informationen zur Verwendung von S3 Access Points mit einem Transfer Family Family-Server finden Sie im Blogbeitrag Enhance data access control with AWS Transfer Family and Amazon S3.

> **Note**
>
> AWS Transfer Family unterstützt derzeit keine Amazon S3 Multi-Region Access Points.

## HeadObject Verhalten von Amazon S3

> **Note**
>
> Wenn Sie einen Transfer Family Family-Server erstellen oder aktualisieren, können Sie die Leistung Ihrer Amazon S3 S3-Verzeichnisse optimieren, wodurch `HeadObject` Anrufe vermieden werden.

In Amazon S3 sind Buckets und Objekte die primären Ressourcen, in denen Objekte in Buckets gespeichert werden. Amazon S3 kann ein hierarchisches Dateisystem nachahmen, sich aber manchmal anders verhalten als ein typisches Dateisystem. Beispielsweise sind Verzeichnisse in Amazon S3 kein erstklassiges Konzept, sondern basieren stattdessen auf Objektschlüsseln. AWS

Transfer Family leitet einen Verzeichnispfad ab, indem der Schlüssel eines Objekts durch den Schrägstrich (/) geteilt wird, das letzte Element als Dateinamen behandelt und dann Dateinamen, die dasselbe Präfix haben, unter demselben Pfad gruppiert werden. Null-Byte-Objekte werden erstellt, um den Pfad eines Ordners darzustellen, wenn Sie mit `mkdir` oder mithilfe der Amazon S3 S3-Konsole ein leeres Verzeichnis erstellen. Der Schlüssel für diese Objekte endet mit einem abschließenden Schrägstrich. Diese Null-Byte-Objekte werden unter Organisieren von Objekten in der Amazon S3 S3-Konsole mithilfe von Ordnern im Amazon S3 S3-Benutzerhandbuch beschrieben.

Wenn Sie einen `ls` Befehl ausführen und einige Ergebnisse Amazon S3 S3-Null-Byte-Objekte sind (diese Objekte haben Schlüssel, die mit einem Schrägstrich enden), gibt Transfer Family eine `HeadObject` Anfrage für jedes dieser Objekte aus (Einzelheiten finden Sie HeadObjectin der Amazon Simple Storage Service API-Referenz). Dies kann zu den folgenden Problemen führen, wenn Sie Amazon S3 als Speicher mit Transfer Family verwenden.

Gewähren Sie die Möglichkeit, nur Dateien zu schreiben und aufzulisten

In einigen Fällen möchten Sie möglicherweise nur Schreibzugriff auf Ihre Amazon S3 S3-Objekte anbieten. Beispielsweise möchten Sie möglicherweise Zugriff auf Schreib- (oder Upload) und Auflisten von Objekten in einem Bucket gewähren, nicht jedoch auf Objekte zum Lesen (Herunterladen). Um `mkdir` Befehle mithilfe von Dateiübertragungsclients ausführen `ls` zu können, benötigen Sie Amazon S3 `ListObjects` und die `PutObject` entsprechenden Berechtigungen. Wenn Transfer Family jedoch einen `HeadObject` Aufruf tätigen muss, um Dateien zu schreiben oder aufzulisten, schlägt der Anruf mit der Fehlermeldung Zugriff verweigert fehl, da für diesen Aufruf die `GetObject` entsprechende Genehmigung erforderlich ist.

> **Note**
>
> Wenn Sie einen Transfer Family Family-Server erstellen oder aktualisieren, können Sie die Leistung Ihrer Amazon S3 S3-Verzeichnisse optimieren, wodurch `HeadObject` Anrufe vermieden werden.

In diesem Fall können Sie Zugriff gewähren, indem Sie eine AWS Identity and Access Management (IAM-) Richtlinienbedingung hinzufügen, die die `GetObject` Berechtigung nur für Objekte hinzufügt, die mit einem Schrägstrich (`/`) enden. Diese Bedingung verhindert das `GetObject` Aufrufen von Dateien (sodass sie nicht gelesen werden können), ermöglicht es dem Benutzer jedoch, Ordner aufzulisten und zu durchsuchen. Die folgende Beispielrichtlinie bietet nur Schreib- und Listenzugriff

auf Ihre Amazon S3 S3-Buckets. Um diese Richtlinie zu verwenden, *DOC-EXAMPLE-BUCKET* ersetzen Sie sie durch den Namen Ihres Buckets.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowListing",
            "Effect": "Allow",
            "Action": "s3:ListBucket",
            "Resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET"
        },
        {
            "Sid": "AllowReadWrite",
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
                "s3:GetObjectVersion"
            ],
            "Resource": [
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
            ]
         },
      {
            "Sid": "DenyIfNotFolder",
            "Effect": "Deny",
            "Action": [
                "s3:GetObject",
                "s3:GetObjectVersion"
            ],
            "NotResource": [
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*/"
            ]
         }
      ]
}
```

> **Note**
>
> Diese Richtlinie erlaubt es Benutzern nicht, Dateien anzuhängen. Mit anderen Worten, ein Benutzer, dem diese Richtlinie zugewiesen wurde, kann keine Dateien öffnen, um ihnen

Inhalte hinzuzufügen oder sie zu ändern. Wenn Ihr Anwendungsfall außerdem vor dem Hochladen einer Datei einen `HeadObject` Anruf erfordert, funktioniert diese Richtlinie nicht für Sie.

Große Anzahl von Null-Byte-Objekten verursacht Latenzprobleme

Wenn Ihre Amazon S3 S3-Buckets eine große Anzahl dieser Null-Byte-Objekte enthalten, gibt Transfer Family viele `HeadObject` Aufrufe aus, was zu Verarbeitungsverzögerungen führen kann. Die empfohlene Lösung für dieses Problem besteht darin, Optimized Directories zu aktivieren, um die Latenz zu reduzieren.

Nehmen wir zum Beispiel an, Sie gehen in Ihr Home-Verzeichnis und Sie haben 10.000 Unterverzeichnisse. Mit anderen Worten, Ihr Amazon S3 S3-Bucket hat 10.000 Ordner. Wenn Sie in diesem Szenario den Befehl `ls` (list) ausführen, dauert der Listenvorgang zwischen sechs und acht Minuten. Wenn Sie Ihre Verzeichnisse optimieren, dauert dieser Vorgang jedoch nur wenige Sekunden. Sie legen diese Option während der Servererstellung oder -aktualisierung im Fenster Zusätzliche Details konfigurieren fest. Diese Verfahren werden unter dem Konfiguration eines SFTP-, FTPS- oder FTP-Serverendpunkts Thema detailliert beschrieben.

> **Note**
>
> GUI-Clients geben möglicherweise einen `ls` Befehl aus, auf den Sie keinen Einfluss haben. Daher ist es wichtig, diese Einstellung zu aktivieren, wenn Sie können.

Wenn Sie Ihre Verzeichnisse nicht optimieren oder nicht optimieren können, besteht eine alternative Lösung für dieses Problem darin, alle Ihre Null-Byte-Objekte zu löschen. Beachten Sie Folgendes:

- Leere Verzeichnisse werden nicht mehr existieren. Verzeichnisse existieren nur, weil ihre Namen im Schlüssel eines Objekts stehen.
- Hindert jemanden nicht daran, erneut anzurufen `mkdir` und Dinge kaputt zu machen. Sie könnten dies mildern, indem Sie eine Richtlinie erstellen, die die Erstellung von Verzeichnissen verhindert.
- In einigen Szenarien werden diese 0-Byte-Objekte verwendet. Sie haben beispielsweise eine Struktur wie /inboxes/customer1000 und das Posteingangsverzeichnis wird täglich gereinigt.

Schließlich besteht eine weitere mögliche Lösung darin, die Anzahl der sichtbaren Objekte durch eine Richtlinienbedingung zu begrenzen, um die Anzahl der Aufrufe zu reduzieren. `HeadObject`

Damit dies eine praktikable Lösung ist, müssen Sie akzeptieren, dass Sie möglicherweise nur eine begrenzte Anzahl all Ihrer Unterverzeichnisse anzeigen können.

## Ein Amazon EFS-Dateisystem konfigurieren

AWS Transfer Family greift auf Amazon Elastic File System (Amazon EFS) zu, um die Übertragungsanfragen Ihrer Benutzer zu bearbeiten. Daher müssen Sie im Rahmen der Einrichtung Ihres File-Transfer-Protokoll-fähigen Servers ein Amazon EFS-Dateisystem bereitstellen. Sie können ein vorhandenes Dateisystem verwenden oder ein neues erstellen.

Beachten Sie Folgendes:

- Wenn Sie einen Transfer Family Family-Server und ein Amazon EFS-Dateisystem verwenden, müssen sich der Server und das Dateisystem im selben System befinden AWS-Region.
- Der Server und das Dateisystem müssen sich nicht im selben Konto befinden. Wenn sich der Server und das Dateisystem nicht in demselben Konto befinden, muss die Dateisystemrichtlinie der Benutzerrolle eine ausdrückliche Genehmigung erteilen.

  Informationen zum Einrichten mehrerer Konten finden Sie im AWS Organizations Benutzerhandbuch unter Verwaltung der AWS Konten in Ihrer Organisation.
- Wenn Sie Ihre Benutzer einrichten, weisen Sie ihnen jeweils eine IAM-Rolle zu. Diese Rolle bestimmt die Zugriffsebene, die sie auf Ihr Amazon EFS-Dateisystem haben.
- Einzelheiten zum Mounten eines Amazon EFS-Dateisystems finden Sie unter Bereitstellen von Amazon EFS-Dateisystemen.

Weitere Informationen zur Zusammenarbeit mit AWS Transfer Family Amazon EFS finden Sie unter Verwenden für AWS Transfer Family den Zugriff auf Dateien in Ihrem Amazon EFS-Dateisystem im Amazon Elastic File System-Benutzerhandbuch.

### Besitz von Amazon EFS-Dateien

Amazon EFS verwendet das POSIX-Dateiberechtigungsmodell (Portable Operating System Interface), um den Dateibesitz darzustellen.

In POSIX werden Benutzer im System in drei verschiedene Berechtigungsklassen eingeteilt: Wenn Sie einem Benutzer den Zugriff auf Dateien ermöglichen, die in einem Amazon EFS-Dateisystem gespeichert sind AWS Transfer Family, müssen Sie ihm ein „POSIX-Profil“ zuweisen. Dieses Profil wird verwendet, um ihren Zugriff auf Dateien und Verzeichnisse im Amazon EFS-Dateisystem zu bestimmen.

- Benutzer (u): Besitzer der Datei oder des Verzeichnisses. Normalerweise ist der Ersteller einer Datei oder eines Verzeichnisses auch der Eigentümer.
- Gruppe (g): Gruppe von Benutzern, die identischen Zugriff auf Dateien und Verzeichnisse benötigen, die sie gemeinsam nutzen.
- Andere (o): Alle anderen Benutzer, die Zugriff auf das System haben, mit Ausnahme des Besitzers und der Gruppenmitglieder. Diese Berechtigungsklasse wird auch als „Öffentlich“ bezeichnet.

Im POSIX-Berechtigungsmodell ist jedes Dateisystemobjekt (Dateien, Verzeichnisse, symbolische Links, Named Pipes und Sockets) mit den zuvor genannten drei Berechtigungssätzen verknüpft. Amazon EFS-Objekten ist ein Modus im UNIX-Stil zugeordnet. Dieser Moduswert definiert die Berechtigungen zum Ausführen von Aktionen für dieses Objekt.

Darüber hinaus werden Benutzer und Gruppen auf Unix-Systemen numerischen Bezeichnern zugeordnet, die Amazon EFS zur Darstellung von Dateibesitz verwendet. Bei Amazon EFS gehören Objekte einem einzelnen Besitzer und einer einzelnen Gruppe. Amazon EFS verwendet die zugeordneten numerischen IDs, um die Berechtigungen zu prüfen, wenn ein Benutzer versucht, auf ein Dateisystemobjekt zuzugreifen.

## Amazon EFS-Benutzer für Transfer Family einrichten

Bevor Sie Ihre Amazon EFS-Benutzer einrichten, können Sie einen der folgenden Schritte ausführen:

- Sie können Benutzer erstellen und ihre Basisordner in Amazon EFS einrichten. Details dazu finden Sie unter Transfer Family Family-Benutzer auf Amazon EFS konfigurieren.
- Wenn Sie mit dem Hinzufügen eines Root-Benutzers vertraut sind, können Sie das tunErstellen Sie einen Amazon EFS-Root-Benutzer.

> **Note**
>
> Transfer Family Family-Server unterstützen keine Amazon EFS-Zugriffspunkte zur Festlegung von POSIX-Berechtigungen. Die POSIX-Profile von Transfer Family Family-Benutzern (im vorherigen Abschnitt beschrieben) bieten die Möglichkeit, POSIX-Berechtigungen festzulegen. Diese Berechtigungen werden auf Benutzerebene für einen detaillierten Zugriff auf der Grundlage von UID, GID und sekundären GIDs festgelegt.

Transfer Family Family-Benutzer auf Amazon EFS konfigurieren

Transfer Family ordnet die Benutzer der von Ihnen angegebenen UID/GID und den von Ihnen angegebenen Verzeichnissen zu. Wenn die UID/GID/Verzeichnisse noch nicht in EFS vorhanden sind, sollten Sie sie erstellen, bevor Sie sie in Transfer an einen Benutzer zuweisen. Die Einzelheiten zum Erstellen von Amazon EFS-Benutzern werden unter Arbeiten mit Benutzern, Gruppen und Berechtigungen auf Netzwerkdateisystemebene (NFS) im Amazon Elastic File System-Benutzerhandbuch beschrieben.

Schritte zum Einrichten von Amazon EFS-Benutzern in Transfer Family

1. Ordnen Sie die EFS-UID und GID für Ihren Benutzer in Transfer Family mithilfe der `PosixProfile`Felder zu.
2. Wenn Sie möchten, dass der Benutzer bei der Anmeldung in einem bestimmten Ordner startet, können Sie das EFS-Verzeichnis unter dem `HomeDirectory`Feld angeben.

Sie können den Prozess automatisieren, indem Sie eine CloudWatch Regel und eine Lambda-Funktion verwenden. Ein Beispiel für eine Lambda-Funktion, die mit EFS interagiert, finden Sie unter Verwenden von Amazon EFS für AWS Lambda in Ihren serverlosen Anwendungen.

Darüber hinaus können Sie logische Verzeichnisse für Ihre Transfer Family Family-Benutzer konfigurieren. Einzelheiten finden Sie im Logische Verzeichnisse für Amazon EFS konfigurieren Abschnitt des Verwendung logischer Verzeichnisse zur Vereinfachung Ihrer Transfer Family Family-Verzeichnisstrukturen Themas.

Erstellen Sie einen Amazon EFS-Root-Benutzer

Wenn Ihre Organisation bereit ist, den Root-Benutzerzugriff über SFTP/FTPS für die Konfiguration Ihrer Benutzer zu aktivieren, können Sie einen Benutzer erstellen, dessen UID und GID 0 sind (Root-Benutzer), dann diesen Root-Benutzer verwenden, um Ordner zu erstellen und den übrigen Benutzern POSIX-ID-Besitzer zuzuweisen. Der Vorteil dieser Option besteht darin, dass das Amazon EFS-Dateisystem nicht bereitgestellt werden muss.

Führen Sie die unter beschriebenen Schritte aus und geben Sie sowohl für die Benutzer-ID als auch für die Gruppen-ID 0 (Null) ein. Hinzufügen von serviceverwalteten Amazon-EFS-Benutzern

## Unterstützte Amazon EFS-Befehle

Die folgenden Befehle werden für Amazon EFS for unterstützt AWS Transfer Family.

- `cd`
- `ls/dir`
- `pwd`
- `put`
- `get`
- `rename`
- `chown`: Nur Root-Benutzer (d. h. Benutzer mit uid=0) können den Besitz und die Berechtigungen von Dateien und Verzeichnissen ändern.
- `chmod`: Nur Root kann den Besitz und die Berechtigungen von Dateien und Verzeichnissen ändern.
- `chgrp`: Wird entweder für Root-Benutzer oder für den Eigentümer der Datei unterstützt, der die Gruppe einer Datei nur in eine seiner sekundären Gruppen ändern kann.
- `ln -s/symlink`
- `mkdir`
- `rm/delete`
- `rmdir`
- `chmtime`

## Erstellen Sie eine IAM-Rolle und -Richtlinie

In diesem Thema werden die Arten von Richtlinien und Rollen beschrieben, die zusammen verwendet werden können AWS Transfer Family, und der Prozess der Erstellung einer Benutzerrolle wird beschrieben. Außerdem wird beschrieben, wie Sitzungsrichtlinien funktionieren, und es wird ein Beispiel für eine Benutzerrolle bereitgestellt.

AWS Transfer Family verwendet die folgenden Rollentypen:

- Benutzerrolle — Ermöglicht dienstverwalteten Benutzern den Zugriff auf die erforderlichen Transfer Family Family-Ressourcen. AWS Transfer Family nimmt diese Rolle im Kontext eines Transfer Family Family-Benutzer-ARN an.
- Zugriffsrolle — Ermöglicht den Zugriff nur auf die Amazon S3 S3-Dateien, die übertragen werden. Für eingehende AS2-Übertragungen verwendet die Zugriffsrolle den Amazon-Ressourcennamen (ARN) für die Vereinbarung. Für ausgehende AS2-Übertragungen verwendet die Zugriffsrolle den ARN für den Connector.

- Aufrufrolle — Zur Verwendung mit Amazon API Gateway als benutzerdefiniertem Identitätsanbieter des Servers. Transfer Family übernimmt diese Rolle im Kontext eines Transfer Family Family-Servers ARN.
- Rolle „Protokollierung“ — Wird verwendet, um Einträge bei Amazon zu protokollieren CloudWatch. Transfer Family verwendet diese Rolle, um Erfolgs- und Fehlschlagsdetails sowie Informationen zu Dateiübertragungen zu protokollieren. Transfer Family übernimmt diese Rolle im Kontext eines Transfer Family Family-Servers ARN. Für ausgehende AS2-Übertragungen verwendet die Protokollierungsrolle den Connector-ARN.
- Ausführungsrolle — Ermöglicht es einem Transfer Family Family-Benutzer, Workflows aufzurufen und zu starten. Transfer Family übernimmt diese Rolle im Zusammenhang mit einem Transfer Family Family-Workflow-ARN.

Zusätzlich zu diesen Rollen können Sie auch Sitzungsrichtlinien verwenden. Eine Sitzungsrichtlinie wird verwendet, um den Zugriff bei Bedarf einzuschränken. Beachten Sie, dass es sich bei diesen Richtlinien um eigenständige Richtlinien handelt, d. h., Sie fügen diese Richtlinien keiner Rolle hinzu. Stattdessen fügen Sie einem Transfer Family Family-Benutzer direkt eine Sitzungsrichtlinie hinzu.

> **Note**
>
> Wenn Sie einen vom Dienst verwalteten Transfer Family Family-Benutzer erstellen, können Sie die Option Richtlinie automatisch auf Basis des Basisordners generieren auswählen. Dies ist eine nützliche Tastenkombination, wenn Sie den Benutzerzugriff auf ihre eigenen Ordner einschränken möchten. Einzelheiten zu Sitzungsrichtlinien und ein Beispiel finden Sie auch unterSo funktionieren Sitzungsrichtlinien. Weitere Informationen zu Sitzungsrichtlinien finden Sie auch unter Sitzungsrichtlinien im IAM-Benutzerhandbuch.

Themen


- Eine Benutzerrolle erstellen
- So funktionieren Sitzungsrichtlinien
- Beispiel für eine Lese-/Schreibzugriffsrichtlinie


## Eine Benutzerrolle erstellen

Wenn Sie einen Benutzer erstellen, treffen Sie eine Reihe von Entscheidungen über den Benutzerzugriff. Zu diesen Entscheidungen gehört, auf welche Amazon S3 S3-Buckets oder Amazon

EFS-Dateisysteme der Benutzer zugreifen kann, auf welche Teile jedes Amazon S3 S3-Buckets und auf welche Dateien im Dateisystem zugegriffen werden kann und welche Berechtigungen der Benutzer hat (z. B. `PUT` oder`GET`).

Um den Zugriff festzulegen, erstellen Sie eine identitätsbasierte AWS Identity and Access Management (IAM) -Richtlinie und Rolle, die diese Zugriffsinformationen bereitstellen. Im Rahmen dieses Prozesses gewähren Sie Ihrem Benutzer Zugriff auf den Amazon S3-Bucket oder das Amazon EFS-Dateisystem, das das Ziel oder die Quelle für Dateioperationen ist. Die folgenden grundlegenden Schritte, die im Weiteren detailliert erläutert werden, skizzieren das Verfahren:

Eine Benutzerrolle erstellen

1. Erstellen Sie eine IAM-Richtlinie für AWS Transfer Family. Dies wird unter beschrieben. Um eine IAM-Richtlinie zu erstellen für AWS Transfer Family
2. Erstellen Sie eine IAM-Rolle und fügen Sie die neue IAM-Richtlinie an. Ein Beispiel finden Sie unter Beispiel für eine Lese-/Schreibzugriffsrichtlinie.
3. Stellen Sie eine Vertrauensbeziehung zwischen AWS Transfer Family und der IAM-Rolle her. Dies wird unter beschrieben. So stellen Sie eine Vertrauensbeziehung her

In den folgenden Verfahren wird beschrieben, wie Sie eine IAM-Richtlinie und -Rolle erstellen.

Um eine IAM-Richtlinie zu erstellen für AWS Transfer Family

1. Öffnen Sie die IAM-Konsole unter https://console.aws.amazon.com/iam/.
2. Wählen Sie im Navigationsbereich Policies (Richtlinien) und dann Create policy (Richtlinie erstellen).
3. Wählen Sie auf der Seite Create Policy (Richtlinie erstellen) die Registerkarte JSON aus.
4. Ersetzen Sie im daraufhin angezeigten Editor den Inhalt des Editors durch die IAM-Richtlinie, die Sie der IAM-Rolle zuordnen möchten.

   Sie können Lese-/Schreibzugriff gewähren oder Benutzern den Zugriff auf ihr Home-Verzeichnis einschränken. Weitere Informationen finden Sie unter Beispiel für eine Lese-/Schreibzugriffsrichtlinie.
5. Wählen Sie Richtlinie überprüfen aus, geben Sie einen Namen und eine Beschreibung für Ihre Richtlinie ein und wählen Sie dann Richtlinie erstellen aus.

Nun erstellen Sie eine IAM-Rolle und fügen an diese die neue IAM-Richtlinie an.

Um eine IAM-Rolle zu erstellen für AWS Transfer Family

1. Wählen Sie im Navigationsbereich Roles (Rollen) und dann Create role (Rolle erstellen).

   Vergewissern Sie sich, dass auf der Seite Rolle erstellen der AWS Dienst ausgewählt ist.
2. Wählen Sie in der Service-Liste Transfer (Übertragung) und dann Next: Permissions (Weiter: Berechtigungen) aus. Dadurch wird eine Vertrauensbeziehung zwischen AWS Transfer Family und hergestellt AWS.
3. Suchen Sie im Abschnitt Berechtigungsrichtlinien anhängen nach der Richtlinie, die Sie gerade erstellt haben, wählen Sie sie aus und klicken Sie dann auf Weiter: Tags.
4. (Optional) Geben Sie einen Schlüssel und einen Wert für ein Tag ein und wählen Sie Next: Review (Weiter: Prüfen) aus.
5. Geben Sie auf der Seite Review (Prüfen) einen Namen und eine Beschreibung für die neue Rolle ein und wählen Sie dann Create role (Rolle erstellen) aus.

Als Nächstes richten Sie eine Vertrauensbeziehung zwischen AWS Transfer Family und ein AWS.

So stellen Sie eine Vertrauensbeziehung her

> **Note**
>
> In unseren Beispielen verwenden wir `ArnLike` sowohl als auch`ArnEquals`. Sie sind funktionell identisch, weshalb Sie beide verwenden können, wenn Sie Ihre Richtlinien erstellen. Die Dokumentation Transfer Family verwendet`ArnLike`, wenn die Bedingung ein Platzhalterzeichen enthält, und `ArnEquals` um eine exakte Übereinstimmungsbedingung anzugeben.

1. Wählen Sie in der IAM-Konsole die von Ihnen eben erstellte Rolle aus.
2. Wählen Sie auf der Seite Summary (Übersicht) die Option Trust relationships (Vertrauensbeziehungen) und anschließend Edit trust relationship (Vertrauensbeziehung bearbeiten) aus.
3. Stellen Sie im Editor „Vertrauensverhältnis bearbeiten“ sicher, dass der Dienst aktiviert ist`"transfer.amazonaws.com"`. Die Zugriffsrichtlinie wird im Folgenden dargestellt.

```
{
  "Version": "2012-10-17",
```

```
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "transfer.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

Wir empfehlen Ihnen, die Tasten `aws:SourceAccount` und `aws:SourceArn` Condition zu verwenden, um sich vor dem Problem mit dem verwirrten Stellvertreter zu schützen. Das Quellkonto ist der Besitzer des Servers und der Quell-ARN ist der ARN des Benutzers. Beispielsweise:

```
"Condition": {
    "StringEquals": {
        "aws:SourceAccount": "account_id"
    },
    "ArnLike": {
        "aws:SourceArn": "arn:aws:transfer:region:account_id:user/*"
    }
}
```

Sie können die `ArnLike` Bedingung auch verwenden, wenn Sie die Einschränkung auf einen bestimmten Server statt auf einen beliebigen Server im Benutzerkonto vornehmen möchten. Beispielsweise:

```
"Condition": {
    "ArnLike": {
        "aws:SourceArn": "arn:aws:transfer:region:account-id:user/server-id/*"
    }
}
```

> **Note**
>
> Ersetzen Sie in den obigen Beispielen jeden *Platzhalter für Benutzereingaben* durch Ihre eigenen Informationen.

Einzelheiten zum Problem mit dem verwirrten Stellvertreter und weitere Beispiele finden Sie unterServiceübergreifende Confused-Deputy-Prävention.

4. Wählen Sie „Vertrauensrichtlinie aktualisieren“, um die Zugriffsrichtlinie zu aktualisieren.

Sie haben jetzt eine IAM-Rolle erstellt, mit der AWS Transfer Family Sie AWS Dienste in Ihrem Namen aufrufen können. Sie haben der Rolle die IAM-Richtlinie angehängt, die Sie erstellt haben, um Ihrem Benutzer Zugriff zu gewähren. In diesem Erste Schritte mit AWS Transfer Family Serverendpunkten Abschnitt werden diese Rolle und Richtlinie Ihrem oder Ihren Benutzern zugewiesen.

Informationen finden Sie auch unter:

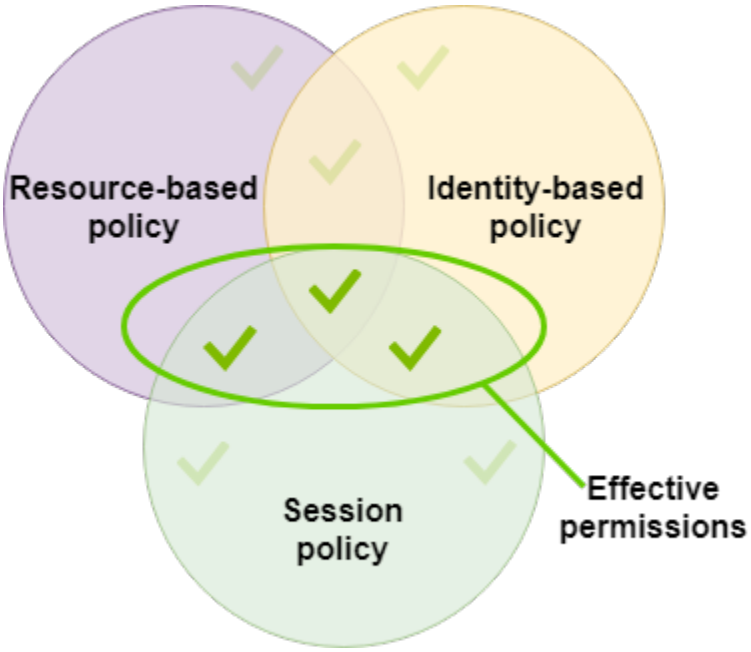
- Weitere allgemeine Informationen zu IAM-Rollen finden Sie im IAM-Benutzerhandbuch unter Erstellen einer Rolle zum Delegieren von Berechtigungen für einen AWS Dienst.
- Weitere Informationen zu identitätsbasierten Richtlinien für Amazon S3-Ressourcen finden Sie unter Identitäts- und Zugriffsmanagement in Amazon S3 im Amazon Simple Storage Service-Benutzerhandbuch.
- Weitere Informationen zu identitätsbasierten Richtlinien für Amazon EFS-Ressourcen finden Sie unter Verwenden von IAM zur Steuerung des Dateisystemdatenzugriffs im Amazon Elastic File System-Benutzerhandbuch.

## So funktionieren Sitzungsrichtlinien

Wenn ein Administrator eine Rolle erstellt, umfasst die Rolle häufig umfassende Berechtigungen, die mehrere Anwendungsfälle oder Teammitglieder abdecken. Wenn ein Administrator eine Konsolen-URL konfiguriert, kann er mithilfe einer Sitzungsrichtlinie die Berechtigungen für die resultierende Sitzung reduzieren. Wenn Sie beispielsweise eine Rolle mit Lese-/Schreibzugriff erstellen, können Sie eine URL einrichten, die den Zugriff der Benutzer nur auf ihre Home-Verzeichnisse beschränkt.

Sitzungsrichtlinien sind erweiterte Richtlinien, die Sie als Parameter übergeben, wenn Sie programmgesteuert eine temporäre Sitzung für eine Rolle oder einen Benutzer erstellen. Sitzungsrichtlinien sind nützlich, um Benutzer zu sperren, sodass sie nur Zugriff auf Bereiche Ihres Buckets haben, in denen Objektpräfixe ihren Benutzernamen enthalten. Das folgende Diagramm zeigt, dass die Berechtigungen der Sitzungsrichtlinie die Schnittmenge der Sitzungsrichtlinien

und der ressourcenbasierten Richtlinien sowie die Schnittmenge der Sitzungsrichtlinien und identitätsbasierten Richtlinien bilden.




Weitere Informationen finden Sie unter Sitzungsrichtlinien im IAM-Benutzerhandbuch.

In AWS Transfer Family, eine Sitzungsrichtlinie wird nur unterstützt, wenn Sie zu oder von Amazon S3 übertragen. Die folgende Beispielrichtlinie ist eine Sitzungsrichtlinie, die den Zugriff von Benutzern nur auf ihre home Verzeichnisse beschränkt. Beachten Sie Folgendes:

- Die `PutObjectACL` Anweisungen `GetObjectACL` und sind nur erforderlich, wenn Sie den kontenübergreifenden Zugriff aktivieren müssen. Das heißt, Ihr Transfer Family Family-Server muss auf einen Bucket in einem anderen Konto zugreifen.
- Die maximale Länge einer Sitzungsrichtlinie beträgt 2048 Zeichen. Weitere Informationen finden Sie unter dem Anforderungsparameter Policy für die `CreateUser` Aktion in der API-Referenz.
- Wenn Ihr Amazon S3 S3-Bucket mit AWS Key Management Service (AWS KMS) verschlüsselt ist, müssen Sie in Ihrer Richtlinie zusätzliche Berechtigungen angeben. Details hierzu finden Sie unter Datenverschlüsselung in Amazon S3.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowListingOfUserFolder",
            "Action": [
```

```
                "s3:ListBucket"
            ],
            "Effect": "Allow",
            "Resource": [
                "arn:aws:s3:::${transfer:HomeBucket}"
            ],
            "Condition": {
                "StringLike": {
                    "s3:prefix": [
                        "${transfer:HomeFolder}/*",
                        "${transfer:HomeFolder}"
                    ]
                }
            }
        },
        {
            "Sid": "HomeDirObjectAccess",
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
                "s3:DeleteObject",
                "s3:DeleteObjectVersion",
                "s3:GetObjectVersion",
                "s3:GetObjectACL",
                "s3:PutObjectACL"
            ],
            "Resource": "arn:aws:s3:::${transfer:HomeDirectory}/*"
        }
    ]
}
```

> **Note**
>
> Im obigen Richtlinienbeispiel wird davon ausgegangen, dass die Home-Verzeichnisse der Benutzer so eingestellt sind, dass sie einen abschließenden Schrägstrich enthalten, um anzuzeigen, dass es sich um ein Verzeichnis handelt. Wenn Sie dagegen das eines Benutzers `HomeDirectory` ohne den abschließenden Schrägstrich angeben, sollten Sie es in Ihre Richtlinie aufnehmen.

Beachten Sie in der vorherigen Beispielrichtlinie die Verwendung der `transfer:HomeFolder` `transfer:HomeDirectory` Richtlinienparametertransfer:HomeBucket, und. Diese Parameter werden für den festgelegtHomeDirectory, der für den Benutzer konfiguriert ist, wie unter HomeDirectoryund beschriebenImplementierung Ihrer API-Gateway-Methode. Diese Parameter haben die folgenden Definitionen:

- Der `transfer:HomeBucket` Parameter wird durch die erste Komponente von ersetztHomeDirectory.
- Der `transfer:HomeFolder` Parameter wird durch die verbleibenden Teile des `HomeDirectory` Parameters ersetzt.
- Für den `transfer:HomeDirectory` Parameter wurde der führende Schrägstrich (/) entfernt, sodass er als Teil eines S3-Amazon-Ressourcennamens (ARN) in einer `Resource` Anweisung verwendet werden kann.

> **Note**
>
> Wenn Sie logische Verzeichnisse verwenden, also die Verzeichnisse des homeDirectoryType Benutzers, werden `LOGICAL` diese Richtlinienparameter (HomeBucketHomeDirectory, undHomeFolder) nicht unterstützt.

Nehmen wir beispielsweise an, dass der `HomeDirectory` Parameter, der für den Transfer Family Family-Benutzer konfiguriert ist`/home/bob/amazon/stuff/`,

- `transfer:HomeBucket`ist auf eingestellt`/home`.
- `transfer:HomeFolder`ist auf gesetzt`/bob/amazon/stuff/`.
- `transfer:HomeDirectory`wird`home/bob/amazon/stuff/`.

Die erste `"Sid"` ermöglicht es dem Benutzer, alle Verzeichnisse aufzulisten, beginnend mit`/home/bob/amazon/stuff/`.

Die zweite `"Sid"` schränkt den `get` Zugriff des Benutzers `put` auf denselben Pfad ein,`/home/bob/amazon/stuff/`.

## Beispiel für eine Lese-/Schreibzugriffsrichtlinie

Lese-/Schreibzugriff auf den Amazon S3 S3-Bucket gewähren

Die folgende Beispielrichtlinie für AWS Transfer Family gewährt Lese-/Schreibzugriff auf Objekte in Ihrem Amazon S3 S3-Bucket.

Beachten Sie Folgendes:

- Ersetzen Sie DOC-EXAMPLE-BUCKET durch den Namen Ihres Amazon-S3-Buckets.
- Die `PutObjectACL` Anweisungen `GetObjectACL` und sind nur erforderlich, wenn Sie den kontoübergreifenden Zugriff aktivieren müssen. Das heißt, Ihr Transfer Family Family-Server muss auf einen Bucket in einem anderen Konto zugreifen.
- Die `DeleteObjectVersion` Anweisungen `GetObjectVersion` und sind nur erforderlich, wenn die Versionierung für den Amazon S3 S3-Bucket aktiviert ist, auf den zugegriffen wird.

> **Note**
>
> Wenn Sie jemals die Versionierung für Ihren Bucket aktiviert haben, benötigen Sie diese Berechtigungen, da Sie die Versionierung in Amazon S3 nur aussetzen und nicht vollständig ausschalten können. Weitere Informationen finden Sie unter Buckets ohne Version, mit aktivierter Versionsverwaltung und Sperrung der Versionierung.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowListingOfUserFolder",
            "Action": [
                "s3:ListBucket",
                "s3:GetBucketLocation"
            ],
            "Effect": "Allow",
            "Resource": [
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET"
            ]
        },
        {
            "Sid": "HomeDirObjectAccess",
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
```

```
                "s3:DeleteObject",
                "s3:DeleteObjectVersion",
                "s3:GetObjectVersion",
                "s3:GetObjectACL",
                "s3:PutObjectACL"
            ],
            "Resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
        }
    ]
}
```

Dateisystemzugriff auf Dateien im Amazon EFS-Dateisystem gewähren

> **Note**
>
> Zusätzlich zur Richtlinie müssen Sie auch sicherstellen, dass Ihre POSIX-Dateiberechtigungen den entsprechenden Zugriff gewähren. Weitere Informationen finden Sie unter Arbeiten mit Benutzern, Gruppen und Berechtigungen auf Netzwerkdateisystem (NFS)-Ebene im Benutzerhandbuch für Amazon Elastic File System.

Die folgende Beispielrichtlinie gewährt Root-Dateisystemzugriff auf Dateien in Ihrem Amazon EFS-Dateisystem.

> **Note**
>
> Ersetzen Sie in den folgenden Beispielen *region* durch Ihre Region, *account-id* durch das Konto, in dem sich die Datei befindet, und *file-system-id*durch die ID Ihres Amazon Elastic File System (Amazon EFS).

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "RootFileSystemAccess",
            "Effect": "Allow",
            "Action": [
                "elasticfilesystem:ClientRootAccess",
```

```
                "elasticfilesystem:ClientMount",
                "elasticfilesystem:ClientWrite"
            ],
            "Resource": "arn:aws:elasticfilesystem:region:account-id:file-system/file-
system-id"
        }
    ]
}
```

Die folgende Beispielrichtlinie gewährt Benutzerdateisystemzugriff auf Dateien in Ihrem Amazon EFS-Dateisystem.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "UserFileSystemAccess",
            "Effect": "Allow",
            "Action": [
                "elasticfilesystem:ClientMount",
                "elasticfilesystem:ClientWrite"
            ],
            "Resource": "arn:aws:elasticfilesystem:region:account-id:file-system/file-
system-id"
        }
    ]
}
```

# Tutorials zur Transfer Family

Das AWS Transfer Family Benutzerhandbuch bietet detaillierte Anleitungen für verschiedene Anwendungsfälle.

- Erste Schritte mit AWS Transfer Family Serverendpunkten: Dieses Tutorial führt Sie durch die Erstellung eines SFTP Transfer Family Family-Servers und eines dienstverwalteten Benutzers und zeigt dann, wie Sie eine Datei mithilfe eines Clients übertragen.
- Einrichtung und Verwendung von SFTP-Anschlüssen: Dieses Tutorial zeigt, wie Sie einen SFTP-Connector einrichten und anschließend Dateien zwischen Amazon S3 S3-Speicher und einem SFTP-Server übertragen.
- Einrichtung einer Amazon API Gateway Gateway-Methode als benutzerdefinierter Identitätsanbieter : Dieses Tutorial zeigt, wie Sie eine Amazon API Gateway Gateway-Methode einrichten und sie als benutzerdefinierten Identitätsanbieter verwenden, um Dateien auf einen AWS Transfer Family Server hochzuladen.
- Einen verwalteten Workflow zum Entschlüsseln einer Datei einrichten: Dieses Tutorial zeigt, wie Sie einen verwalteten Workflow einrichten, der einen Entschlüsselungsschritt enthält, und wie Sie eine verschlüsselte Datei in einen Amazon S3 S3-Bucket hochladen und dann die entschlüsselte Datei anzeigen.
- Einrichtung einer AS2-Konfiguration: In diesem Tutorial werden die Schritte beschrieben, die zur Konfiguration eines AS2 Transfer Family Family-Servers erforderlich sind. Es gibt Anweisungen zum Importieren von Zertifikaten, zum Erstellen von Profilen und Vereinbarungen, zum optionalen Erstellen eines AS2-Connectors und zum anschließenden Testen der Konfiguration.

Themen


- Erste Schritte mit AWS Transfer Family Serverendpunkten
- Einen verwalteten Workflow zum Entschlüsseln einer Datei einrichten
- Einrichtung und Verwendung von SFTP-Anschlüssen
- Einrichtung einer Amazon API Gateway Gateway-Methode als benutzerdefinierter Identitätsanbieter
- Einrichtung einer AS2-Konfiguration

# Erste Schritte mit AWS Transfer Family Serverendpunkten

Verwenden Sie dieses Tutorial, um mit AWS Transfer Family (Transfer Family) zu beginnen. Sie erfahren, wie Sie mithilfe von Amazon S3-Speicher einen SFTP-fähigen Server mit öffentlich zugänglichem Endpunkt erstellen, einen Benutzer mit service-verwalteter Authentifizierung hinzufügen und eine Datei mit Cyberduck übertragen.

Themen


- Voraussetzungen
- Schritt 1: Melden Sie sich bei der AWS Transfer Family -Konsole an
- Schritt 2: Erstellen Sie einen SFTP-fähigen Server
- Schritt 3: Fügen Sie einen vom Service verwalteten Benutzer hinzu
- Schritt 4: Eine Datei mit einem Client übertragen


## Voraussetzungen

Bevor Sie beginnen, stellen Sie sicher, dass Sie die Anforderungen unter erfüllen. Voraussetzungen Im Rahmen dieser Einrichtung erstellen Sie einen Amazon Simple Storage Service (Amazon S3) -Bucket und eine AWS Identity and Access Management (IAM) -Benutzerrolle.

Es sind Berechtigungen für die Verwendung der AWS Transfer Family Konsole erforderlich, und es sind Berechtigungen für die Konfiguration anderer AWS Dienste erforderlich, die Transfer Family verwendet, wie Amazon Simple Storage Service AWS Certificate Manager, Amazon Elastic File System und Amazon Route 53. Beispielsweise FullAccess gewährt AmazonS3 Benutzern, die AWS mit Transfer Family Dateien in und aus Transfer Family übertragen, Berechtigungen zum Einrichten und Verwenden eines Amazon S3 S3-Buckets. Einige der Berechtigungen in dieser Richtlinie sind erforderlich, um Amazon S3 S3-Buckets zu erstellen.

Um die Transfer Family Family-Konsole verwenden zu können, benötigen Sie Folgendes:

- AWSTransferConsoleFullAccessgewährt Ihrem SFTP-Benutzer Berechtigungen zum Erstellen von Transfer Family Family-Ressourcen.
- IAM FullAccess (oder speziell eine Richtlinie, die die Erstellung von IAM-Rollen ermöglicht) ist nur erforderlich, wenn Sie möchten, dass Transfer Family automatisch eine Protokollierungsrolle für Ihren Server in Amazon CloudWatch Logs oder eine Benutzerrolle für einen Benutzer, der sich bei einem Server anmeldet, erstellt.

- Um VPC-Servertypen zu erstellen und zu löschen, müssen Sie Ihrer Richtlinie die Aktionen ec2: CreateVpc Endpoint und ec2: DeleteVpc Endpoints hinzufügen.

> **Note**
>
> Die AmazonS3- FullAccess und FullAccessIAM-Richtlinien selbst werden für die allgemeine Verwendung von nicht benötigt. AWS Transfer Family Sie werden hier vorgestellt, um auf einfache Weise sicherzustellen, dass alle von Ihnen benötigten Berechtigungen abgedeckt sind. Darüber hinaus handelt AWS es sich um verwaltete Richtlinien, bei denen es sich um Standardrichtlinien handelt, die allen AWS Kunden zur Verfügung stehen. Sie können die einzelnen Berechtigungen in diesen Richtlinien einsehen und festlegen, welche Mindestberechtigungen Sie für Ihre Zwecke benötigen.

## Schritt 1: Melden Sie sich bei der AWS Transfer Family -Konsole an

Um sich bei Transfer Family anzumelden

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Geben Sie als Konto-ID oder Alias die ID für Ihr Konto ein AWS-Konto.
3. Geben Sie als IAM-Benutzername den Namen der Benutzerrolle ein, die Sie für Transfer Family erstellt haben.
4. Geben Sie unter Passwort Ihr AWS Kontopasswort ein.
5. Klicken Sie auf Sign in.

## Schritt 2: Erstellen Sie einen SFTP-fähigen Server

Das Secure Shell (SSH) File Transfer Protocol (SFTP) ist ein Netzwerkprotokoll, das für die sichere Übertragung von Daten über das Internet verwendet wird. Das Protokoll unterstützt die volle Sicherheits- und Authentifizierungsfunktionalität von SSH. Es wird häufig für den Austausch von Daten, einschließlich sensibler Informationen, zwischen Geschäftspartnern in einer Vielzahl von Branchen wie Finanzdienstleistungen, Gesundheitswesen, Einzelhandel und Werbung verwendet.

Um einen SFTP-fähigen Server zu erstellen

1. Wählen Sie im Navigationsbereich Server und dann Server erstellen aus.
2. Wählen Sie unter Protokolle auswählen die Option SFTP und dann Weiter aus.
3. Wählen Sie unter Wählen Sie einen Identitätsanbieter die Option Service managed to storage user identities and keys in Transfer Family aus, und klicken Sie dann auf Weiter.
4. Gehen Sie unter Endpunkt auswählen wie folgt vor:
   a. Wählen Sie als Endpunkttyp den Typ Öffentlich zugänglicher Endpunkt aus.
   b. Wählen Sie für Benutzerdefinierter Hostname die Option Keine aus.
   c. Wählen Sie Weiter aus.
5. Wählen Sie unter Domain auswählen die Option Amazon S3 aus.
6. Wählen Sie unter Zusätzliche Details konfigurieren unter Optionen für kryptografische Algorithmen eine Sicherheitsrichtlinie aus, die die kryptografischen Algorithmen enthält, die für die Verwendung durch Ihren Server aktiviert sind. Unsere neueste Sicherheitsrichtlinie ist die Standardeinstellung: Einzelheiten finden Sie unter. Sicherheitsrichtlinien für AWS Transfer Family Server

   > **Note**
   >
   > Nur wenn Sie einen verwalteten Workflow für Ihren Server hinzufügen, wählen Sie Neue Rolle für die CloudWatchProtokollierung erstellen. Um Serverereignisse zu protokollieren, müssen Sie keine IAM-Rolle erstellen.

7. Wählen Sie unter Überprüfen und erstellen die Option Server erstellen aus. Sie werden zur Seite Server weitergeleitet.

Es kann einige Minuten dauern, bis sich der Status Ihres neuen Servers auf Online ändert. Zu diesem Zeitpunkt kann Ihr Server Dateioperationen ausführen, aber Sie müssen zuerst einen Benutzer erstellen. Einzelheiten zum Erstellen von Benutzern finden Sie unterVerwalten von Benutzern für Serverendpunkte.

# Schritt 3: Fügen Sie einen vom Service verwalteten Benutzer hinzu

Um einen Benutzer zum SFTP-fähigen Server hinzuzufügen

1. Wählen Sie auf der Seite Server den Server aus, zu dem Sie einen Benutzer hinzufügen möchten.
2. Wählen Sie Benutzer hinzufügen.
3. Geben Sie im Abschnitt Benutzerkonfiguration unter Benutzername den Benutzernamen ein. Dieser Benutzername muss mindestens 3 und maximal 100 Zeichen lang sein. Sie können die folgenden Zeichen im Benutzernamen verwenden: a—z, A-Z, 0—9, Unterstrich '_', Bindestrich '-', Punkt ' . 'und beim Zeichen (@). Der Benutzername darf nicht mit einem Bindestrich, Punkt oder einem AT-Zeichen beginnen.
4. Wählen Sie für Access die IAM-Rolle aus, die Sie in erstellt haben. Erstellen Sie eine IAM-Rolle und -Richtlinie Diese IAM-Rolle umfasst eine IAM-Richtlinie, die Berechtigungen für den Zugriff auf Ihren Amazon S3 S3-Bucket sowie eine Vertrauensbeziehung mit dem AWS Transfer Family Service enthält. Das unter beschriebene Verfahren So stellen Sie eine Vertrauensbeziehung her zeigt, wie Sie die richtige Vertrauensbeziehung aufbauen können.
5. Wählen Sie für Richtlinie die Option Keine aus.
6. Wählen Sie für das Home-Verzeichnis den Amazon S3 S3-Bucket aus, in dem Sie die Daten speichern möchten, mit denen Sie die Daten übertragen AWS Transfer Family. Geben Sie den Pfad zum home Verzeichnis ein. Dies ist das Verzeichnis, das Ihren Benutzern angezeigt wird, wenn sie sich mit ihrem Client anmelden.

   Wir empfehlen, einen Verzeichnispfad zu verwenden, der den Benutzernamen enthält, sodass Sie die Möglichkeit haben, eine Sitzungsrichtlinie zu verwenden. Eine Sitzungsrichtlinie beschränkt den Zugriff eines Benutzers im Amazon S3 S3-Bucket auf das home Verzeichnis dieses Benutzers. Weitere Informationen zur Verwendung von Sitzungsrichtlinien finden Sie unterSo funktionieren Sitzungsrichtlinien.

   Wenn Sie möchten, können Sie diesen Parameter leer lassen, um das `root` Verzeichnis Ihres Amazon S3 S3-Buckets zu verwenden. Wenn Sie diese Option wählen, stellen Sie sicher, dass Ihre IAM-Rolle Zugriff auf das `root` Verzeichnis bietet.
7. Aktivieren Sie das Kontrollkästchen Eingeschränkt, um zu verhindern, dass Ihre Benutzer auf Inhalte außerhalb ihres home Verzeichnisses zugreifen. Dadurch wird auch verhindert, dass Benutzer den Amazon S3 S3-Bucket- oder Ordnernamen sehen.

8. Geben Sie für den öffentlichen SSH-Schlüssel den öffentlichen SSH-Schlüsselteil des SSH-Schlüsselpaars im Format ein. `ssh-rsa <string>`

   Ihr Schlüssel muss vom Dienst validiert werden, bevor Sie Ihren neuen Benutzer hinzufügen können. Weitere Hinweise zum Generieren eines SSH-Schlüsselpaars finden Sie unterGenerieren Sie SSH-Schlüssel für vom Service verwaltete Benutzer.
9. (Optional) Geben Sie für Schlüssel und Wert ein oder mehrere Tags als Schlüssel-Wert-Paare ein und wählen Sie Tag hinzufügen aus.
10. Wählen Sie Add (Hinzufügen), um den neuen Benutzer dem ausgewählten Server hinzuzufügen.

    Der neue Benutzer wird auf der Seite mit den Serverdetails im Bereich Benutzer angezeigt.

## Schritt 4: Eine Datei mit einem Client übertragen

Sie übertragen Dateien über den AWS Transfer Family Service, indem Sie den Übertragungsvorgang in einem Client angeben. AWS Transfer Family unterstützt mehrere Clients. Details hierzu finden Sie unter Übertragung von Dateien über einen Serverendpunkt mit einem Client

Dieser Abschnitt enthält Verfahren zur Verwendung von Cyberduck und OpenSSH.

Themen

- Verwenden Sie Cyberduck
- OpenSSH verwenden

### Verwenden Sie Cyberduck

Um Dateien AWS Transfer Family mit Cyberduck zu übertragen

1. Öffnen Sie den Cyberduck-Client.
2. Wählen Sie Verbindung öffnen.
3. Wählen Sie im Dialogfeld „Verbindung öffnen“ die Option SFTP (SSH File Transfer Protocol).
4. Geben Sie für Server Ihren Serverendpunkt ein. Der Serverendpunkt befindet sich auf der Seite mit den Serverdetails, siheSFTP-, FTPS- und FTP-Serverdetails anzeigen.
5. Geben Sie als Portnummer den Wert **22** für SFTP ein.
6. Geben Sie in das Feld Username (Benutzername) den Namen des Benutzers ein, den Sie in Verwalten von Benutzern für Serverendpunkte erstellt haben.

7. Wählen Sie für SSH Private Key den privaten SSH-Schlüssel aus oder geben Sie ihn ein.
8. Wählen Sie Connect aus.
9. Führen Sie Ihre Dateiübertragung durch.

   Führen Sie nun – abhängig von der Position der Dateien – einen der folgenden Schritte durch:

   - Wählen Sie in Ihrem lokalen Verzeichnis (der Quelle) die Dateien aus, die Sie übertragen möchten, und ziehen Sie sie per Drag & Drop in das Amazon S3 S3-Verzeichnis (das Ziel).
   - Wählen Sie im Amazon S3 S3-Verzeichnis (der Quelle) die Dateien aus, die Sie übertragen möchten, und ziehen Sie sie per Drag & Drop in Ihr lokales Verzeichnis (das Ziel).

## OpenSSH verwenden

Unten wird beschrieben, wie Dateien in der Befehlszeile mit OpenSSH übertragen werden.

> **Note**
>
> Dieser Client funktioniert nur mit einem SFTP-fähigen Server.

Um Dateien AWS Transfer Family mit dem OpenSSH-Befehlszeilenprogramm zu übertragen

1. Öffnen Sie unter Linux oder Macintosh ein Befehls-Terminal.
2. Geben Sie an der Eingabeaufforderung den folgenden Befehl ein: `% sftp -i transfer-key sftp_user@service_endpoint`

   Im vorherigen Befehl `sftp_user` ist dies der Benutzername und `transfer-key` der private SSH-Schlüssel. Hier `service_endpoint` ist der Endpunkt des Servers, wie er in der AWS Transfer Family Konsole für den ausgewählten Server angezeigt wird.

   Eine `sftp`-Eingabeaufforderung sollte angezeigt werden.
3. (Optional) Um das Home-Verzeichnis des Benutzers anzuzeigen, geben Sie an der `sftp` Eingabeaufforderung den folgenden Befehl ein: `sftp> pwd`
4. Geben Sie den folgenden Text in die nächste Zeile ein: `sftp> cd /mybucket/home/sftp_user`

   In dieser Übung für die ersten Schritte ist dieser Amazon S3 S3-Bucket das Ziel der Dateiübertragung.

5. Geben Sie den folgenden Befehl in die nächste Zeile ein: `sftp> put filename.txt`

   Der `put` Befehl überträgt die Datei in den Amazon S3 S3-Bucket.

   Eine Meldung wie die Folgende wird angezeigt und gibt an, dass die Dateiübertragung läuft oder abgeschlossen wurde.

   ```
   Uploading filename.txt to /my-bucket/home/sftp_user/filename.txt

   some-file.txt 100% 127 0.1KB/s 00:00
   ```

# Einen verwalteten Workflow zum Entschlüsseln einer Datei einrichten

In diesem Tutorial wird veranschaulicht, wie Sie einen verwalteten Workflow einrichten, der einen Entschlüsselungsschritt enthält. Das Tutorial zeigt auch, wie Sie eine verschlüsselte Datei in einen Amazon S3 S3-Bucket hochladen und dann die entschlüsselte Datei in demselben Bucket anzeigen.

> **Note**
>
> Der AWS Speicher-Blog enthält einen Beitrag, in dem beschrieben wird, wie Dateien mithilfe von Transfer Family Managed Workflows, Verschlüsseln und Entschlüsseln von Dateien mit PGP und einfach entschlüsselt werden können, ohne Code zu schreiben, beschrieben werden. AWS Transfer Family

Themen


- Schritt 1: Konfigurieren Sie eine Ausführungsrolle
- Schritt 2: Erstellen Sie einen verwalteten Workflow
- Schritt 3: Fügen Sie den Workflow einem Server hinzu und erstellen Sie einen Benutzer
- Schritt 4: Erstellen Sie ein PGP-Schlüsselpaar
- Schritt 5: Speichern Sie den privaten PGP-Schlüssel in AWS Secrets Manager
- Schritt 6: Verschlüsseln Sie eine Datei
- Schritt 7: Führen Sie den Workflow aus und sehen Sie sich die Ergebnisse an

## Schritt 1: Konfigurieren Sie eine Ausführungsrolle

Erstellen Sie eine AWS Identity and Access Management (IAM-) Ausführungsrolle, mit der Transfer Family einen Workflow starten kann. Der Prozess der Erstellung einer Ausführungsrolle wird unter beschriebenIAM-Richtlinien für Workflows.

> **Note**
>
> Stellen Sie beim Erstellen einer Ausführungsrolle sicher, dass eine Vertrauensbeziehung zwischen der Ausführungsrolle und der Transfer Family hergestellt wird, wie unter beschriebenSo stellen Sie eine Vertrauensbeziehung her.

Die folgende Richtlinie für Ausführungsrollen enthält alle erforderlichen Berechtigungen, um den Workflow zu starten, den Sie in diesem Tutorial erstellen. Wenn Sie diese Beispielrichtlinie verwenden möchten, ersetzen Sie *`user input placeholders`* durch Ihre eigenen Informationen. `DOC-EXAMPLE-BUCKET`Ersetzen Sie es durch den Namen des Amazon S3 S3-Buckets, in den Sie Ihre verschlüsselten Dateien hochladen.

> **Note**
>
> Nicht jeder Workflow benötigt alle Berechtigungen, die in diesem Beispiel aufgeführt sind. Sie können die Berechtigungen je nach Art der Schritte in Ihrem spezifischen Workflow einschränken. Die für jeden vordefinierten Schritttyp erforderlichen Berechtigungen werden unter beschriebenVerwenden Sie vordefinierte Schritte. Die für einen benutzerdefinierten Schritt erforderlichen Berechtigungen werden unter beschriebenIAM-Berechtigungen für einen benutzerdefinierten Schritt.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "WorkflowsS3Permissions",
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:GetObjectTagging",
                "s3:GetObjectVersion",
```

```
                "s3:PutObject",
                "s3:PutObjectTagging",
                "s3:ListBucket",
                "s3:PutObjectTagging",
                "s3:PutObjectVersionTagging",
                "s3:DeleteObjectVersion",
                "s3:DeleteObject"
            ],
            "Resource": ["arn:aws:s3:::DOC-EXAMPLE-BUCKET/*",
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET"]
            "Condition": {
                "StringEquals": {
                    "s3:RequestObjectTag/Archive": "yes"
                }
            }
        },
        {
            "Sid": "DecryptSecret",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue"
            ],
            "Resource": "arn:aws:secretsmanager:region:account-id:secret:aws/transfer/
*"
        }
    ]
}
```

## Schritt 2: Erstellen Sie einen verwalteten Workflow

Jetzt müssen Sie einen Workflow erstellen, der einen Entschlüsselungsschritt enthält.

Um einen Workflow zu erstellen, der einen Entschlüsselungsschritt enthält

1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Wählen Sie im linken Navigationsbereich Workflows und dann Workflow erstellen aus.
3. Geben Sie die folgenden Details ein:
   - Geben Sie beispielsweise eine Beschreibung ein**`Decrypt workflow example`**.
   - Wählen Sie im Abschnitt Nominale Schritte die Option Schritt hinzufügen aus.
4. Wählen Sie unter Schritttyp auswählen die Option Datei entschlüsseln und dann Weiter aus.

5. Geben Sie im Dialogfeld „Parameter konfigurieren“ Folgendes an:

   - Geben Sie einen aussagekräftigen Schrittnamen ein, **`decrypt-step`** z. B. Leerzeichen sind in Schrittnamen nicht zulässig.
   - Wählen Sie als Ziel für entschlüsselte Dateien Amazon S3.
   - Wählen Sie für den Namen des Ziel-Buckets denselben Amazon S3 S3-Bucket aus, den Sie `DOC-EXAMPLE-BUCKET` in der IAM-Richtlinie, die Sie in Schritt 1 erstellt haben, angegeben haben.
   - Geben Sie für das Zielschlüsselpräfix den Namen des Präfixes (Ordners) ein, in dem Sie Ihre entschlüsselten Dateien in Ihrem Ziel-Bucket speichern möchten, z. B. **`decrypted-files/`**

     > **Note**
     >
     > Achten Sie darauf, Ihrem Präfix einen abschließenden Schrägstrich (**`/`**) hinzuzufügen.

   - Lassen Sie für dieses Tutorial die Option Existierendes überschreiben deaktiviert. Wenn diese Einstellung deaktiviert ist und Sie versuchen, eine Datei mit dem identischen Namen einer vorhandenen Datei zu entschlüsseln, wird die Workflow-Verarbeitung gestoppt und die neue Datei wird nicht verarbeitet.

   Wählen Sie Weiter, um zum Überprüfungsbildschirm zu gelangen.
6. Überprüfen Sie die Details für den Schritt. Wenn alles korrekt ist, wählen Sie Schritt erstellen.
7. Ihr Workflow benötigt nur den einzigen Entschlüsselungsschritt, sodass keine zusätzlichen Schritte zur Konfiguration erforderlich sind. Wählen Sie Workflow erstellen, um den neuen Workflow zu erstellen.

Notieren Sie sich die Workflow-ID für Ihren neuen Workflow. Sie benötigen diese ID für den nächsten Schritt. In diesem Tutorial wird die Workflow-ID *`w-1234abcd5678efghi`* als Beispiel verwendet.

## Schritt 3: Fügen Sie den Workflow einem Server hinzu und erstellen Sie einen Benutzer

Da Sie nun über einen Workflow mit einem Entschlüsselungsschritt verfügen, müssen Sie ihn einem Transfer Family Family-Server zuordnen. Dieses Tutorial zeigt, wie Sie den Workflow an einen vorhandenen Transfer Family Family-Server anhängen. Alternativ können Sie einen neuen Server für Ihren Workflow erstellen.

Nachdem Sie den Workflow an einen Server angehängt haben, müssen Sie einen Benutzer erstellen, der per SFTP auf den Server zugreifen und die Ausführung des Workflows auslösen kann.

So konfigurieren Sie einen Transfer Family Family-Server für die Ausführung eines Workflows

1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Wählen Sie im linken Navigationsbereich Server und dann einen Server aus der Liste aus. Stellen Sie sicher, dass dieser Server das SFTP-Protokoll unterstützt.
3. Scrollen Sie auf der Detailseite für den Server nach unten zum Abschnitt Zusätzliche Details und wählen Sie dann Bearbeiten aus.
4. Wählen Sie auf der Seite Zusätzliche Details bearbeiten im Abschnitt Verwaltete Workflows Ihren Workflow und wählen Sie eine entsprechende Ausführungsrolle aus.
   - Wählen Sie unter Workflow für vollständige Datei-Uploads den Workflow aus, den Sie erstellt habenSchritt 2: Erstellen Sie einen verwalteten Workflow, **`w-1234abcd5678efghi`** z. B. in.
   - Wählen Sie für die Ausführungsrolle für verwaltete Workflows die IAM-Rolle aus, in der Sie sie erstellt haben. Schritt 1: Konfigurieren Sie eine Ausführungsrolle
5. Scrollen Sie zum Ende der Seite und wählen Sie Speichern, um Ihre Änderungen zu speichern.

Notieren Sie sich die ID des Servers, den Sie verwenden. Der Name des AWS Secrets Manager Geheimnisses, das Sie zum Speichern Ihrer PGP-Schlüssel verwenden, basiert teilweise auf der Server-ID.

Um einen Benutzer hinzuzufügen, der den Workflow auslösen kann

1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Wählen Sie im linken Navigationsbereich Server und dann den Server aus, den Sie für den Entschlüsselungsworkflow verwenden.
3. Scrollen Sie auf der Seite mit den Serverdetails nach unten zum Abschnitt Benutzer und wählen Sie Benutzer hinzufügen aus.
4. Geben Sie für Ihren neuen Benutzer die folgenden Details ein:
   - Geben Sie für Username (Benutzername) **`decrypt-user`** ein.
   - Wählen Sie unter Rolle eine Benutzerrolle aus, die auf Ihren Server zugreifen kann.
   - Wählen Sie für Home-Verzeichnis den Amazon S3 S3-Bucket aus, den Sie zuvor verwendet haben, zum Beispiel`DOC-EXAMPLE-BUCKET`.

- Fügen Sie für öffentliche SSH-Schlüssel einen öffentlichen Schlüssel ein, der einem privaten Schlüssel entspricht, den Sie haben. Details hierzu finden Sie unter Generieren Sie SSH-Schlüssel für vom Service verwaltete Benutzer.

5. Wählen Sie Hinzufügen, um Ihren neuen Benutzer zu speichern.

Notieren Sie sich den Namen Ihres Transfer Family Family-Benutzers für diesen Server. Das Geheimnis basiert teilweise auf dem Namen des Benutzers. Der Einfachheit halber verwendet dieses Tutorial ein Standardgeheimnis, das von jedem Benutzer des Servers verwendet werden kann.

## Schritt 4: Erstellen Sie ein PGP-Schlüsselpaar

Verwenden Sie einen der unterstützten PGP-Clients, um ein PGP-Schlüsselpaar zu generieren. Dieser Vorgang wird ausführlich unter beschrieben. Generieren Sie PGP-Schlüssel

Um ein PGP-Schlüsselpaar zu generieren

1. Für dieses Tutorial können Sie den Client gpg (GnuPG) Version 2.0.22 verwenden, um ein PGP-Schlüsselpaar zu generieren, das RSA als Verschlüsselungsalgorithmus verwendet. Führen Sie für diesen Client den folgenden Befehl aus und geben Sie eine E-Mail-Adresse und eine Passphrase ein. Sie können einen beliebigen Namen oder eine E-Mail-Adresse verwenden. Stellen Sie sicher, dass Sie sich die von Ihnen verwendeten Werte merken, da Sie sie später im Tutorial eingeben müssen.

   ```
   gpg --gen-key
   ```

   > **Note**
   >
   > Wenn Sie GnuPG Version 2.3.0 oder neuer verwenden, müssen Sie ausführengpg `--full-gen-key`. Wenn Sie nach dem Typ des zu erstellenden Schlüssels gefragt werden, wählen Sie RSA oder ECC. Wenn Sie jedoch ECC wählen, stellen Sie sicher, dass Sie BrainPool für die elliptische Kurve entweder NIST oder wählen. Wählen Sie nicht. Curve 25519

2. Exportieren Sie den privaten Schlüssel, indem Sie den folgenden Befehl ausführen. *`user@example.com`*Ersetzen Sie ihn durch die E-Mail-Adresse, die Sie bei der Generierung des Schlüssels verwendet haben.

```
gpg --output workflow-tutorial-key.pgp --armor --export-secret-key user@example.com
```

Dieser Befehl exportiert den privaten Schlüssel in die **`workflow-tutorial-key.pgp`** Datei. Sie können der Ausgabedatei einen beliebigen Namen geben. Sie können die Datei mit dem privaten Schlüssel auch löschen, nachdem Sie sie hinzugefügt haben AWS Secrets Manager.

## Schritt 5: Speichern Sie den privaten PGP-Schlüssel in AWS Secrets Manager

Sie müssen den privaten Schlüssel in Secrets Manager auf eine ganz bestimmte Weise speichern, damit der Workflow den privaten Schlüssel finden kann, wenn der Workflow einen Entschlüsselungsschritt für eine hochgeladene Datei ausführt.

> **Note**
>
> Wenn Sie Geheimnisse im Secrets Manager speichern, AWS-Konto fallen Gebühren an. Informationen zu Preisen erhalten Sie unter AWS Secrets Manager -Preise.

Um einen privaten PGP-Schlüssel in Secrets Manager zu speichern

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die AWS Secrets Manager Konsole unter https://console.aws.amazon.com/secretsmanager/.
2. Wählen Sie im linken Navigationsbereich Secrets aus.
3. Wählen Sie auf der Seite Secrets die Option Neues Geheimnis speichern aus.
4. Wählen Sie auf der Seite Geheimtyp auswählen für Geheimtyp die Option Anderer Geheimtyp aus.
5. Wählen Sie im Abschnitt Schlüssel/Wert-Paare die Registerkarte Schlüssel/Wert aus.
   - Schlüssel — Geben Sie ein. **`PGPPrivateKey`**
   - Wert — Fügen Sie den Text Ihres privaten Schlüssels in das Wertfeld ein.
6. Wählen Sie Zeile hinzufügen und wählen Sie im Abschnitt Schlüssel/Wert-Paare die Registerkarte Schlüssel/Wert-Paare.
   - Schlüssel — Geben Sie ein. **`PGPPassphrase`**

- Wert — Geben Sie die Passphrase ein, die Sie bei der Generierung Ihres PGP-Schlüsselpaars in verwendet haben. Schritt 4: Erstellen Sie ein PGP-Schlüsselpaar

7. Wählen Sie Weiter aus.
8. Geben Sie auf der Seite Geheimes Passwort konfigurieren einen Namen und eine Beschreibung für Ihr Geheimnis ein. Sie können ein Geheimnis für einen bestimmten Benutzer oder ein Geheimnis erstellen, das von allen Benutzern verwendet werden kann. Wenn Ihre Server-ID lautet *s-11112222333344445*, benennen Sie das Geheimnis wie folgt.

   - Um ein Standardgeheimnis für alle Benutzer zu erstellen, geben Sie dem Geheimnis einen Namen**aws/transfer/*s-11112222333344445*/@pgp-default**.
   - Um ein Geheimnis nur für den Benutzer zu erstellen, den Sie zuvor erstellt haben, geben Sie dem Geheimnis einen Namen**aws/transfer/*s-11112222333344445*/decrypt-user**.
9. Wählen Sie Weiter und akzeptieren Sie dann die Standardeinstellungen auf der Seite „Rotation konfigurieren“. Wählen Sie anschließend Weiter.
10. Wählen Sie auf der Seite „Überprüfen“ die Option Speichern aus, um das Geheimnis zu erstellen und zu speichern.

Weitere Informationen zum Hinzufügen Ihres privaten PGP-Schlüssels zu Secrets Manager finden Sie unter AWS Secrets Manager Zum Speichern Ihres PGP-Schlüssels verwenden.

## Schritt 6: Verschlüsseln Sie eine Datei

Verwenden Sie das gpg Programm, um eine Datei für die Verwendung in Ihrem Workflow zu verschlüsseln. Führen Sie den folgenden Befehl aus, um eine Datei zu verschlüsseln:

```
gpg -e -r marymajor@example.com --openpgp testfile.txt
```

Bevor Sie diesen Befehl ausführen, beachten Sie Folgendes:

- Ersetzen *marymajor@example.com* Sie das `-r` Argument durch die E-Mail-Adresse, die Sie bei der Erstellung des PGP-Schlüsselpaars verwendet haben.
- Die `--openpgp` Markierung ist optional. Dieses Flag sorgt dafür, dass die verschlüsselte Datei dem OpenPGP-Standard RFC4880 entspricht.
- Dieser Befehl erstellt eine Datei mit dem Namen **`testfile.txt.gpg`** am selben Ort wie. **`testfile.txt`**

# Schritt 7: Führen Sie den Workflow aus und sehen Sie sich die Ergebnisse an

Um den Workflow auszuführen, stellen Sie mit dem Benutzer, den Sie in Schritt 3 erstellt haben, eine Verbindung zum Transfer Family Family-Server her. Anschließend können Sie in dem Amazon S3 S3-Bucket, den Sie in Schritt 2.5 angegeben haben, die Zielparameter so konfigurieren, dass die entschlüsselte Datei angezeigt wird.

Um den Entschlüsselungs-Workflow auszuführen

1. Öffnen Sie ein Befehlsterminal.
2. Führen Sie den folgenden Befehl aus und *`your-endpoint`* ersetzen Sie ihn durch Ihren tatsächlichen Endpunkt und *`transfer-key`* den privaten SSH-Schlüssel Ihres Benutzers:

   ```
   sftp -i transfer-key decrypt-user@your-endpoint
   ```

   Wenn der private Schlüssel beispielsweise in `~/.ssh/decrypt-user` gespeichert ist und Ihr Endpunkt darin gespeichert ist`s-11112222333344445.server.transfer.us-east-2.amazonaws.com`, lautet der Befehl wie folgt:

   ```
   sftp -i  ~/.ssh/decrypt-user decrypt-user@s-11112222333344445.server.transfer.us-east-2.amazonaws.com
   ```
3. Führen Sie den Befehl pwd aus. Bei Erfolg gibt dieser Befehl Folgendes zurück:

   ```
   Remote working directory: /DOC-EXAMPLE-BUCKET/decrypt-user
   ```

   Ihr Verzeichnis spiegelt den Namen Ihres Amazon S3 S3-Buckets wider.
4. Führen Sie den folgenden Befehl aus, um die Datei hochzuladen und die Ausführung des Workflows auszulösen:

   ```
   put testfile.txt.gpg
   ```
5. Für das Ziel der entschlüsselten Dateien haben Sie den `decrypted-files/` Ordner angegeben, als Sie den Workflow erstellt haben. Jetzt können Sie zu diesem Ordner navigieren und den Inhalt auflisten.

   ```
   cd ../decrypted-files/
   ```

```
ls
```

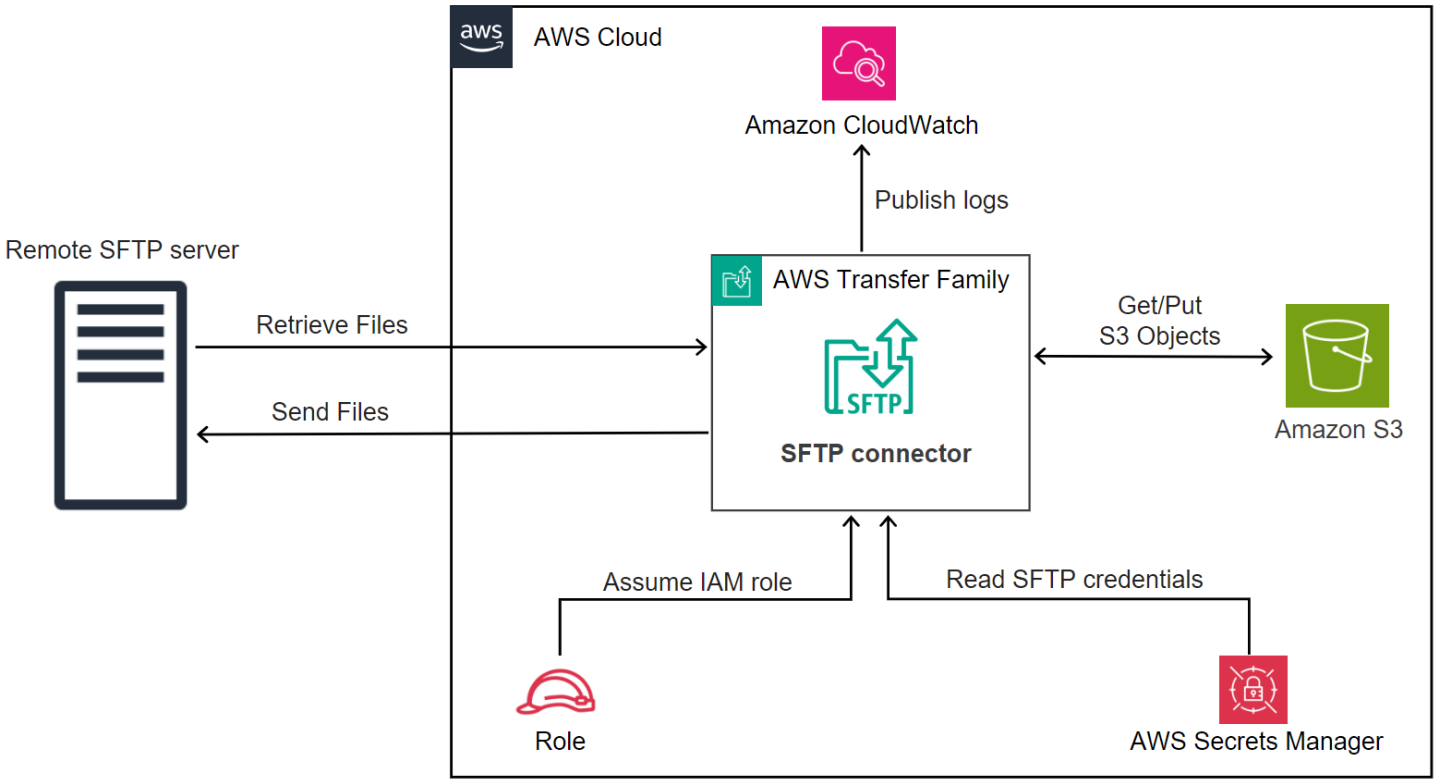
Bei Erfolg listet der `ls` Befehl die `testfile.txt` Datei auf. Sie können diese Datei herunterladen und überprüfen, ob sie mit der Originaldatei identisch ist, die Sie zuvor verschlüsselt haben.

# Einrichtung und Verwendung von SFTP-Anschlüssen

Der Zweck eines Connectors besteht darin, eine Beziehung zwischen Ihrem AWS Speicher und dem SFTP-Server eines Partners herzustellen. Sie können Dateien von Amazon S3 an ein externes, partnereigenes Ziel senden. Sie können auch einen SFTP-Connector verwenden, um Dateien vom SFTP-Server eines Partners abzurufen.

Dieses Tutorial zeigt, wie Sie einen SFTP-Connector einrichten und anschließend Dateien zwischen Amazon S3 S3-Speicher und einem SFTP-Server übertragen.

Ein SFTP-Connector ruft SFTP-Anmeldeinformationen ab, um sich bei einem Remote-SFTP-Server AWS Secrets Manager zu authentifizieren und eine Verbindung herzustellen. Der Connector sendet Dateien an den Remote-Server oder ruft Dateien vom Remote-Server ab und speichert die Dateien in Amazon S3. Eine IAM-Rolle wird verwendet, um den Zugriff auf den Amazon S3 S3-Bucket und auf die in Secrets Manager gespeicherten Anmeldeinformationen zu ermöglichen. Und Sie können sich bei Amazon anmelden CloudWatch.




Die folgenden Blogbeiträge bieten eine Referenzarchitektur für die Erstellung eines MFT-Workflows mithilfe von SFTP-Konnektoren, einschließlich der Verschlüsselung von Dateien mit PGP, bevor sie mithilfe von SFTP-Konnektoren an einen Remote-SFTP-Server gesendet werden: Architektur sicherer und richtlinienkonformer verwalteter Dateiübertragungen mit SFTP-Konnektoren und PGP-Verschlüsselung. AWS Transfer Family

Themen

- Schritt 1: Erstellen Sie die erforderlichen unterstützenden Ressourcen
- Schritt 2: Erstellen und testen Sie einen SFTP-Connector
- Schritt 3: Senden und Abrufen von Dateien mithilfe des SFTP-Connectors
- Verfahren zum Erstellen eines Transfer Family Family-Servers, der als Remote-SFTP-Server verwendet werden kann

## Schritt 1: Erstellen Sie die erforderlichen unterstützenden Ressourcen

Sie können SFTP-Konnektoren verwenden, um Dateien zwischen Amazon S3 und einem beliebigen Remote-SFTP-Server zu kopieren. Für dieses Tutorial verwenden wir einen AWS Transfer Family Server als Remote-SFTP-Server. Wir müssen die folgenden Ressourcen erstellen und konfigurieren:

- Erstellen Sie Amazon S3 S3-Buckets, um Dateien in Ihrer AWS Umgebung zu speichern und Dateien vom Remote-SFTP-Server zu senden und abzurufen:. Amazon S3 S3-Buckets erstellen
- Erstellen Sie eine AWS Identity and Access Management Rolle für den Zugriff auf Amazon S3 S3-Speicher und unser Geheimnis in Secrets Manager:Erstellen Sie eine IAM-Rolle mit den erforderlichen Berechtigungen.
- Erstellen Sie einen Transfer Family Family-Server, der das SFTP-Protokoll verwendet, und einen vom Dienst verwalteten Benutzer, der den SFTP-Connector verwendet, um Dateien zum oder vom SFTP-Server zu übertragen:. Erstellen Sie einen Transfer Family SFTP-Server und einen Benutzer
- Erstellen Sie einen AWS Secrets Manager geheimen Schlüssel, der die Anmeldeinformationen speichert, die der SFTP-Connector für die Anmeldung am Remote-SFTP-Server verwendet:. Erstellen und speichern Sie ein Geheimnis in AWS Secrets Manager

## Amazon S3 S3-Buckets erstellen

So erstellen Sie einen Amazon-S3-Bucket

1. Melden Sie sich unter https://console.aws.amazon.com/s3/ bei der AWS Transfer Family Konsole an.
2. Wählen Sie eine Region und geben Sie einen Namen ein.

   In diesem Tutorial ist unser Bucket **`US East (N. Virginia) us-east-1`** dabei und der Name lautet**`sftp-server-storage-east`**.
3. Akzeptieren Sie die Standardeinstellungen und wählen Sie Create Bucket.

Vollständige Informationen zum Erstellen von Amazon S3 S3-Buckets finden Sie unter Wie erstelle ich einen S3-Bucket? im Amazon Simple Storage Service-Benutzerhandbuch.

## Erstellen Sie eine IAM-Rolle mit den erforderlichen Berechtigungen

Erstellen Sie für die Zugriffsrolle eine Richtlinie mit den folgenden Berechtigungen.

Das folgende Beispiel gewährt die erforderlichen Berechtigungen für den Zugriff auf den *`DOC-EXAMPLE-BUCKET`* in Amazon S3 und den angegebenen Secret, der in Secrets Manager gespeichert ist.

```
{
  "Version": "2012-10-17",
```

```
  "Statement": [
    {
        "Sid": "AllowListingOfUserFolder",
        "Action": [
            "s3:ListBucket",
            "s3:GetBucketLocation"
        ],
        "Effect": "Allow",
        "Resource": [
            "arn:aws:s3:::DOC-EXAMPLE-BUCKET"
        ]
    },
    {
        "Sid": "HomeDirObjectAccess",
        "Effect": "Allow",
        "Action": [
            "s3:PutObject",
            "s3:GetObject",
            "s3:DeleteObject",
            "s3:DeleteObjectVersion",
            "s3:GetObjectVersion",
            "s3:GetObjectACL",
            "s3:PutObjectACL"
        ],
        "Resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
    },
    {
        "Sid": "GetConnectorSecretValue",
        "Effect": "Allow",
        "Action": [
            "secretsmanager:GetSecretValue"
        ],
        "Resource": "arn:aws:secretsmanager:region:account-id:secret:aws/
transfer/SecretName-6RandomCharacters"
    }
  ]
}
```

Ersetzen Sie Elemente wie folgt:

- Für *DOC-EXAMPLE-BUCKET verwendet das* Tutorial. **s3-storage-east**
- Für die *Region verwendet das Tutorial.* **us-east-1**
- Verwenden Sie für die *Konto-ID* Ihre AWS-Konto ID.

- Bei *SecretName-6 RandomCharacters* sind wir **using sftp-connector1** für den Namen (Sie haben Ihre eigenen sechs zufälligen Zeichen für Ihr Geheimnis).

Sie müssen außerdem sicherstellen, dass diese Rolle eine Vertrauensstellung beinhaltet, die es dem Connector ermöglicht, auf Ihre Ressourcen zuzugreifen, wenn er Übertragungsanfragen Ihrer Benutzer bearbeitet. Einzelheiten zum Aufbau einer Vertrauensbeziehung finden Sie unterSo stellen Sie eine Vertrauensbeziehung her.

> **Note**
>
> Einzelheiten zu der Rolle, die wir für das Tutorial verwenden, finden Sie unterKombinierte Benutzer- und Zugriffsrolle.

## Erstellen und speichern Sie ein Geheimnis in AWS Secrets Manager

Wir müssen ein Geheimnis in Secrets Manager speichern, um Benutzeranmeldeinformationen für Ihren SFTP-Connector zu speichern. Sie können ein Passwort, einen privaten SSH-Schlüssel oder beides verwenden. Für das Tutorial verwenden wir einen privaten Schlüssel.

> **Note**
>
> Wenn Sie Geheimnisse im Secrets Manager speichern, AWS-Konto fallen Gebühren an. Informationen zu Preisen erhalten Sie unter AWS Secrets Manager -Preise.

Bevor Sie mit dem Verfahren zum Speichern des Geheimnisses beginnen, müssen Sie Ihren privaten Schlüssel abrufen und formatieren. Der private Schlüssel muss dem öffentlichen Schlüssel entsprechen, der für den Benutzer auf dem Remote-SFTP-Server konfiguriert ist. Für unser Tutorial muss der private Schlüssel dem öffentlichen Schlüssel entsprechen, der für unseren Testbenutzer auf dem Transfer Family SFTP-Server gespeichert ist, den wir als Remote-Server verwenden.

Führen Sie dazu den folgenden Befehl aus:

```
jq -sR . path-to-private-key-file
```

Wenn sich Ihre private Schlüsseldatei beispielsweise in befindet`~/.ssh/sftp-testuser-privatekey`, lautet der Befehl wie folgt.

```
jq -sR . ~/.ssh/sftp-testuser-privatekey
```

Dadurch wird der Schlüssel im richtigen Format (mit eingebetteten Zeilenumbruchzeichen) in die Standardausgabe ausgegeben. Kopieren Sie diesen Text irgendwo, da Sie ihn im folgenden Verfahren (in Schritt 6) einfügen müssen.

Um Benutzeranmeldeinformationen in Secrets Manager für einen SFTP-Connector zu speichern

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die AWS Secrets Manager Konsole unter https://console.aws.amazon.com/secretsmanager/.
2. Wählen Sie im linken Navigationsbereich Secrets aus.
3. Wählen Sie auf der Seite Secrets die Option Neues Geheimnis speichern aus.
4. Wählen Sie auf der Seite Geheimtyp auswählen für Geheimtyp die Option Anderer Geheimtyp aus.
5. Wählen Sie im Abschnitt Schlüssel/Wert-Paare die Registerkarte Schlüssel/Wert aus.
   - Schlüssel — Geben Sie ein. **`Username`**
   - Wert — Geben Sie den Namen unseres Benutzers ein,**`sftp-testuser`**.
6. Um den Schlüssel einzugeben, empfehlen wir, die Registerkarte Klartext zu verwenden.
   a. Wählen Sie Zeile hinzufügen und geben Sie dann die Eingabetaste ein**`PrivateKey`**.
   b. Wählen Sie die Registerkarte Klartext. Das Feld enthält jetzt den folgenden Text:

      ```
      {"Username":"sftp-testuser","PrivateKey":""}
      ```

   c. Fügen Sie den Text für Ihren privaten Schlüssel (zuvor gespeichert) zwischen den leeren doppelten Anführungszeichen („“) ein.

      Ihr Bildschirm sollte wie folgt aussehen (Schlüsseldaten sind ausgegraut).

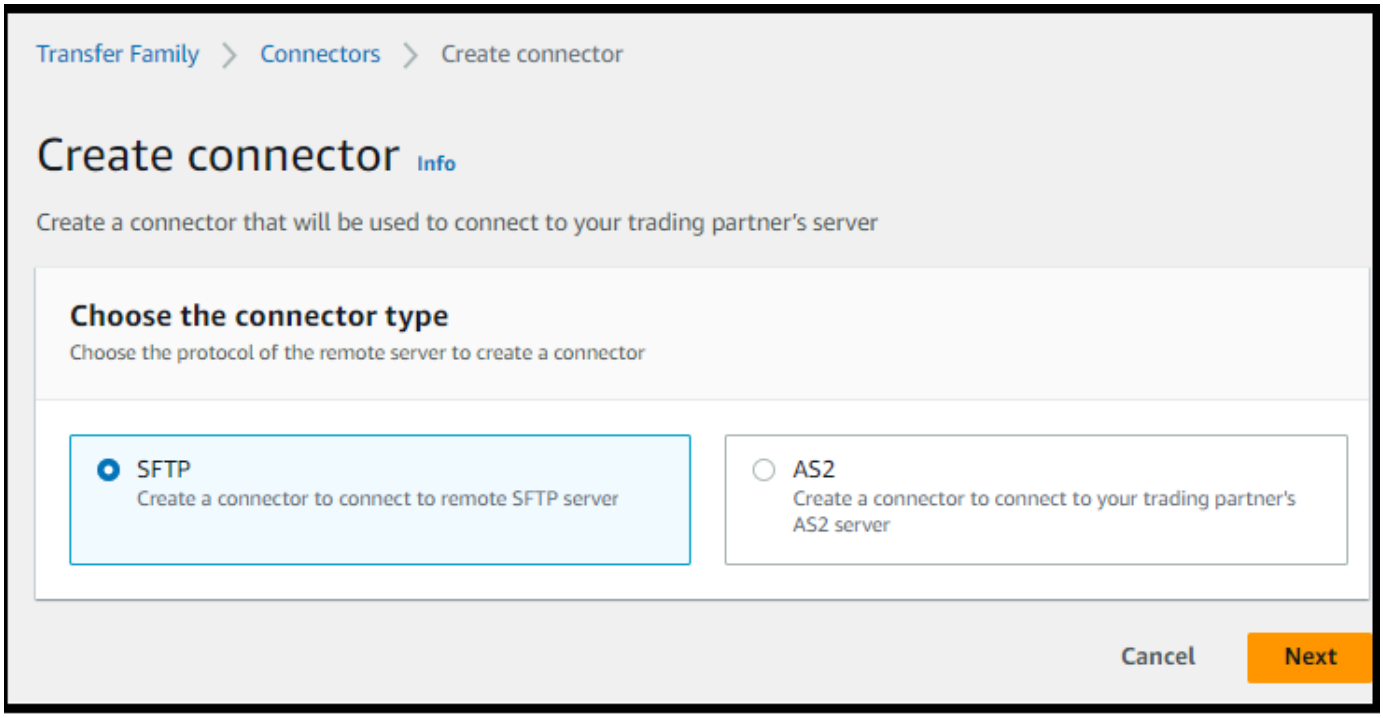

7. Wählen Sie Weiter aus.
8. Geben Sie auf der Seite Geheim konfigurieren einen Namen für Ihr Geheimnis ein. In diesem Tutorial geben wir dem Geheimnis einen Namen**`aws/transfer/sftp-connector1`**.
9. Wählen Sie Weiter und akzeptieren Sie dann die Standardeinstellungen auf der Seite „Rotation konfigurieren“. Wählen Sie anschließend Weiter.
10. Wählen Sie auf der Seite „Überprüfen“ die Option Speichern aus, um das Geheimnis zu erstellen und zu speichern.

## Schritt 2: Erstellen und testen Sie einen SFTP-Connector

In diesem Abschnitt erstellen wir einen SFTP-Connector, der alle Ressourcen verwendet, die wir zuvor erstellt haben. Weitere Details finden Sie unter Konfigurieren Sie SFTP-Anschlüsse.

Um einen SFTP-Connector zu erstellen

1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Wählen Sie im linken Navigationsbereich Connectors und anschließend Create Connector aus.
3. Wählen Sie SFTP als Konnektortyp aus, um einen SFTP-Connector zu erstellen, und wählen Sie dann Weiter aus.




4. Geben Sie im Abschnitt Connector-Konfiguration die folgenden Informationen ein:

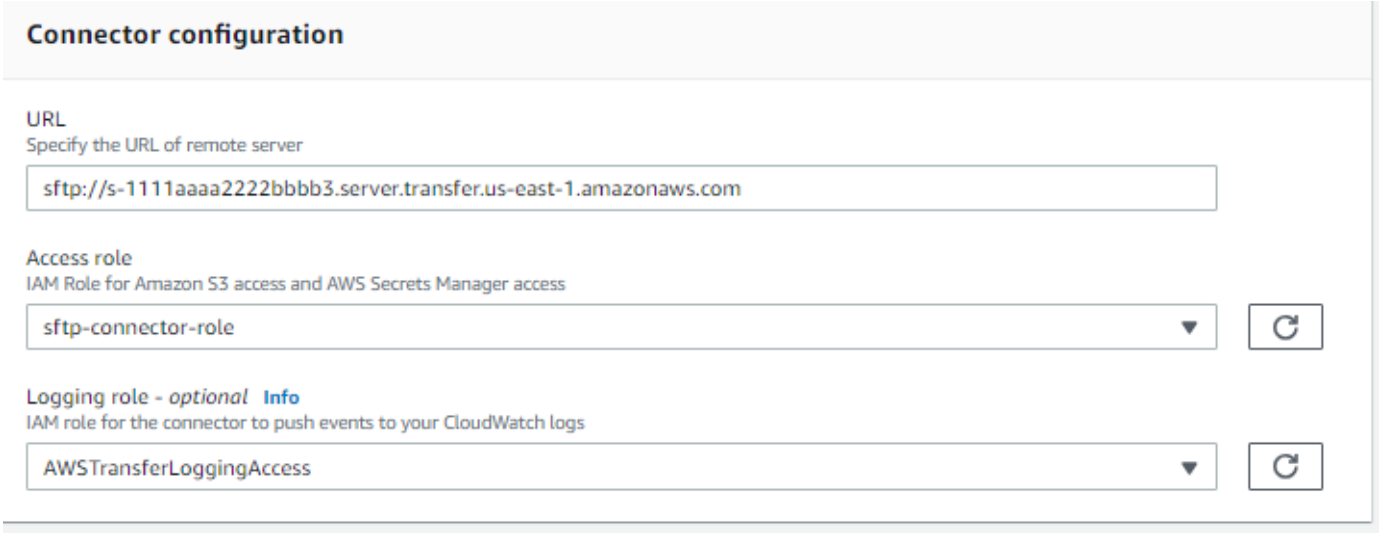
   - Geben Sie für die URL die URL des Remote-SFTP-Servers ein. Für das Tutorial geben wir die URL des Transfer Family Family-Servers ein, den wir als Remote-SFTP-Server verwenden.

     ```
     sftp://s-1111aaaa2222bbbb3.server.transfer.us-east-1.amazonaws.com
     ```

     Ersetzen Sie *1111aaaa2222bbbb3* durch Ihre Transfer Family Family-Server-ID.

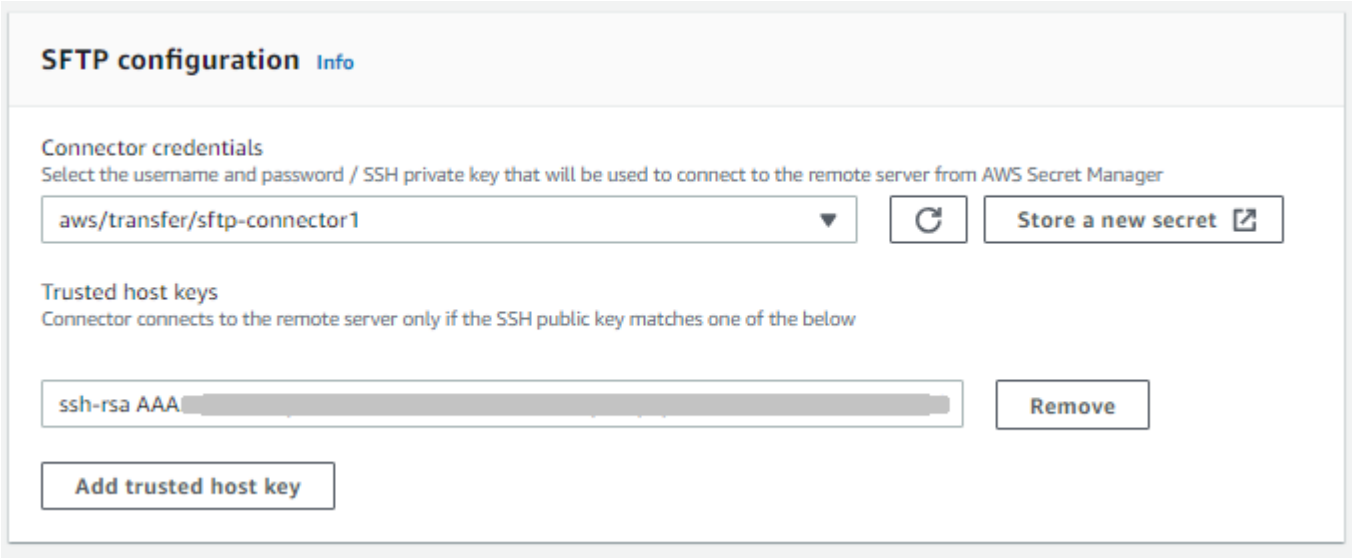
   - Geben Sie für die Access-Rolle die Rolle ein, die wir zuvor erstellt haben. **`sftp-connector-role`**
   - Wählen Sie für die Logging-Rolle**`AWSTransferLoggingAccess`**.

     > **Note**
     >
     > AWSTransferLoggingAccess ist eine AWS verwaltete Richtlinie. Diese Richtlinie wird ausführlich unter beschriebenAWS verwaltete Richtlinie: AWSTransferLoggingAccess.




5. Geben Sie im Abschnitt SFTP-Konfiguration die folgenden Informationen ein:

   - Wählen Sie für Connector-Anmeldeinformationen den Namen Ihrer Secrets Manager Manager-Ressource, die SFTP-Anmeldeinformationen enthält. Für das Tutorial wählen **`aws/transfer/sftp-connector1`** Sie.
   - Fügen Sie für vertrauenswürdige Hostschlüssel den öffentlichen Teil des Hostschlüssels ein. Sie können diesen Schlüssel abrufen, indem Sie ihn `ssh-keyscan` für Ihren SFTP-Server ausführen. Einzelheiten zum Formatieren und Speichern des vertrauenswürdigen Host-Schlüssels finden Sie in der Dokumentation zum SftpConnectorConfigDatentyp.



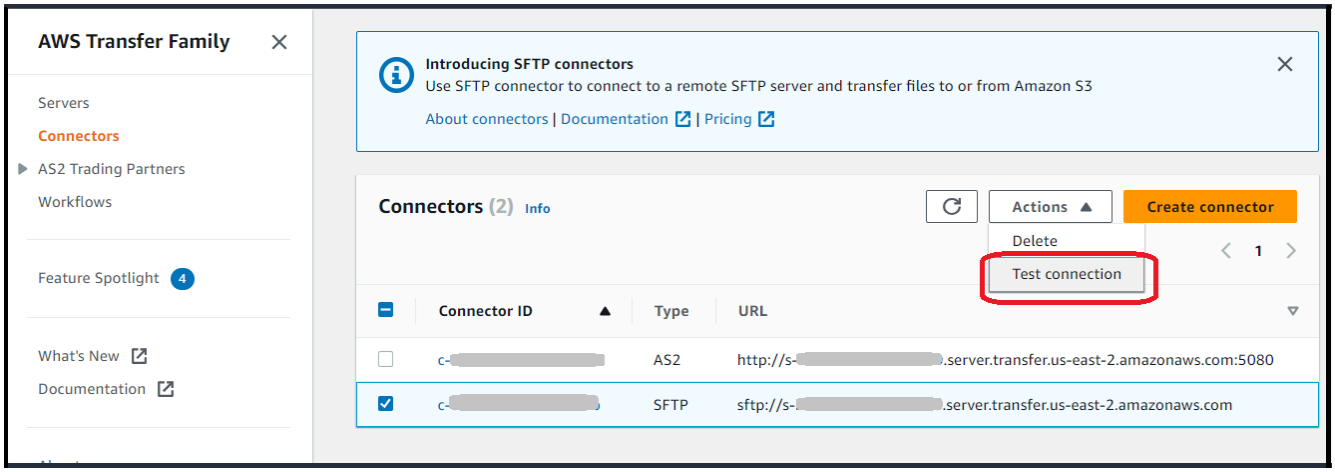
6. Nachdem Sie alle Ihre Einstellungen bestätigt haben, wählen Sie Connector erstellen, um den SFTP-Connector zu erstellen.

Nachdem Sie einen SFTP-Connector erstellt haben, empfehlen wir Ihnen, ihn zu testen, bevor Sie versuchen, Dateien mit Ihrem neuen Connector zu übertragen.

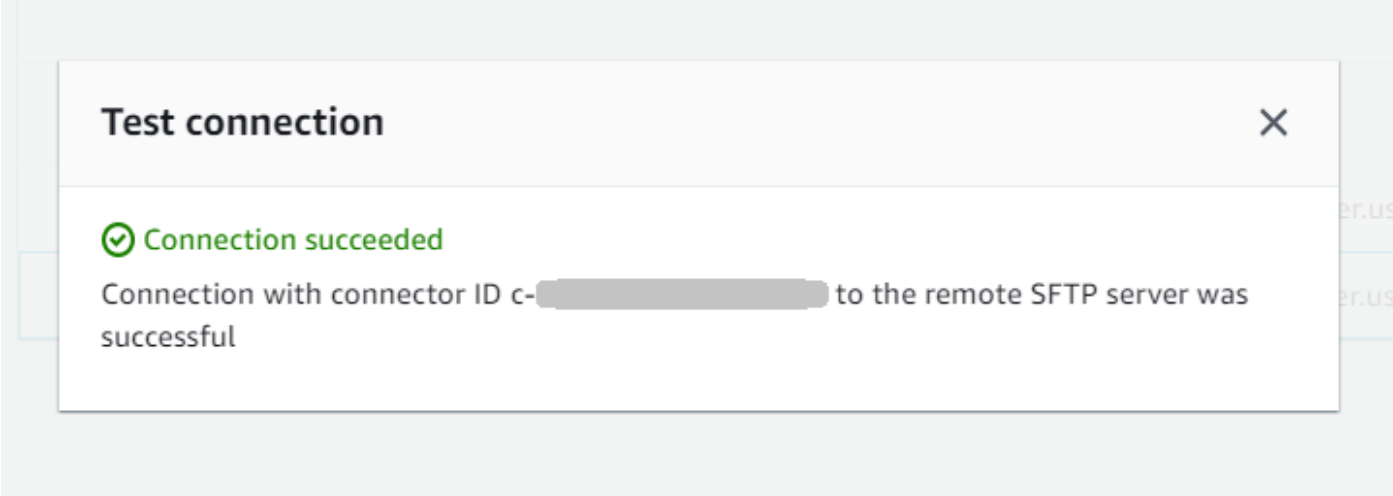
Test a connector using the console

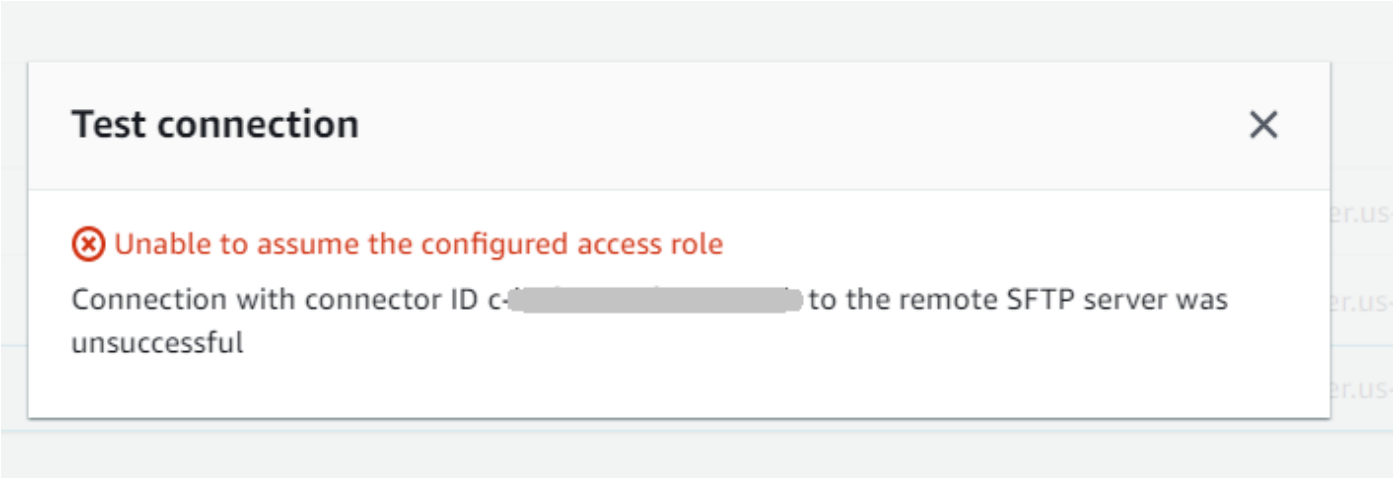
Um einen SFTP-Connector zu testen

1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Wählen Sie im linken Navigationsbereich Connectors und wählen Sie einen Connector aus.
3. Wählen Sie im Menü Aktionen die Option Verbindung testen aus.




Das System gibt eine Meldung zurück, in der angegeben wird, ob der Test erfolgreich war oder nicht. Wenn der Test fehlschlägt, gibt das System eine Fehlermeldung aus, die auf dem Grund basiert, warum der Test fehlgeschlagen ist.




Test a connector using the CLI

Um einen Connector mit dem zu testen AWS Command Line Interface, führen Sie den folgenden Befehl an einer Befehlszeile aus (ersetzen Sie *connector-id durch Ihre tatsächliche Connector-ID*):

```
aws transfer test-connection --connector-id c-connector-id
```

Wenn der Test erfolgreich ist, werden die folgenden Zeilen zurückgegeben:

```
{
   "Status": "OK",
   "StatusMessage": "Connection succeeded"
}
```

Wenn der Test nicht erfolgreich ist, erhalten Sie eine beschreibende Fehlermeldung, zum Beispiel:

```
{
   "Status": "ERROR",
   "StatusMessage": "Unable to assume the configured access role"
}
```

## Schritt 3: Senden und Abrufen von Dateien mithilfe des SFTP-Connectors

Der Einfachheit halber gehen wir davon aus, dass Sie bereits Dateien in Ihrem Amazon S3 S3-Bucket haben.

> **Note**
>
> Das Tutorial verwendet Amazon S3 S3-Buckets sowohl für Quell- als auch für Zielspeicherorte. Wenn Ihr SFTP-Server keinen Amazon S3 S3-Speicher verwendet, können Sie, wo immer Sie `sftp-server-storage-east` in den folgenden Befehlen sehen, den Pfad durch einen Pfad zu Dateispeicherorten ersetzen, auf die von Ihrem SFTP-Server aus zugegriffen werden kann.

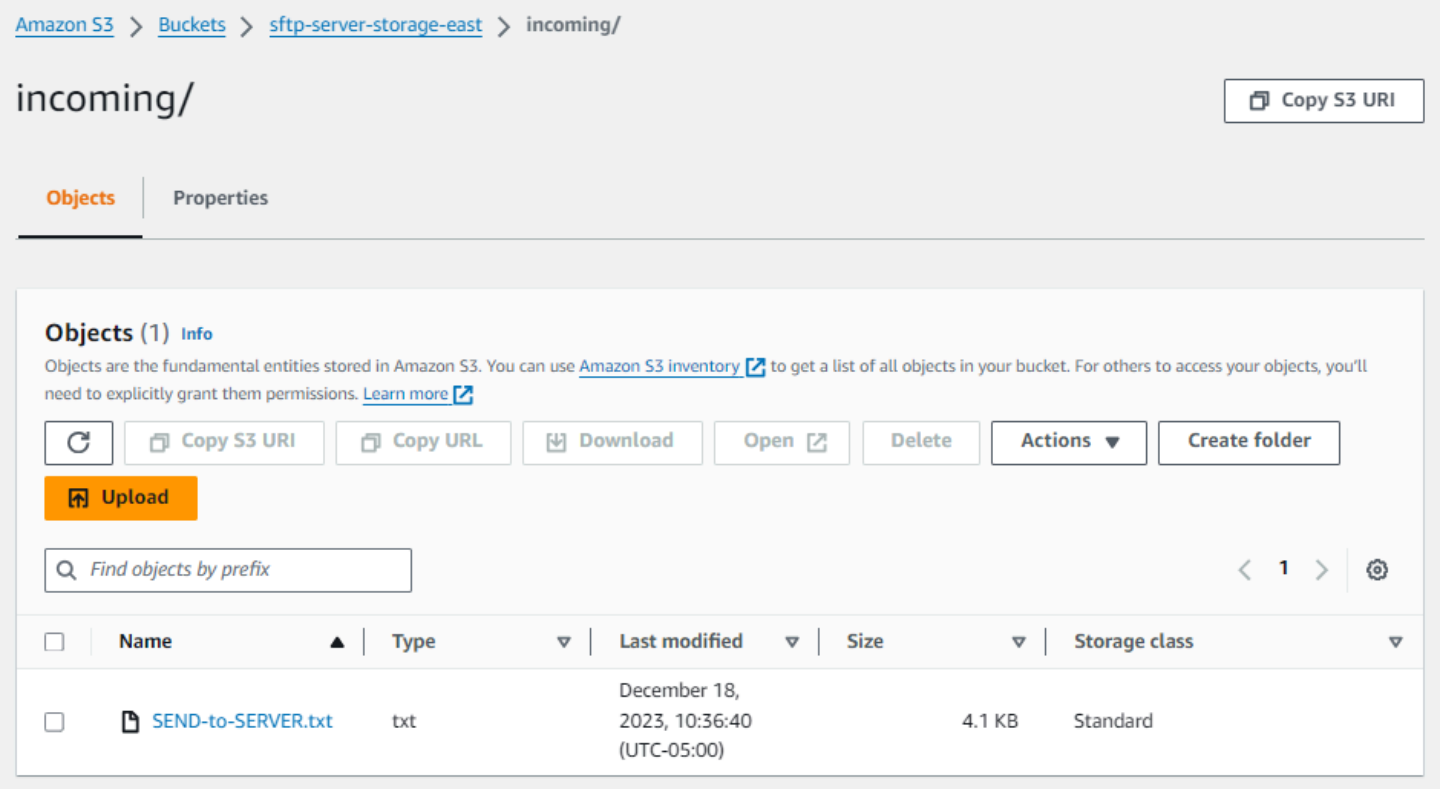
- Wir senden eine `SEND-to-SERVER.txt` vom Amazon S3 S3-Speicher benannte Datei an den SFTP-Server.
- Wir rufen eine Datei mit dem Namen `RETRIEVE-to-S3.txt` vom SFTP-Server in den Amazon S3 S3-Speicher ab.

> **Note**
>
> Ersetzen Sie in den folgenden Befehlen die *`Connector-ID durch Ihre Connector-ID`*.

Zunächst senden wir eine Datei von unserem Amazon S3 S3-Bucket an den Remote-SFTP-Server. Führen Sie in einer Befehlszeile den folgenden Befehl aus:

```
aws transfer start-file-transfer --connector-id c-connector-id --send-file-paths "/s3-
storage-east/SEND-to-SERVER.txt" /
   --remote-directory-path "/sftp-server-storage-east/incoming"
```

Ihr `sftp-server-storage-east` Bucket sollte jetzt so aussehen.




Wenn Sie die Datei nicht wie erwartet sehen, überprüfen Sie Ihre CloudWatch Protokolle.

Um deine CloudWatch Logs zu überprüfen

1. Öffnen Sie die CloudWatch Amazon-Konsole unter https://console.aws.amazon.com/cloudwatch/
2. Wählen Sie im linken Navigationsmenü Protokollgruppen aus.
3. Geben Sie Ihre Connector-ID in die Suchleiste ein, um Ihre Logs zu finden.
4. Wählen Sie den Protokollstream aus, der bei der Suche zurückgegeben wird.
5. Erweitern Sie den neuesten Protokolleintrag.

Bei Erfolg sieht der Protokolleintrag wie folgt aus:

```
{
    "operation": "SEND",
    "timestamp": "2023-12-18T15:26:57.346283Z",
    "connector-id": "connector-id",
    "transfer-id": "transfer-id",
    "file-transfer-id": "transfer-id/file-transfer-id",
    "url": "sftp://server-id.server.transfer.us-east-1.amazonaws.com",
    "file-path": "/s3-storage-east/SEND-to-SERVER.txt",
```

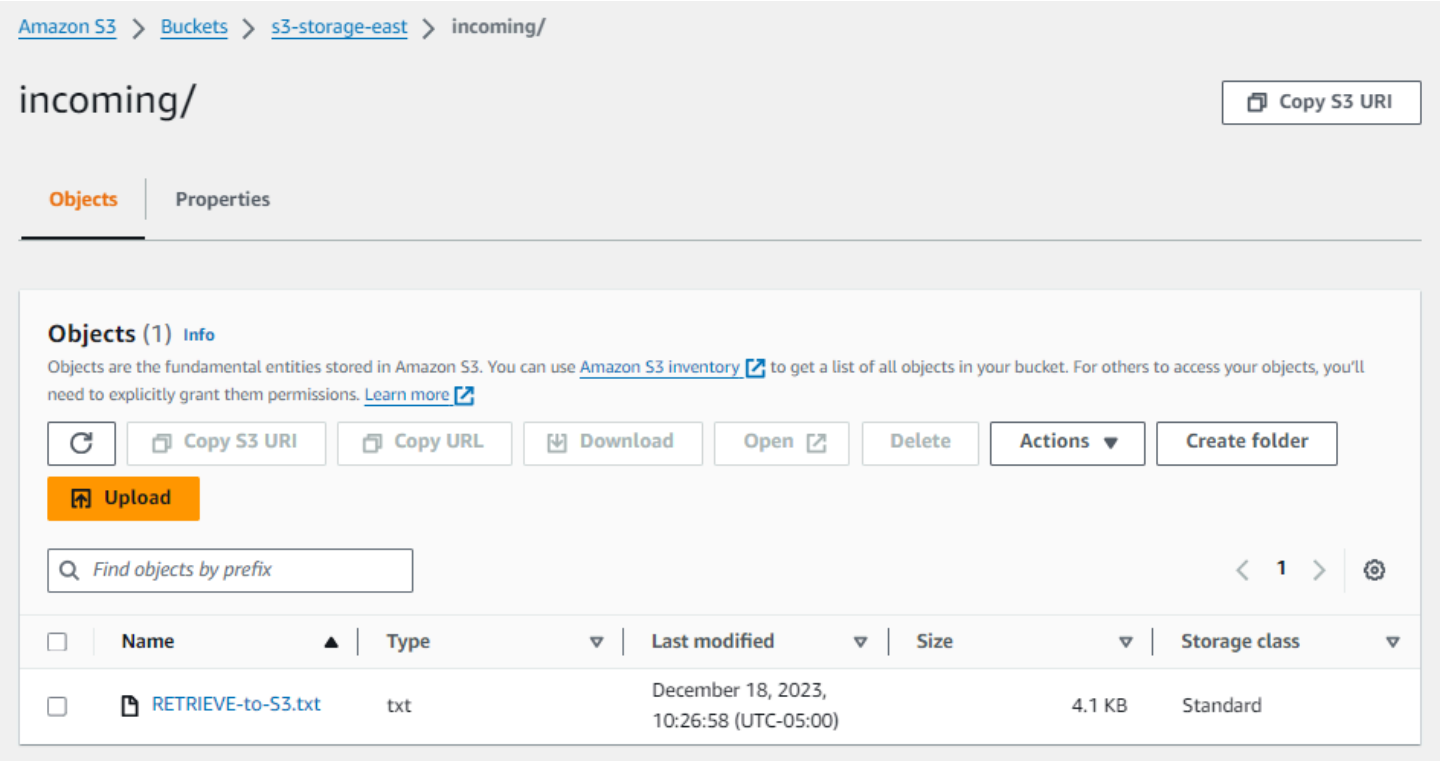
```
    "status-code": "COMPLETED",
    "start-time": "2023-12-18T15:26:56.915864Z",
    "end-time": "2023-12-18T15:26:57.298122Z",
    "account-id": "500655546075",
    "connector-arn": "arn:aws:transfer:us-east-1:500655546075:connector/connector-id",
    "remote-directory-path": "/sftp-server-storage-east/incoming"
}
```

Wenn die Dateiübertragung fehlgeschlagen ist, enthält der Protokolleintrag eine Fehlermeldung, die das Problem spezifiziert. Häufige Ursachen für Fehler sind Probleme mit den IAM-Berechtigungen und falsche Dateipfade.

Als Nächstes rufen wir eine Datei vom SFTP-Server in einen Amazon S3-Bucket ab. Führen Sie an einer Eingabeaufforderung den folgenden Befehl aus:

```
aws transfer start-file-transfer --connector-id c-connector-id --retrieve-file-paths "/
sftp-server-storage-east/RETRIEVE-to-S3.txt" --local-directory-path "/s3-storage-east/
incoming"
```

Wenn die Übertragung erfolgreich ist, enthält Ihr Amazon S3 S3-Bucket die übertragene Datei, wie hier gezeigt.




Bei Erfolg sieht der Protokolleintrag wie folgt aus:

```
{
    "operation": "RETRIEVE",
    "timestamp": "2023-12-18T15:36:40.017800Z",
    "connector-id": "c-connector-id",
    "transfer-id": "transfer-id",
    "file-transfer-id": "transfer-id/file-transfer-id",
    "url": "sftp://s-server-id.server.transfer.us-east-1.amazonaws.com",
    "file-path": "/sftp-server-storage-east/RETRIEVE-to-S3.txt",
    "status-code": "COMPLETED",
    "start-time": "2023-12-18T15:36:39.727626Z",
    "end-time": "2023-12-18T15:36:39.895726Z",
    "account-id": "500655546075",
    "connector-arn": "arn:aws:transfer:us-east-1:500655546075:connector/c-connector-
id",
    "local-directory-path": "/s3-storage-east/incoming"
}
```

## Verfahren zum Erstellen eines Transfer Family Family-Servers, der als Remote-SFTP-Server verwendet werden kann

Im Folgenden beschreiben wir die Schritte zum Erstellen eines Transfer Family Family-Servers, der als Remote-SFTP-Server für dieses Tutorial dient. Beachten Sie Folgendes:

- Wir verwenden einen Transfer Family Family-Server, um einen Remote-SFTP-Server darzustellen. Typische Benutzer von SFTP-Connectoren haben ihren eigenen Remote-SFTP-Server. Siehe Erstellen Sie einen Transfer Family SFTP-Server und einen Benutzer.
- Da wir einen Transfer Family Family-Server verwenden, verwenden wir auch einen vom Service verwalteten SFTP-Benutzer. Der Einfachheit halber haben wir die Berechtigungen, die dieser Benutzer für den Zugriff auf den Transfer Family Family-Server benötigt, mit den Berechtigungen kombiniert, die er für die Verwendung unseres Connectors benötigt. Auch hier haben die meisten Anwendungsfälle für SFTP-Connectoren einen separaten SFTP-Benutzer, der keinem Transfer Family Family-Server zugeordnet ist. Siehe Erstellen Sie einen Transfer Family SFTP-Server und einen Benutzer.
- Da wir Amazon S3-Speicher für unseren Remote-SFTP-Server verwenden, müssen wir für das Tutorial einen zweiten Bucket erstellen**`s3-storage-east`**, damit wir Dateien von einem Bucket in einen anderen übertragen können.

## Erstellen Sie einen Transfer Family SFTP-Server und einen Benutzer

Die meisten Benutzer müssen keinen Transfer Family Family-SFTP-Server und keinen Benutzer erstellen, da Sie bereits über einen SFTP-Server mit Benutzern verfügen und diesen Server verwenden können, um Dateien von und zu übertragen. In diesem Tutorial verwenden wir der Einfachheit halber jedoch einen Transfer Family Family-Server, der als Remote-SFTP-Server fungiert.

Folgen Sie den unter Erstellen Sie einen SFTP-fähigen Server So erstellen Sie einen Server und fügen Sie einen Benutzer Schritt 3: Fügen Sie einen vom Service verwalteten Benutzer hinzu hinzu. Dies sind die Benutzerdetails, die wir für das Tutorial verwenden:

- Erstellen Sie Ihren vom Service verwalteten Benutzer,`sftp-testuser`.
  - Stellen Sie das Home-Verzeichnis ein auf `/sftp-server-storage-east/sftp-testuser`
  - Wenn Sie den Benutzer erstellen, speichern Sie einen öffentlichen Schlüssel. Später, wenn Sie das Geheimnis in Secrets Manager erstellen, müssen Sie den entsprechenden privaten Schlüssel angeben.
- Rolle:`sftp-connector-role`. Für das Tutorial verwenden wir dieselbe IAM-Rolle sowohl für unseren SFTP-Benutzer als auch für den Zugriff auf den SFTP-Connector. Wenn Sie Konnektoren für Ihre Organisation erstellen, haben Sie möglicherweise separate Benutzer- und Zugriffsrollen.
- Server-Hostschlüssel: Sie müssen den Server-Hostschlüssel verwenden, wenn Sie den Connector erstellen. Sie können diesen Schlüssel abrufen, indem Sie ihn `ssh-keyscan` für Ihren Server ausführen. Wenn Ihre Server-ID beispielsweise lautet `s-1111aaaa2222bbbb3` und ihr Endpunkt in ist`us-east-1`, ruft der folgende Befehl den Server-Host-Schlüssel ab:

  ```
  ssh-keyscan s-1111aaaa2222bbbb3.server.transfer.us-east-1.amazonaws.com
  ```

  Kopieren Sie diesen Text irgendwo, da Sie ihn in die Schritt 2: Erstellen und testen Sie einen SFTP-Connector Prozedur einfügen müssen.

## Kombinierte Benutzer- und Zugriffsrolle

Für das Tutorial verwenden wir eine einzelne, kombinierte Rolle. Wir verwenden diese Rolle sowohl für unseren SFTP-Benutzer als auch für den Zugriff auf den Connector. Das folgende Beispiel enthält die Details für diese Rolle, falls Sie die Aufgaben im Tutorial ausführen möchten.

Das folgende Beispiel gewährt die erforderlichen Berechtigungen für den Zugriff auf unsere beiden Buckets in Amazon S3 und das Secret namens, das in Secrets Manager `aws/transfer/sftp-connector1` gespeichert ist. Für das Tutorial wird diese Rolle benannt`sftp-connector-role`.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowListingOfUserFolder",
            "Action": [
                "s3:ListBucket",
                "s3:GetBucketLocation"
            ],
            "Effect": "Allow",
            "Resource": [
                "arn:aws:s3:::sftp-server-storage-east",
                "arn:aws:s3:::s3-storage-east"
            ]
        },
        {
            "Sid": "HomeDirObjectAccess",
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
                "s3:DeleteObject",
                "s3:DeleteObjectVersion",
                "s3:GetObjectVersion",
                "s3:GetObjectACL",
                "s3:PutObjectACL"
            ],
            "Resource": [
                "arn:aws:s3:::sftp-server-storage-east/*",
                "arn:aws:s3:::s3-storage-east/*"
            ]
        },
        {
            "Sid": "GetConnectorSecretValue",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue"
            ],
```

```
            "Resource": "arn:aws:secretsmanager:us-east-1:500655546075:secret:aws/
transfer/sftp-connector1-6RandomCharacters"
        }
    ]
}
```

Vollständige Informationen zum Erstellen von Rollen für Transfer Family finden Sie unter Eine Benutzerrolle erstellen So erstellen Sie eine Rolle.

# Einrichtung einer Amazon API Gateway Gateway-Methode als benutzerdefinierter Identitätsanbieter

Dieses Tutorial zeigt, wie Sie eine Amazon API Gateway Gateway-Methode einrichten und sie als benutzerdefinierten Identitätsanbieter verwenden, um Dateien auf einen AWS Transfer Family Server hochzuladen. In diesem Tutorial werden die Stack-Vorlage Basic und andere grundlegende Funktionen nur als Beispiel verwendet.

Themen


- Voraussetzungen
- Schritt 1: Erstellen Sie einen CloudFormation Stack
- Schritt 2: Überprüfen Sie die Konfiguration der API Gateway Gateway-Methode für Ihren Server
- Schritt 3: Die Transfer Family Family-Serverdetails anzeigen
- Schritt 4: Testen Sie, ob Ihr Benutzer eine Verbindung zum Server herstellen kann
- Schritt 5: Testen Sie die SFTP-Verbindung und die Dateiübertragung
- Schritt 6: Beschränken Sie den Zugriff auf den Bucket
- Lambda aktualisieren, wenn Sie Amazon EFS verwenden


## Voraussetzungen

Bevor Sie die Transfer Family Family-Ressourcen in erstellen AWS CloudFormation, erstellen Sie Ihren Speicher und Ihre Benutzerrolle.

Um Speicherplatz anzugeben und eine Benutzerrolle zu erstellen

1. Je nachdem, welchen Speicher Sie verwenden, finden Sie weitere Informationen in der folgenden Dokumentation:

- Informationen zum Erstellen eines Amazon S3 S3-Buckets finden Sie unter Wie erstelle ich einen S3-Bucket? im Amazon Simple Storage Service-Benutzerhandbuch.
- Informationen zum Erstellen eines Amazon EFS-Dateisystems finden Sie unterEin Amazon EFS-Dateisystem konfigurieren.

2. Informationen zum Erstellen einer Benutzerrolle finden Sie unter Erstellen Sie eine IAM-Rolle und -Richtlinie

Sie geben die Details für Ihren Speicher und Ihre Benutzerrolle ein, wenn Sie Ihren AWS CloudFormation Stack im nächsten Abschnitt erstellen.

## Schritt 1: Erstellen Sie einen CloudFormation Stack

Um einen AWS CloudFormation Stapel aus der bereitgestellten Vorlage zu erstellen

1. Öffnen Sie die AWS CloudFormation Konsole unter https://console.aws.amazon.com/cloudformation.
2. Wählen Sie Stack erstellen und anschließend Mit neuen Ressourcen (Standard) aus.
3. Wählen Sie im Bereich Voraussetzung — Vorlage vorbereiten die Option Vorlage ist bereit aus.
4. Kopieren Sie diesen Link, Basic Stack Template, und fügen Sie ihn in das Amazon S3 S3-URL-Feld ein.
5. Klicken Sie auf Weiter.
6. Geben Sie Parameter an, einschließlich eines Namens für Ihren Stack. Stellen Sie sicher, dass Sie Folgendes tun:

   - Ersetzen Sie die Standardwerte für UserNameund UserPassword.
   - Geben Sie für UserHomeDirectorydie Details für den Speicher (entweder einen Amazon S3 S3-Bucket oder ein Amazon EFS-Dateisystem) ein, den Sie zuvor erstellt haben.
   - Ersetzen Sie den Standard UserRoleArndurch die Benutzerrolle, die Sie zuvor erstellt haben. Die AWS Identity and Access Management (IAM-) Rolle muss über die entsprechenden Berechtigungen verfügen. Ein Beispiel für eine IAM-Rolle und eine Bucket-Richtlinie finden Sie unter. Schritt 6: Beschränken Sie den Zugriff auf den Bucket
   - Wenn Sie sich mit einem öffentlichen Schlüssel statt mit einem Passwort authentifizieren möchten, geben Sie Ihren öffentlichen Schlüssel in das Feld UserPublicKey1 ein. Wenn Sie zum ersten Mal über SFTP eine Verbindung zum Server herstellen, geben Sie anstelle eines Kennworts den privaten Schlüssel an.

7. Wählen Sie Weiter und klicken Sie dann auf der Seite Stack-Optionen konfigurieren erneut auf Weiter.
8. Überprüfen Sie die Details für den Stack, den Sie gerade erstellen, und wählen Sie dann Stapel erstellen aus.

> **Note**
>
> Unten auf der Seite müssen Sie unter Funktionen angeben, dass dadurch AWS CloudFormation möglicherweise IAM-Ressourcen erstellt werden.

## Schritt 2: Überprüfen Sie die Konfiguration der API Gateway Gateway-Methode für Ihren Server

> **Note**
>
> Um die Sicherheit zu verbessern, können Sie eine Firewall für Webanwendungen konfigurieren. AWS WAF ist eine Firewall für Webanwendungen, mit der Sie die HTTP- und HTTPS-Anfragen überwachen können, die an ein Amazon API Gateway weitergeleitet werden. Details hierzu finden Sie unter Fügen Sie eine Firewall für Webanwendungen hinzu.

Um die Konfiguration der API Gateway Gateway-Methode für Ihren Server zu überprüfen und bereitzustellen

1. Öffnen Sie die API Gateway-Konsole unter https://console.aws.amazon.com/apigateway/.
2. Wählen Sie die Basisvorlagen-API für Transfer Custom Identity Provider aus, die von der AWS CloudFormation Vorlage generiert wurde.
3. Wählen Sie im Bereich Ressourcen die Option GET und anschließend Method Request aus.
4. Wählen Sie für Aktionen die Option Deploy API aus. Wählen Sie für die Bereitstellungsphase die Option prod und dann Deploy aus.

   Nachdem die API Gateway Gateway-Methode erfolgreich bereitgestellt wurde, können Sie sich ihre Leistung im Abschnitt Stage Editor ansehen.

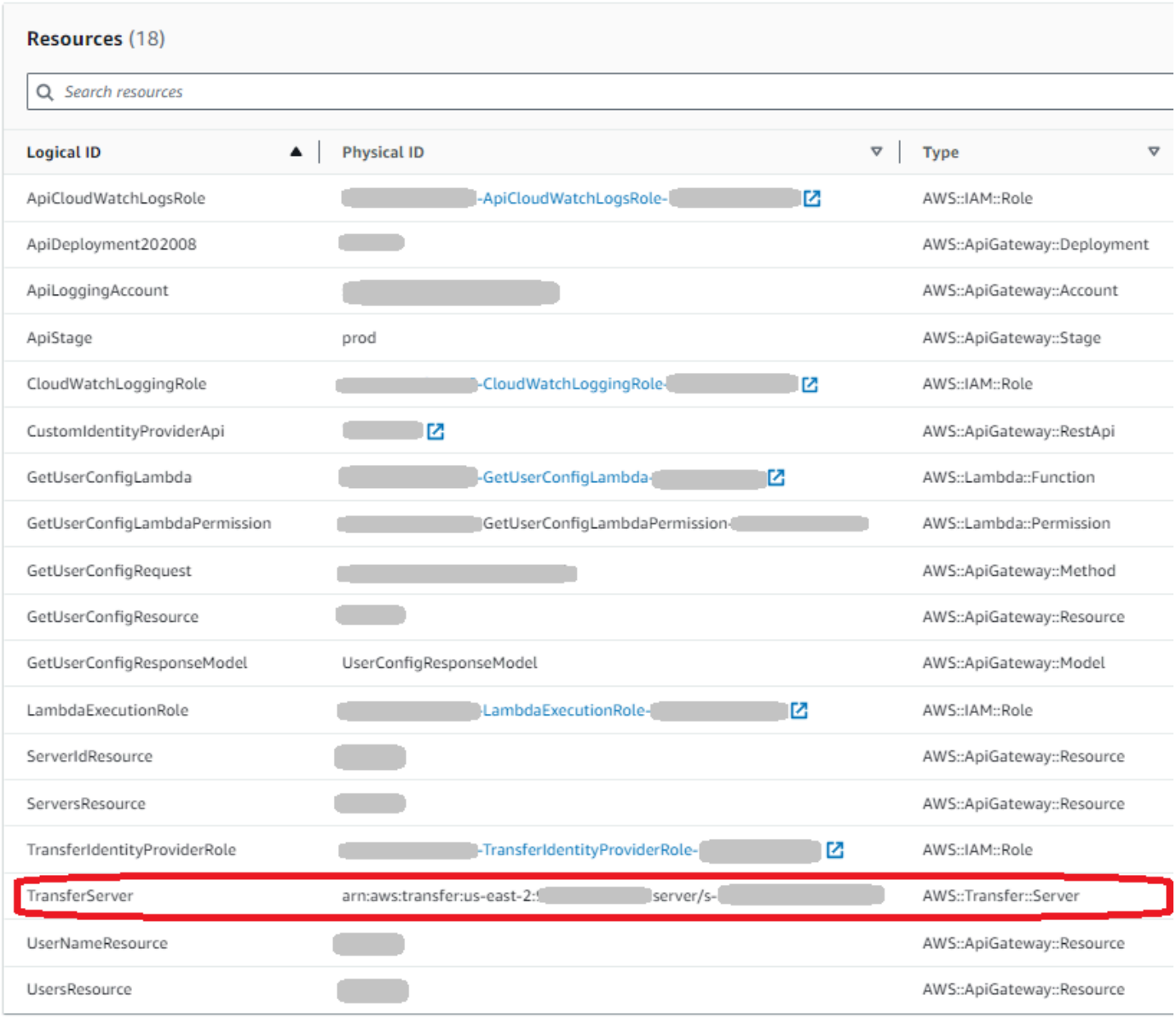
> **Note**
>
> Kopieren Sie die Aufruf-URL-Adresse, die oben auf der Seite angezeigt wird. Sie benötigen sie für den nächsten Schritt.

## Schritt 3: Die Transfer Family Family-Serverdetails anzeigen

Wenn Sie die Vorlage verwenden, um einen AWS CloudFormation Stack zu erstellen, wird automatisch ein Transfer Family Family-Server erstellt.

So zeigen Sie Ihre Transfer Family Family-Serverdetails an

1. Öffnen Sie die AWS CloudFormation Konsole unter https://console.aws.amazon.com/cloudformation.
2. Wählen Sie den Stack aus, den Sie erstellt haben.
3. Wählen Sie die Registerkarte Resources (Ressourcen) aus.




Der Server-ARN wird in der Spalte Physikalische ID für die TransferServerZeile angezeigt. Die Server-ID ist im ARN enthalten, zum Beispiel s-11112222333344445.

4. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/ und wählen Sie auf der Seite Server den neuen Server aus.

   Die Server-ID entspricht der ID, die für die TransferServerRessource in angezeigt wird AWS CloudFormation.

## Schritt 4: Testen Sie, ob Ihr Benutzer eine Verbindung zum Server herstellen kann

Um zu testen, ob Ihr Benutzer mithilfe der Transfer Family Family-Konsole eine Verbindung zum Server herstellen kann

1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Wählen Sie auf der Seite Server Ihren neuen Server aus, klicken Sie auf Aktionen und dann auf Test.
3. Geben Sie den Text für Ihre Anmeldeinformationen in das Feld Benutzername und in das Feld Passwort ein. Dies sind die Werte, die Sie bei der Bereitstellung des AWS CloudFormation Stacks festgelegt haben.
4. Wählen Sie für Serverprotokoll die Option SFTP aus, und geben **`127.0.0.1`** Sie für Quell-IP den Wert ein.
5. Wählen Sie Test aus.

   Wenn die Benutzerauthentifizierung erfolgreich ist, gibt der Test eine `StatusCode: 200` HTML-Antwort und ein JSON-Objekt zurück, das die Details der Rollen und Berechtigungen des Benutzers enthält. Beispielsweise:

   ```
   {
       "Response": "{\"Role\": \"arn:aws:iam::123456789012:role/my-user-role\",
   \"HomeDirectory\": \"/${transfer:HomeBucket}/\"}",
       "StatusCode": 200,
       "Message": "",
       "Url": "https://1a2b3c4d5e.execute-api.us-east-2.amazonaws.com/prod/servers/
   s-1234abcd5678efgh0/users/myuser/config"
   }
   ```

   Wenn der Test fehlschlägt, fügen Sie der Rolle, die Sie für Ihre API verwenden, eine der von API Gateway AWS verwalteten Richtlinien hinzu.

## Schritt 5: Testen Sie die SFTP-Verbindung und die Dateiübertragung

Um die SFTP-Verbindung zu testen

1. Öffnen Sie auf einem Linux- oder macOS-Gerät ein Befehlsterminal.

2. Geben Sie einen der folgenden Befehle ein, je nachdem, ob Sie ein Passwort oder ein key pair für die Authentifizierung verwenden.

   - Wenn Sie ein Passwort verwenden, geben Sie diesen Befehl ein:

     ```
     sftp -o PubkeyAuthentication=no myuser@server-ID.server.transfer.region-code.amazonaws.com
     ```

     Geben Sie bei der Aufforderung Ihr Passwort ein.

   - Wenn Sie ein key pair verwenden, geben Sie diesen Befehl ein:

     ```
     sftp -i private-key-file myuser@server-ID.server.transfer.region-code.amazonaws.com
     ```

   > **Note**
   >
   > Geben Sie für diese `sftp` Befehle den Code für den Standort AWS-Region Ihres Transfer Family Family-Servers ein. Wenn sich Ihr Server beispielsweise in USA Ost (Ohio) befindet, geben Sie ein**`us-east-2`**.

3. Stellen Sie bei der `sftp>` Aufforderung sicher, dass Sie Verzeichnisse und Dateien hochladen (`put`get), herunterladen () und anzeigen können (`pwd`und`ls`).

## Schritt 6: Beschränken Sie den Zugriff auf den Bucket

Sie können einschränken, wer auf einen bestimmten Amazon S3 S3-Bucket zugreifen kann. Das folgende Beispiel zeigt die Einstellungen, die in Ihrem CloudFormation Stack und in der Richtlinie, die Sie für Ihren Benutzer auswählen, zu verwenden sind.

In diesem Beispiel legen wir die folgenden Parameter für den AWS CloudFormation Stack fest:

- CreateServer: `true`
- UserHomeDirectory: `/myuser-bucket`
- UserName: `myuser`
- UserPassword: `MySuperSecretPassword`

> **Important**
>
> Dies ist ein Beispielpasswort. Achten Sie bei der Konfiguration Ihrer API-Gateway-Methode darauf, dass Sie ein sicheres Passwort eingeben.

- UserPublicKey1: *your-public-key*
- UserRoleArn: `arn:aws:iam::`*role-id*`:role/myuser-api-gateway-role`

Die UserPublicKey1 ist ein öffentlicher Schlüssel, den Sie als Teil eines öffentlichen/privaten key pair generiert haben.

Der *role-id* ist einzigartig für die Benutzerrolle, die Sie erstellen. Die dem beigefügte Richtlinie `myuser-api-gateway-role` lautet wie folgt:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": "s3:ListBucket",
            "Resource": "arn:aws:s3:::myuser-bucket"
        },
        {
            "Sid": "VisualEditor1",
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:GetObjectAcl",
                "s3:GetObject",
                "s3:DeleteObjectVersion",
                "s3:DeleteObject",
                "s3:PutObjectAcl",
                "s3:GetObjectVersion"
            ],
            "Resource": "arn:aws:s3:::myuser-bucket/*"
        }
    ]
}
```

Um über SFTP eine Verbindung zum Server herzustellen, geben Sie an der Eingabeaufforderung einen der folgenden Befehle ein.

- Wenn Sie ein Passwort zur Authentifizierung verwenden, führen Sie den folgenden Befehl aus:

  ```
  sftp -o PubkeyAuthentication=no myuser@transfer-server-ID.server.transfer.region-id.amazonaws.com
  ```

  Geben Sie bei der Aufforderung Ihr Passwort ein.

- Wenn Sie ein key pair zur Authentifizierung verwenden, führen Sie den folgenden Befehl aus:

  ```
  sftp -i private-key-file myuser@transfer-server-ID.server.transfer.region-id.amazonaws.com
  ```

> **Note**
>
> Verwenden Sie für diese `sftp` Befehle die ID des Ortes, AWS-Region an dem sich Ihr Transfer Family Family-Server befindet. Wenn sich Ihr Server beispielsweise in USA Ost (Ohio) befindet, verwenden Sie`us-east-2`.

An der `sftp` Eingabeaufforderung werden Sie zu Ihrem Home-Verzeichnis weitergeleitet, das Sie anzeigen können, indem Sie den pwd Befehl ausführen. Beispielsweise:

```
sftp> pwd
Remote working directory: /myuser-bucket
```

Der Benutzer kann keine Verzeichnisse oberhalb des Home-Verzeichnisses anzeigen. Beispielsweise:

```
sftp> pwd
Remote working directory: /myuser-bucket
sftp> cd ..
sftp> ls
Couldn't read directory: Permission denied
```

## Lambda aktualisieren, wenn Sie Amazon EFS verwenden

Wenn Sie Amazon EFS als Speicheroption für Ihren Transfer Family Family-Server ausgewählt haben, müssen Sie die Lambda-Funktion für Ihren Stack bearbeiten.

Um Ihrer Lambda-Funktion ein Posix-Profil hinzuzufügen

1. Öffnen Sie die Lambda-Konsole unter https://console.aws.amazon.com/lambda/.
2. Wählen Sie die Lambda-Funktion aus, die Sie zuvor erstellt haben. *Die Lambda-Funktion hat das Format **Stack-Name GetUserConfigLambda - Lambda-Identifier, wobei Stack-Name der Stack-Name und Lambda-Identifier** der Bezeichner für die CloudFormation Funktion ist.*
3. Wählen Sie auf der Registerkarte Code die Option index.js aus, um den Code für die Funktion anzuzeigen.
4. Fügen Sie im `response` die folgende Zeile zwischen `Policy` und hinzu`HomeDirectory`:

   `PosixProfile: {"Uid": `*uid-value*`, "Gid": `*gid-value*`},`

   Wobei der *UID-Wert* und der *GID-Wert* ganze Zahlen (0 oder größer) sind, die jeweils die Benutzer-ID und die Gruppen-ID darstellen.

   Nachdem Sie das Posix-Profil hinzugefügt haben, könnte das Antwortfeld beispielsweise wie folgt aussehen:

```
response = {
     Role: 'arn:aws:iam::123456789012:role/api-gateway-transfer-efs-role', // The
user will be authenticated if and only if the Role field is not blank
     Policy: '', // Optional JSON blob to further restrict this user's permissions
     PosixProfile: {"Gid": 65534, "Uid": 65534},
     HomeDirectory: '/fs-fab2c234' // Not required, defaults to '/'
   };
```

# Einrichtung einer AS2-Konfiguration

In diesem Tutorial wird beschrieben, wie Sie eine AS2-Konfiguration (Applicability Statement 2) mit einrichten. AWS Transfer Family Nachdem Sie die hier beschriebenen Schritte ausgeführt haben, verfügen Sie über einen AS2-fähigen Server, der bereit ist, AS2-Nachrichten von einem

Beispielhandelspartner anzunehmen. Sie werden auch über einen Konnektor verfügen, mit dem AS2-Nachrichten an den Beispielhandelspartner gesendet werden können.

> **Note**
>
> In einigen Teilen des Beispiel-Setups wird das AWS Command Line Interface (AWS CLI) verwendet. Falls Sie das noch nicht installiert haben AWS CLI, finden Sie weitere Informationen unter Installieren oder Aktualisieren der neuesten Version von AWS CLI im AWS Command Line Interface Benutzerhandbuch.

1. Erstellen Sie Zertifikate für sich und Ihren Handelspartner. Wenn Sie bereits Zertifikate haben, die Sie verwenden können, können Sie diesen Abschnitt überspringen.

   Dieser Vorgang wird unter beschriebenSchritt 1: Zertifikate für AS2 erstellen.

2. Erstellen Sie einen AWS Transfer Family Server, der das AS2-Protokoll verwendet. Optional können Sie dem Server eine Elastic IP-Adresse hinzufügen, sodass er mit dem Internet verbunden ist.

   Dieser Prozess wird unter beschrieben. Schritt 2: Erstellen Sie einen Transfer Family Family-Server, der das AS2-Protokoll verwendet

   > **Note**
   >
   > Sie müssen einen Transfer Family Family-Server nur für eingehende Übertragungen erstellen. Wenn Sie nur ausgehende Übertragungen durchführen, benötigen Sie keinen Transfer Family Family-Server.

3. Importieren Sie die Zertifikate, die Sie in Schritt 1 erstellt haben.

   Dieser Vorgang wird unter beschriebenSchritt 3: Zertifikate als Transfer Family Family-Zertifikatsressourcen importieren.

4. Um Ihre Handelspartner einzurichten, erstellen Sie ein lokales Profil und ein Partnerprofil.

   Dieser Vorgang wird unter beschriebenSchritt 4: Erstellen Sie Profile für Sie und Ihren Handelspartner.

5. Erstellen Sie eine Vereinbarung zwischen Ihnen und Ihrem Handelspartner.

Dieser Prozess wird unter beschriebenSchritt 5: Erstellen Sie eine Vereinbarung zwischen Ihnen und Ihrem Partner.

> **Note**
>
> Sie müssen eine Vereinbarung nur für eingehende Übertragungen erstellen. Wenn Sie nur ausgehende Überweisungen durchführen, benötigen Sie keine Vereinbarung.

6. Stellen Sie eine Verbindung zwischen Ihnen und Ihrem Handelspartner her.

   Dieser Prozess wird unter beschriebenSchritt 6: Stellen Sie eine Verbindung zwischen Ihnen und Ihrem Partner her.

   > **Note**
   >
   > Sie müssen einen Connector nur für ausgehende Übertragungen erstellen. Wenn Sie nur eingehende Übertragungen durchführen, benötigen Sie keinen Connector.

7. Testen Sie einen AS2-Dateiaustausch.

   Dieser Vorgang wird unter beschrieben. Schritt 7: Testen Sie den Austausch von Dateien über AS2 mithilfe von Transfer Family

Nachdem Sie diese Schritte abgeschlossen haben, können Sie wie folgt vorgehen:

- Senden Sie Dateien mit dem Befehl Transfer Family `start-file-transfer` AWS Command Line Interface (AWS CLI) an einen AS2-fähigen Remote-Partnerserver.
- Empfangen Sie Dateien von einem AS2-fähigen Remote-Partnerserver auf Port 5080 über Ihren Virtual Private Cloud (VPC) -Endpunkt.

## Schritt 1: Zertifikate für AS2 erstellen

Beide Parteien in einem AS2-Austausch benötigen X.509-Zertifikate. Sie können diese Zertifikate auf beliebige Weise erstellen. In diesem Thema wird beschrieben, wie Sie OpenSSL von der Befehlszeile aus verwenden, um ein Stammzertifikat zu erstellen und anschließend untergeordnete Zertifikate zu signieren. Beide Parteien müssen ihre eigenen Zertifikate generieren.

> **Note**
>
> Die Schlüssellänge für AS2-Zertifikate muss mindestens 2048 Bit und höchstens 4096 Bit betragen.

Beachten Sie Folgendes, um Dateien mit einem Partner zu übertragen:

- Sie können Zertifikate an Profile anhängen. Die Zertifikate enthalten öffentliche oder private Schlüssel.
- Ihr Handelspartner sendet Ihnen seine öffentlichen Schlüssel und Sie senden ihm Ihre.
- Ihr Handelspartner verschlüsselt Nachrichten mit Ihrem öffentlichen Schlüssel und signiert sie mit seinem privaten Schlüssel. Umgekehrt verschlüsseln Sie Nachrichten mit dem öffentlichen Schlüssel Ihres Partners und signieren sie mit Ihrem privaten Schlüssel.

> **Note**
>
> Wenn Sie es vorziehen, Schlüssel mit einer GUI zu verwalten, Portecleist dies eine Option, die Sie verwenden können.

Um Beispielzertifikate zu generieren

> **Important**
>
> Senden Sie Ihrem Partner nicht Ihre privaten Schlüssel. In diesem Beispiel generieren Sie einen Satz selbstsignierter öffentlicher und privater Schlüssel für eine Partei. Wenn Sie zu Testzwecken als beide Geschäftspartner agieren möchten, können Sie diese Anweisungen wiederholen, um zwei Schlüsselsätze zu generieren: einen für jeden Geschäftspartner. In diesem Fall müssen Sie nicht zwei Root-Zertifizierungsstellen (CAs) generieren.

1. Führen Sie den folgenden Befehl aus, um einen privaten RSA-Schlüssel mit einem 2048-Bit-Modul zu generieren.

```
/usr/bin/openssl genrsa -out root-ca-key.pem 2048
```

2. Führen Sie den folgenden Befehl aus, um ein selbstsigniertes Zertifikat mit Ihrer Datei zu erstellen. `root-ca-key.pem`

```
/usr/bin/openssl req \
-x509 -new -nodes -sha256 \
-days 1825 \
-subj "/C=US/ST=MA/L=Boston/O=TransferFamilyCustomer/OU=IT-dept/CN=ROOTCA" \
-key root-ca-key.pem \
-out root-ca.pem
```

Das `-subj` Argument besteht aus den folgenden Werten.

| | Name | Beschreibung |
|---|---|---|
| C | Ländercode | Ein aus zwei Buchstaben bestehender Code für das Land, in dem Ihre Organisation ansässig ist. |
| ST | Bundesland, Region oder Provinz | Das Bundesland, die Region oder die Provinz, in der Ihre Organisation ansässig ist. (In diesem Fall bezieht sich Region nicht auf Ihre AWS-Region.) |
| L | Ortsname | Die Stadt, in der sich Ihre Organisation befindet. |
| O | Name der Organisation | Der vollständige offizielle Name Ihrer Organisation, einschließlich Suffixen wie LLC, Corp usw. |
| OU | Name der Organisationseinheit | Die Abteilung in Ihrer Organisation, die sich mit diesem Zertifikat befasst. |

| | Name | Beschreibung |
|---|---|---|
| CN | Allgemeiner Name oder vollqualifizierter Domänenname (FQDN) | In diesem Fall erstellen wir ein Stammzertifikat, der Wert ist `ROOTCA` also. In diesen Beispielen beschreiben wir den Zweck des Zertifikats. `CN` |

3. Erstellen Sie einen Signaturschlüssel und einen Verschlüsselungsschlüssel für Ihr lokales Profil.

```
/usr/bin/openssl genrsa -out signing-key.pem 2048
/usr/bin/openssl genrsa -out encryption-key.pem 2048
```

> **Note**
>
> Einige AS2-fähige Server, wie OpenAS2, erfordern, dass Sie dasselbe Zertifikat sowohl für das Signieren als auch für die Verschlüsselung verwenden. In diesem Fall können Sie denselben privaten Schlüssel und dasselbe Zertifikat für beide Zwecke importieren. Führen Sie dazu diesen Befehl anstelle der beiden vorherigen Befehle aus:
>
> ```
> /usr/bin/openssl genrsa -out signing-and-encryption-key.pem 2048
> ```

4. Führen Sie die folgenden Befehle aus, um Certificate Signing Requests (CSRs) für den zu signierenden Stammschlüssel zu erstellen.

```
/usr/bin/openssl req -new -key signing-key.pem -subj \
"/C=US/ST=MA/L=Boston/O=TransferFamilyCustomer/OU=IT-dept/CN=Signer" -out signing-
key-csr.pem
```

```
/usr/bin/openssl req -new -key encryption-key.pem -subj \
"/C=US/ST=MA/L=Boston/O=TransferFamilyCustomer/OU=IT-dept/CN=Encrypter" -out
 encryption-key-csr.pem
```

5. Als Nächstes müssen Sie eine `signing-cert.conf` Datei und eine `encryption-cert.conf` Datei erstellen.

- Verwenden Sie einen Texteditor, um die `signing-cert.conf` Datei mit dem folgenden Inhalt zu erstellen:

```
authorityKeyIdentifier=keyid,issuer
keyUsage = digitalSignature, nonRepudiation
```

- Verwenden Sie einen Texteditor, um die `encryption-cert.conf` Datei mit dem folgenden Inhalt zu erstellen:

```
authorityKeyIdentifier=keyid,issuer
keyUsage = dataEncipherment
```

6. Schließlich erstellen Sie die signierten Zertifikate, indem Sie die folgenden Befehle ausführen.

```
/usr/bin/openssl x509 -req -sha256 -CAcreateserial -days 1825 -in signing-key-
csr.pem -out signing-cert.pem -CA \
root-ca.pem -CAkey root-ca-key.pem -extfile signing-cert.conf
```

```
/usr/bin/openssl x509 -req -sha256 -CAcreateserial -days 1825 -in encryption-key-
csr.pem -out encryption-cert.pem \
-CA root-ca.pem -CAkey root-ca-key.pem -extfile encryption-cert.conf
```

## Schritt 2: Erstellen Sie einen Transfer Family Family-Server, der das AS2-Protokoll verwendet

In diesem Verfahren wird erklärt, wie Sie mithilfe der Transfer Family einen AS2-fähigen Server erstellen. AWS CLI

> **Note**
>
> In vielen der Beispielschritte werden Befehle verwendet, mit denen Parameter aus einer Datei geladen werden. Weitere Informationen zur Verwendung von Dateien zum Laden von Parametern finden Sie unter So laden Sie Parameter aus einer Datei.

Wenn Sie stattdessen die Konsole verwenden möchten, finden Sie weitere Informationen unterErstellen Sie einen AS2-Server mit der Transfer Family Family-Konsole.

Ähnlich wie beim Erstellen eines SFTP- oder AWS Transfer Family FTPS-Servers erstellen Sie einen AS2-fähigen Server mithilfe des `--protocols AS2` Befehlsparameters. `create-server` AWS CLI Derzeit unterstützt Transfer Family nur VPC-Endpunkttypen und Amazon S3 S3-Speicher mit dem AS2-Protokoll.

Wenn Sie Ihren AS2-fähigen Server für Transfer Family mithilfe des `create-server` Befehls erstellen, wird automatisch ein VPC-Endpunkt für Sie erstellt. Dieser Endpunkt macht den TCP-Port 5080 verfügbar, sodass er AS2-Nachrichten akzeptieren kann.

Wenn Sie Ihren VPC-Endpunkt öffentlich dem Internet zugänglich machen möchten, können Sie Elastic IP-Adressen mit Ihrem VPC-Endpunkt verknüpfen.

Um diese Anweisungen verwenden zu können, benötigen Sie Folgendes:

- Die ID Ihrer VPC (z. B. vpc-abcdef01).
- Die IDs Ihrer VPC-Subnetze (z. B. subnet-abcdef01, subnet-subnet-abcdef01, subnet-021345ab).
- Eine oder mehrere IDs der Sicherheitsgruppen, die eingehenden Datenverkehr über den TCP-Port 5080 von Ihren Handelspartnern zulassen (z. B. sg-1234567890abcdef0 und sg-abcdef01234567890).
- (Optional) Die Elastic IP-Adressen, die Sie Ihrem VPC-Endpunkt zuordnen möchten.
- Wenn Ihr Handelspartner nicht über ein VPN mit Ihrer VPC verbunden ist, benötigen Sie ein Internet-Gateway. Weitere Informationen finden Sie unter Verbinden mit dem Internet über ein Internet-Gateway im Amazon-VPC-Benutzerhandbuch.

Um einen AS2-fähigen Server zu erstellen

1. Führen Sie den folgenden Befehl aus. Ersetzen Sie jeden *`user input placeholder`* durch Ihre Informationen.

```
aws transfer create-server --endpoint-type VPC \
--endpoint-details VpcId=vpc-abcdef01,SubnetIds=subnet-abcdef01,subnet-
abcdef01,subnet-
021345ab,SecurityGroupIds=sg-abcdef01234567890,sg-1234567890abcdef0 --protocols AS2
 \
--protocol-details As2Transports=HTTP
```

2. (Optional) Sie können den VPC-Endpunkt öffentlich machen. Sie können Elastic IP-Adressen nur über einen `update-server` Vorgang an einen Transfer Family Family-Server anhängen. Die

folgenden Befehle stoppen den Server, aktualisieren ihn mit Elastic IP-Adressen und starten ihn dann erneut.

```
aws transfer stop-server --server-id your-server-id
```

```
aws transfer update-server --server-id your-server-id --endpoint-details \
AddressAllocationIds=eipalloc-abcdef01234567890,eipalloc-
1234567890abcdef0,eipalloc-abcd012345cccccc
```

```
aws transfer start-server --server-id your-server-id
```

Dieser `start-server` Befehl erstellt automatisch einen DNS-Eintrag für Sie, der die öffentliche IP-Adresse für Ihren Server enthält. Um Ihrem Handelspartner Zugriff auf den Server zu gewähren, stellen Sie ihm die folgenden Informationen zur Verfügung. *your-region*Bezieht sich in diesem Fall auf Ihre AWS-Region.

`s-`*your-server-id*`.server.transfer.`*your-region*`.amazonaws.com`

Die vollständige URL, die Sie Ihrem Handelspartner zur Verfügung stellen, lautet wie folgt:

`http://s-`*your-server-id*`.server.transfer.`*your-region*`.amazonaws.com:5080`

3. Verwenden Sie die folgenden Befehle, um zu testen, ob auf Ihren AS2-fähigen Server zugegriffen werden kann. Stellen Sie sicher, dass auf Ihren Server entweder über die private DNS-Adresse Ihres VPC-Endpunkts oder über Ihren öffentlichen Endpunkt (falls Sie Ihrem Endpunkt eine Elastic IP-Adresse zugeordnet haben) zugegriffen werden kann.

   Wenn Ihr Server korrekt konfiguriert ist, ist die Verbindung erfolgreich. Sie erhalten jedoch eine Antwort mit dem HTTP-Statuscode 400 (Bad Request), da Sie keine gültige AS2-Nachricht senden.

   - Führen Sie für einen öffentlichen Endpunkt (wenn Sie im vorherigen Schritt eine Elastic IP-Adresse zugewiesen haben) den folgenden Befehl aus und ersetzen Sie dabei Ihre Server-ID und Region.

     ```
     curl -vv -X POST http://s-your-server-id.transfer.your-region.amazonaws.com:5080
     ```

   - Wenn Sie eine Verbindung innerhalb Ihrer VPC herstellen, suchen Sie den privaten DNS-Namen Ihres VPC-Endpunkts, indem Sie die folgenden Befehle ausführen.

```
aws transfer describe-server --server-id s-your-server-id
```

Dieser `describe-server` Befehl gibt Ihre VPC-Endpunkt-ID im `VpcEndpointId` Parameter zurück. Verwenden Sie diesen Wert, um den folgenden Befehl auszuführen.

```
aws ec2 describe-vpc-endpoints --vpc-endpoint-ids vpce-your-vpc-endpoint-id
```

Dieser `describe-vpc-endpoints` Befehl gibt ein `DNSEntries` Array mit mehreren `DnsName` Parametern zurück. Verwenden Sie im folgenden Befehl den regionalen DNS-Namen (den Namen, der die Availability Zone nicht enthält).

```
curl -vv -X POST http://vpce-your-vpce.vpce-svc-your-vpce-svc.your-
region.vpce.amazonaws.com:5080
```

Der folgende Befehl zeigt beispielsweise Beispielwerte für die Platzhalter im vorherigen Befehl.

```
curl -vv -X POST http://vpce-0123456789abcdefg-fghij123.vpce-
svc-11111aaaa2222bbbb.us-east-1.vpce.amazonaws.com:5080
```

4. (Optional) Konfigurieren Sie eine Protokollierungsrolle. Transfer Family protokolliert den Status von gesendeten und empfangenen Nachrichten in einem strukturierten JSON-Format in CloudWatch Amazon-Protokollen. Um Transfer Family Zugriff auf die CloudWatch Protokolle in Ihrem Konto zu gewähren, müssen Sie eine Protokollierungsrolle auf Ihrem Server konfigurieren.

   Erstellen Sie eine `transfer.amazonaws.com` vertrauenswürdige AWS Identity and Access Management (IAM-) Rolle und fügen Sie die `AWSTransferLoggingAccess` verwaltete Richtlinie hinzu. Details hierzu finden Sie unter Erstellen Sie eine IAM-Rolle und -Richtlinie. Notieren Sie sich den Amazon-Ressourcennamen (ARN) der IAM-Rolle, die Sie gerade erstellt haben, und verknüpfen Sie ihn mit dem Server, indem Sie den folgenden `update-server` Befehl ausführen:

   ```
   aws transfer update-server --server-id your-server-id --logging-role
    arn:aws:iam::your-account-id:role/logging-role-name
   ```

> **Note**
>
> Obwohl die Logging-Rolle optional ist, empfehlen wir dringend, sie so einzurichten, dass Sie den Status Ihrer Nachrichten einsehen und Konfigurationsprobleme beheben können.

## Schritt 3: Zertifikate als Transfer Family Family-Zertifikatsressourcen importieren

In diesem Verfahren wird erklärt, wie Zertifikate mithilfe von importiert AWS CLI werden. Wenn Sie stattdessen die Transfer Family Family-Konsole verwenden möchten, finden Sie weitere Informationen unterthe section called “AS2-Zertifikate importieren”.

Führen Sie die folgenden `import-certificate` Befehle aus, um die Signatur- und Verschlüsselungszertifikate zu importieren, die Sie in Schritt 1 erstellt haben. Wenn Sie dasselbe Zertifikat für Verschlüsselung und Signierung verwenden, importieren Sie dasselbe Zertifikat zweimal (einmal mit der `SIGNING` Verwendung und erneut mit der `ENCRYPTION` Verwendung).

```
aws transfer import-certificate --usage SIGNING --certificate file://signing-cert.pem \
            --private-key file://signing-key.pem --certificate-chain file://root-ca.pem
```

Dieser Befehl gibt Ihre Signatur zurück`CertificateId`. Im nächsten Abschnitt wird diese Zertifikat-ID als bezeichnet*`my-signing-cert-id`*.

```
aws transfer import-certificate --usage ENCRYPTION --certificate file://encryption-
cert.pem \
            --private-key file://encryption-key.pem --certificate-chain file://root-
ca.pem
```

Dieser Befehl gibt Ihre Verschlüsselung zurück`CertificateId`. Im nächsten Abschnitt wird diese Zertifikat-ID als bezeichnet*`my-encrypt-cert-id`*.

Importieren Sie als Nächstes die Verschlüsselungs- und Signaturzertifikate Ihres Partners, indem Sie die folgenden Befehle ausführen.

```
aws transfer import-certificate --usage ENCRYPTION --certificate file://partner-
encryption-cert.pem \
```

```
--certificate-chain file://partner-root-ca.pem
```

Dieser Befehl gibt die Verschlüsselung Ihres Partners zurück`CertificateId`. Im nächsten Abschnitt wird diese Zertifikat-ID als bezeichnet*partner-encrypt-cert-id*.

```
aws transfer import-certificate --usage SIGNING --certificate file://partner-signing-cert.pem \
--certificate-chain file://partner-root-ca.pem
```

Dieser Befehl gibt die Signatur Ihres Partners zurück`CertificateId`. Im nächsten Abschnitt wird diese Zertifikat-ID als bezeichnet*partner-signing-cert-id*.

## Schritt 4: Erstellen Sie Profile für Sie und Ihren Handelspartner

In diesem Verfahren wird erklärt, wie Sie AS2-Profile mithilfe AWS CLI von erstellen. Wenn Sie stattdessen die Transfer Family Family-Konsole verwenden möchten, finden Sie weitere Informationen unterthe section called “Erstellen Sie AS2-Profile”.

Erstellen Sie Ihr lokales AS2-Profil, indem Sie den folgenden Befehl ausführen. Dieser Befehl verweist auf die Zertifikate, die Ihre öffentlichen und privaten Schlüssel enthalten.

```
aws transfer create-profile --as2-id MYCORP --profile-type LOCAL --certificate-ids \
my-signing-cert-id my-encrypt-cert-id
```

Dieser Befehl gibt Ihre Profil-ID zurück. Im nächsten Abschnitt wird diese ID als bezeichnet*my-profile-id*.

Erstellen Sie nun das Partnerprofil, indem Sie den folgenden Befehl ausführen. Dieser Befehl verwendet nur die Public-Key-Zertifikate Ihres Partners. Um diesen Befehl zu verwenden, ersetzen Sie das *user input placeholders* durch Ihre eigenen Informationen, z. B. den AS2-Namen und die Zertifikat-IDs Ihres Partners.

```
aws transfer create-profile --as2-id PARTNER-COMPANY --profile-type PARTNER --
certificate-ids \
partner-signing-cert-id partner-encrypt-cert-id
```

Dieser Befehl gibt die Profil-ID Ihres Partners zurück. Im nächsten Abschnitt wird diese ID als bezeichnet*partner-profile-id*.

> **Note**
>
> Ersetzen Sie in den vorherigen Befehlen *MYCORP* durch den Namen Ihrer Organisation und *PARTNER-COMPANY* durch den Namen der Organisation Ihres Handelspartners.

## Schritt 5: Erstellen Sie eine Vereinbarung zwischen Ihnen und Ihrem Partner

In diesem Verfahren wird erklärt, wie Sie AS2-Vereinbarungen mithilfe von erstellen. AWS CLI Wenn Sie stattdessen die Transfer Family Family-Konsole verwenden möchten, finden Sie weitere Informationen unterthe section called “Erstellen Sie AS2-Vereinbarungen”.

In Vereinbarungen werden die beiden Profile (lokal und Partner), ihre Zertifikate und eine Serverkonfiguration zusammengeführt, die eingehende AS2-Übertragungen zwischen zwei Parteien ermöglicht. Sie können Ihre Artikel auflisten, indem Sie die folgenden Befehle ausführen.

```
aws transfer list-profiles --profile-type LOCAL
aws transfer list-profiles --profile-type PARTNER
aws transfer list-servers
```

Für diesen Schritt sind ein Amazon S3 S3-Bucket und eine IAM-Rolle mit Lese-/Schreibzugriff auf und vom Bucket erforderlich. Die Anweisungen zum Erstellen dieser Rolle entsprechen denen für die SFTP-, FTP- und FTPS-Protokolle der Transfer Family und sind unter verfügbar. Erstellen Sie eine IAM-Rolle und -Richtlinie

Um eine Vereinbarung zu erstellen, benötigen Sie die folgenden Elemente:

- Der Amazon S3 S3-Bucket-Name (und das Objektpräfix, falls angegeben)
- Der ARN der IAM-Rolle mit Zugriff auf den Bucket
- Ihre Transfer Family Family-Server-ID
- Ihre Profil-ID und die Profil-ID Ihres Partners

Erstellen Sie die Vereinbarung, indem Sie den folgenden Befehl ausführen.

```
aws transfer create-agreement --description "ExampleAgreementName" --server-id your-
server-id \
```

```
--local-profile-id your-profile-id --partner-profile-id your-partner-profile-id --base-
directory /DOC-EXAMPLE-DESTINATION-BUCKET/AS2-inbox \
--access-role arn:aws:iam::111111111111:role/TransferAS2AccessRole
```

Bei Erfolg gibt dieser Befehl die ID für die Vereinbarung zurück. Sie können dann die Details der Vereinbarung mit dem folgenden Befehl anzeigen.

```
aws transfer describe-agreement --agreement-id agreement-id --server-id your-server-id
```

## Schritt 6: Stellen Sie eine Verbindung zwischen Ihnen und Ihrem Partner her

In diesem Verfahren wird erklärt, wie Sie AS2-Konnektoren mithilfe von erstellen. AWS CLI Wenn Sie stattdessen die Transfer Family Family-Konsole verwenden möchten, finden Sie weitere Informationen unterthe section called “AS2-Konnektoren konfigurieren”.

Sie können den `StartFileTransfer` API-Vorgang verwenden, um Dateien, die in Amazon S3 gespeichert sind, mithilfe eines Connectors an den AS2-Endpunkt Ihres Handelspartners zu senden. Sie können die zuvor erstellten Profile finden, indem Sie den folgenden Befehl ausführen.

```
aws transfer list-profiles
```

Wenn Sie den Connector erstellen, müssen Sie die AS2-Server-URL Ihres Partners angeben. Kopieren Sie den folgenden Text in eine Datei mit dem Namen`testAS2Config.json`.

```
{
"Compression": "ZLIB",
"EncryptionAlgorithm": "AES256_CBC",
"LocalProfileId": "your-profile-id",
"MdnResponse": "SYNC",
"MdnSigningAlgorithm": "DEFAULT",
"MessageSubject": "Your Message Subject",
"PartnerProfileId": "partner-profile-id",
"SigningAlgorithm": "SHA256"
}
```

> **Note**
>
> Geben Sie den `DES_EDE3_CBC` Algorithmus nur an`EncryptionAlgorithm`, wenn Sie einen älteren Client unterstützen müssen, der ihn benötigt, da es sich um einen schwachen Verschlüsselungsalgorithmus handelt.

Führen Sie dann den folgenden Befehl aus, um den Connector zu erstellen.

```
aws transfer create-connector --url "http://partner-as2-server-url" \
--access-role your-IAM-role-for-bucket-access \
--logging-role arn:aws:iam::your-account-id:role/service-role/AWSTransferLoggingAccess
 \
--as2-config file:///path/to/testAS2Config.json
```

## Schritt 7: Testen Sie den Austausch von Dateien über AS2 mithilfe von Transfer Family

### Erhalten Sie eine Datei von Ihrem Handelspartner

Wenn Sie Ihrem VPC-Endpunkt eine öffentliche Elastic IP-Adresse zugeordnet haben, hat Transfer Family automatisch einen DNS-Namen erstellt, der Ihre öffentliche IP-Adresse enthält. Die Subdomain ist Ihre AWS Transfer Family Server-ID (im Formats`-1234567890abcdef0`). Geben Sie Ihrem Handelspartner Ihre Server-URL im folgenden Format an.

```
http://s-1234567890abcdef0.server.transfer.us-east-1.amazonaws.com:5080
```

Wenn Sie Ihrem VPC-Endpunkt keine öffentliche Elastic IP-Adresse zugeordnet haben, suchen Sie nach dem Hostnamen des VPC-Endpunkts, der AS2-Nachrichten über HTTP POST von Ihren Handelspartnern auf Port 5080 annehmen kann. Verwenden Sie den folgenden Befehl, um die VPC-Endpunktdetails abzurufen.

```
aws transfer describe-server --server-id s-1234567890abcdef0
```

Nehmen wir beispielsweise an, der vorherige Befehl gibt die VPC-Endpunkt-ID von `vpce-1234abcd5678efghi` zurück. Dann würden Sie den folgenden Befehl verwenden, um die DNS-Namen abzurufen.

```
aws ec2 describe-vpc-endpoints --vpc-endpoint-ids vpce-1234abcd5678efghi
```

Dieser Befehl gibt alle Details für den VPC-Endpunkt zurück, die Sie zum Ausführen des folgenden Befehls benötigen.

Der DNS-Name ist im `DnsEntries` Array aufgeführt. Ihr Handelspartner muss sich in Ihrer VPC befinden, um auf Ihren VPC-Endpunkt zugreifen zu können (z. B. über AWS PrivateLink oder ein VPN). Geben Sie Ihrem Partner Ihre VPC-Endpunkt-URL im folgenden Format an.

```
http://vpce-your-vpce-id.vpce-svc-your-vpce-svc-id.your-region.vpce.amazonaws.com:5080
```

Die folgende URL zeigt beispielsweise Beispielwerte für die Platzhalter in den vorherigen Befehlen.

```
http://vpce-0123456789abcdefg-fghij123.vpce-svc-11111aaaa2222bbbb.us-
east-1.vpce.amazonaws.com:5080
```

In diesem Beispiel werden erfolgreiche Übertragungen an dem Speicherort gespeichert, der in dem von Ihnen angegebenen `base-directory` Parameter angegeben ist. Schritt 5: Erstellen Sie eine Vereinbarung zwischen Ihnen und Ihrem Partner Wenn wir erfolgreich Dateien mit dem Namen `myfile1.txt` und empfangen`myfile2.txt`, werden die Dateien unter gespeichert`/path-defined-in-the-agreement/processed/original_filename.messageId.original_extension`. Hier werden die Dateien als `/DOC-EXAMPLE-DESTINATION-BUCKET/AS2-inbox/processed/myfile1.messageId.txt` und gespeichert`/DOC-EXAMPLE-DESTINATION-BUCKET/AS2-inbox/processed/myfile2.messageId.txt`.

Wenn Sie bei der Erstellung Ihres Transfer Family Family-Servers eine Protokollierungsrolle konfiguriert haben, können Sie Ihre CloudWatch Protokolle auch auf den Status von AS2-Nachrichten überprüfen.

## Senden Sie eine Datei an Ihren Handelspartner

Sie können Transfer Family verwenden, um AS2-Nachrichten zu senden, indem Sie auf die Connector-ID und die Pfade zu den Dateien verweisen, wie im folgenden Befehl `start-file-transfer` AWS Command Line Interface (AWS CLI) dargestellt:

```
aws transfer start-file-transfer --connector-id c-1234567890abcdef0 \
```

```
--send-file-paths "/DOC-EXAMPLE-SOURCE-BUCKET/myfile1.txt" "/DOC-EXAMPLE-SOURCE-BUCKET/
myfile2.txt"
```

Führen Sie den folgenden Befehl aus, um die Details für Ihre Konnektoren abzurufen:

```
aws transfer list-connectors
```

Der `list-connectors` Befehl gibt die Connector-IDs, URLs und Amazon-Ressourcennamen (ARNs) für Ihre Konnektoren zurück.

Um die Eigenschaften eines bestimmten Connectors zurückzugeben, führen Sie den folgenden Befehl mit der ID aus, die Sie verwenden möchten:

```
aws transfer describe-connector --connector-id your-connector-id
```

Der `describe-connector` Befehl gibt alle Eigenschaften für den Connector zurück, einschließlich seiner URL, Rollen, Profile, MDNs (Message Disposition Notices), Tags und Überwachungsmetriken.

Sie können überprüfen, ob der Partner die Dateien erfolgreich erhalten hat, indem Sie sich die JSON- und MDN-Dateien ansehen. Diese Dateien werden gemäß den unter beschriebenen Konventionen benannt. Dateinamen und Speicherorte Wenn Sie bei der Erstellung des Connectors eine Protokollierungsrolle konfiguriert haben, können Sie Ihre CloudWatch Protokolle auch auf den Status von AS2-Nachrichten überprüfen.

# Konfiguration eines SFTP-, FTPS- oder FTP-Serverendpunkts

Dieses Thema enthält Einzelheiten zum Erstellen und Verwenden von AWS Transfer Family Serverendpunkten, die eines oder mehrere der Protokolle SFTP, FTPS und FTP verwenden.

Themen


- Optionen für den Identitätsanbieter
- AWS Transfer Family Endpunkt-Typmatrix
- Konfiguration eines SFTP-, FTPS- oder FTP-Serverendpunkts
- Übertragung von Dateien über einen Serverendpunkt mit einem Client
- Verwalten von Benutzern für Serverendpunkte
- Verwendung logischer Verzeichnisse zur Vereinfachung Ihrer Transfer Family Family-Verzeichnisstrukturen


## Optionen für den Identitätsanbieter

AWS Transfer Family bietet verschiedene Methoden zur Authentifizierung und Verwaltung von Benutzern. In der folgenden Tabelle werden die verfügbaren Identitätsanbieter verglichen, die Sie mit Transfer Family verwenden können.

| Aktion | AWS Transfer Family Dienst verwaltet | AWS Managed Microsoft AD | Amazon API Gateway | AWS Lambda |
|---|---|---|---|---|
| Unterstützte Protokolle | SFTP | SFTP, FTPS, FTP | SFTP, FTPS, FTP | SFTP, FTPS, FTP |
| Schlüssel basierte Authentifizierung | Ja | Nein | Ja | Ja |
| Passworta uthentifizierung | Nein | Ja | Ja | Ja |

| Aktion | AWS Transfer Family Dienst verwaltet | AWS Managed Microsoft AD | Amazon API Gateway | AWS Lambda |
|---|---|---|---|---|
| AWS Identity and Access Managemen t (IAM) und POSIX | Ja | Ja | Ja | Ja |
| Logisches Home-Verz eichnis | Ja | Ja | Ja | Ja |
| Parametrisierter Zugriff (benutzer namenbasiert) | Ja | Ja | Ja | Ja |
| Ad-hoc-Zu griffsstruktur | Ja | Nein | Ja | Ja |
| AWS WAF | Nein | Nein | Ja | Nein |

Hinweise:

- IAM wird verwendet, um den Zugriff auf den Amazon S3 S3-Backing-Speicher zu kontrollieren, und POSIX wird für Amazon EFS verwendet.
- Ad-Hoc bezieht sich auf die Fähigkeit, das Benutzerprofil zur Laufzeit zu senden. Sie können beispielsweise Benutzer in ihren Home-Verzeichnissen landen lassen, indem Sie den Benutzernamen als Variable übergeben.
- Einzelheiten dazu finden AWS WAF Sie unterFügen Sie eine Firewall für Webanwendungen hinzu.
- In einem Blogbeitrag wird die Verwendung einer in Microsoft Azure AD integrierten Lambda-Funktion als Identitätsanbieter für Transfer Family beschrieben. Einzelheiten finden Sie unter Authentifizierung AWS Transfer Family mit Azure Active Directory und. AWS Lambda
- Wir bieten verschiedene AWS CloudFormation Vorlagen, mit denen Sie schnell einen Transfer Family Family-Server bereitstellen können, der einen benutzerdefinierten Identitätsanbieter verwendet. Details hierzu finden Sie unter Lambda-Funktionsvorlagen.

Mit den folgenden Verfahren können Sie einen SFTP-fähigen Server, einen FTPS-fähigen Server, einen FTP-fähigen Server oder einen AS2-fähigen Server erstellen.

Nächster Schritt

- Erstellen Sie einen SFTP-fähigen Server
- Erstellen Sie einen FTPS-fähigen Server
- Erstellen Sie einen FTP-fähigen Server
- AS2 konfigurieren

# AWS Transfer Family Endpunkt-Typmatrix

Wenn Sie einen Transfer Family Family-Server erstellen, wählen Sie den zu verwendenden Endpunkttyp aus. In der folgenden Tabelle werden die Merkmale der einzelnen Endpunkttypen beschrieben.

Matrix der Endpunkttypen

| Merkmal | Öffentlich | VPC - Internet | VPC — Intern | VPC_Endpoint (veraltet) |
|---|---|---|---|---|
| Unterstützte Protokolle | SFTP | SFTP, FTPS, AS2 | SFTP, FTP, FTPS, AS2 | SFTP |
| Access | Über das Internet. Für diesen Endpunkttyp ist keine spezielle Konfiguration in Ihrer VPC erforderlich. | Über das Internet und innerhalb von VPC- und VPC-verbundenen Umgebungen, z. B. einem lokalen Rechenzentrum über oder VPN. AWS Direct Connect | Aus VPC- und VPC-verbundenen Umgebungen, z. B. einem lokalen Rechenzentrum über oder VPN. AWS Direct Connect | Aus VPC- und VPC-verbundenen Umgebungen, z. B. einem lokalen Rechenzentrum über oder VPN. AWS Direct Connect |
| Statische IP-Adresse | Sie können keine statische | Sie können Elastic IP- | Private IP-Adressen, die an | Private IP-Adressen, die an |

| Merkmal | Öffentlich | VPC - Internet | VPC — Intern | VPC_Endpoint (veraltet) |
|---|---|---|---|---|
| | IP-Adresse anhängen. AWS stellt IP-Adressen bereit, die sich ändern können. | Adressen an den Endpunkt anhängen. Dies können AWS eigene IP-Adressen oder Ihre eigenen IP-Adressen sein (Bringen Sie Ihre eigenen IP-Adressen mit). Elastische IP-Adressen, die an den Endpunkt angehängt sind, ändern sich nicht.<br><br>Private IP-Adressen, die an den Server angeschlossen sind, ändern sich ebenfalls nicht. | den Endpunkt angeschlossen sind, ändern sich nicht. | den Endpunkt angeschlossen sind, ändern sich nicht. |

| Merkmal | Öffentlich | VPC - Internet | VPC — Intern | VPC_Endpoint (veraltet) |
|---|---|---|---|---|
| Liste der zugelassenen Quell-IP-Adressen | Dieser Endpunkttyp unterstützt keine Zulassungslisten nach Quell-IP-Adressen.<br><br>Der Endpunkt ist öffentlich zugänglich und überwacht den Datenverkehr über Port 22.<br><br>**Note**<br>Für VPC-gehostete Endpunkte können Server der SFTP Transfer Family über Port 22 (Standard), Port 2222 oder Port 22000 | Um den Zugriff über die Quell-IP-Adresse zu ermöglichen, können Sie Sicherheitsgruppen verwenden, die an die Serverendpunkte angehängt sind, und Netzwerk-ACLs, die an das Subnetz angeschlossen sind, in dem sich der Endpunkt befindet. | Um den Zugriff über die Quell-IP-Adresse zu ermöglichen, können Sie Sicherheitsgruppen verwenden, die an die Serverendpunkte angehängt sind, und Netzwerkzugriffskontrolllisten (Netzwerk-ACLs), die an das Subnetz angehängt sind, in dem sich der Endpunkt befindet. | Um den Zugriff über die Quell-IP-Adresse zu ermöglichen, können Sie Sicherheitsgruppen verwenden, die an die Serverendpunkte angehängt sind, und Netzwerk-ACLs, die an das Subnetz angeschlossen sind, in dem sich der Endpunkt befindet. |

| Merkmal | Öffentlich | VPC - Internet | VPC — Intern | VPC_Endpoint (veraltet) |
|---|---|---|---|---|
| | betrieben werden. | | | |
| Liste der zugelassenen Client-Firewalls | Sie müssen den DNS-Namen des Servers zulassen.<br><br>Da sich IP-Adressen ändern können, sollten Sie es vermeiden, IP-Adressen für Ihre Client-Firewall-Zulassungsliste zu verwenden. | Sie können den DNS-Namen des Servers oder die mit dem Server verbundenen Elastic IP-Adressen zulassen. | Sie können die privaten IP-Adressen oder den DNS-Namen der Endpoints zulassen. | Sie können die privaten IP-Adressen oder den DNS-Namen der Endpunkte zulassen. |

> **Note**
>
> Der `VPC_ENDPOINT` Endpunkttyp ist jetzt veraltet und kann nicht zum Erstellen neuer Server verwendet werden. Verwenden Sie statt dessen den neuen VPC-Endpunkttyp (`EndpointType=VPC`), den Sie entweder als Intern - oder Internetanschluss verwenden können, wie in der vorherigen Tabelle beschrieben. `EndpointType=VPC_ENDPOINT` Details hierzu finden Sie unter Einstellung der Verwendung von VPC_ENDPOINT.

Ziehen Sie die folgenden Optionen in Betracht, um die Sicherheit Ihres AWS Transfer Family Servers zu verbessern:

- Verwenden Sie einen VPC-Endpunkt mit internem Zugriff, sodass der Server nur für Clients in Ihrer VPC oder VPC-verbundenen Umgebungen wie einem lokalen Rechenzentrum über oder VPN zugänglich ist. AWS Direct Connect

- Verwenden Sie einen VPC-Endpunkt mit Internetzugriff, um Clients den Zugriff auf den Endpunkt über das Internet zu ermöglichen und Ihren Server zu schützen. Ändern Sie anschließend die Sicherheitsgruppen der VPC so, dass nur Datenverkehr von bestimmten IP-Adressen zugelassen wird, auf denen die Clients Ihrer Benutzer gehostet werden.
- Wenn Sie eine kennwortbasierte Authentifizierung benötigen und einen benutzerdefinierten Identitätsanbieter für Ihren Server verwenden, ist es eine bewährte Methode, dass Ihre Passwortrichtlinie verhindert, dass Benutzer schwache Passwörter erstellen, und dass die Anzahl fehlgeschlagener Anmeldeversuche begrenzt wird.
- AWS Transfer Family ist ein verwalteter Dienst und bietet daher keinen Shell-Zugriff. Sie können nicht direkt auf den zugrunde liegenden SFTP-Server zugreifen, um betriebssystemeigene Befehle auf Transfer Family Family-Servern auszuführen.
- Verwenden Sie einen Network Load Balancer vor einem VPC-Endpunkt mit internem Zugriff. Ändern Sie den Listener-Port auf dem Load Balancer von Port 22 auf einen anderen Port. Dadurch kann das Risiko, dass Portscanner und Bots Ihren Server untersuchen, verringern, aber nicht ausschließen, da Port 22 am häufigsten zum Scannen verwendet wird. Einzelheiten finden Sie im Blogbeitrag Network Load Balancers now support Security Groups.

> **Note**
>
> Wenn Sie einen Network Load Balancer verwenden, zeigen die AWS Transfer Family CloudWatch Protokolle die IP-Adresse für den NLB und nicht die tatsächliche Client-IP-Adresse.

## Konfiguration eines SFTP-, FTPS- oder FTP-Serverendpunkts

Mithilfe des Dienstes können Sie einen Dateiübertragungsserver erstellen. AWS Transfer Family Die folgenden Dateiübertragungsprotokolle sind verfügbar:

- Secure Shell (SSH) File Transfer Protocol (SFTP) — Dateiübertragung über SSH. Details hierzu finden Sie unter the section called “Erstellen Sie einen SFTP-fähigen Server”.

> **Note**
>
> Wir bieten ein AWS CDK Beispiel für die Erstellung eines SFTP Transfer Family Family-Servers. Das Beispiel verwendet TypeScript und ist GitHub hier verfügbar.

- File Transfer Protocol Secure (FTPS) — Dateiübertragung mit TLS-Verschlüsselung. Details hierzu finden Sie unter the section called “Erstellen Sie einen FTPS-fähigen Server”.
- File Transfer Protocol (FTP) — Unverschlüsselte Dateiübertragung. Details hierzu finden Sie unter the section called “Erstellen Sie einen FTP-fähigen Server”.
- Anwendbarkeitserklärung 2 (AS2) — Dateiübertragung für den Transport strukturierter Daten. business-to-business Details hierzu finden Sie unter the section called “AS2 konfigurieren”. Für AS2 können Sie schnell einen AWS CloudFormation Stack zu Demonstrationszwecken erstellen. Dieses Verfahren wird unter beschrieben. Verwenden Sie eine Vorlage, um einen Demo-Stack der Transfer Family AS2 zu erstellen

Sie können einen Server mit mehreren Protokollen erstellen.

> **Note**
>
> Wenn Sie mehrere Protokolle für denselben Serverendpunkt aktiviert haben und den Zugriff über mehrere Protokolle mit demselben Benutzernamen ermöglichen möchten, können Sie dies tun, sofern die für das Protokoll spezifischen Anmeldeinformationen in Ihrem Identitätsanbieter eingerichtet wurden. Für FTP empfehlen wir, separate Anmeldeinformationen von SFTP und FTPS zu verwenden. Dies liegt daran, dass FTP im Gegensatz zu SFTP und FTPS Anmeldeinformationen im Klartext überträgt. Durch die Isolierung von FTP-Anmeldeinformationen von SFTP oder FTPS bleiben Ihre Workloads, die SFTP oder FTPS verwenden, sicher, wenn FTP-Anmeldeinformationen geteilt oder offengelegt werden.

Wenn Sie einen Server erstellen, wählen Sie einen bestimmten Server aus, AWS-Region um die Dateioperationsanforderungen von Benutzern auszuführen, die diesem Server zugewiesen sind. Sie weisen dem Server nicht nur ein oder mehrere Protokolle zu, sondern weisen auch einen der folgenden Identitätsanbietertypen zu:

- Dienst, der mithilfe von SSH-Schlüsseln verwaltet wird. Details hierzu finden Sie unter Arbeiten mit serviceverwalteten Benutzern.
- AWS Directory Service for Microsoft Active Directory (AWS Managed Microsoft AD). Mit dieser Methode können Sie Ihre Microsoft Active Directory-Gruppen integrieren, um Zugriff auf Ihre Transfer Family Family-Server zu gewähren. Details hierzu finden Sie unter Verwenden des AWS Directory Service-Identitätsanbieters.

- Eine benutzerdefinierte Methode. Die benutzerdefinierte Identitätsanbieter-Methode verwendet unser AWS Lambda Amazon API Gateway und ermöglicht es Ihnen, Ihren Verzeichnisdienst zu integrieren, um Ihre Benutzer zu authentifizieren und zu autorisieren. Der Service weist automatisch einen Identifier zu, der den Server eindeutig identifiziert. Details hierzu finden Sie unter Mit Anbietern benutzerdefinierter Identitäten arbeiten. Transfer Family bietet AWS CloudFormation Vorlagen, mit denen Sie schnell Server bereitstellen können, die einen benutzerdefinierten Identitätsanbieter verwenden.
  - Lambda-Funktionen für die Authentifizierungbeschreibt CloudFormation Vorlagen, die eine Lambda-Funktion zur Authentifizierung verwenden.
  - Authentifizierung mit einer API-Gateway-Methodebeschreibt CloudFormation Vorlagen, die eine Amazon API Gateway Gateway-Methode zur Authentifizierung verwenden.

Sie weisen dem Server auch einen Endpunkttyp (öffentlich zugänglich oder von VPC gehostet) und einen Hostnamen zu, indem Sie den Standard-Serverendpunkt oder einen benutzerdefinierten Hostnamen verwenden, indem Sie den Amazon Route 53-Service oder einen Domain Name System (DNS) -Service Ihrer Wahl verwenden. Ein Server-Hostname muss in dem Ort, an AWS-Region dem er erstellt wurde, eindeutig sein.

Darüber hinaus können Sie eine CloudWatch Amazon-Protokollierungsrolle zuweisen, um Ereignisse in Ihre CloudWatch Protokolle zu übertragen, eine Sicherheitsrichtlinie wählen, die die kryptografischen Algorithmen enthält, die für die Verwendung durch Ihren Server aktiviert sind, und dem Server Metadaten in Form von Tags hinzufügen, die Schlüssel-Wert-Paare sind.

> **Important**
>
> Es fallen Kosten für instanziierte Server und für die Datenübertragung an. Informationen zur Preisgestaltung und zur Schätzung der Kosten für die Nutzung von Transfer Family finden Sie unter AWS Transfer Family Preise. AWS Pricing Calculator

## Erstellen Sie einen SFTP-fähigen Server

Das Secure Shell (SSH) File Transfer Protocol (SFTP) ist ein Netzwerkprotokoll, das für die sichere Übertragung von Daten über das Internet verwendet wird. Das Protokoll unterstützt die volle Sicherheits- und Authentifizierungsfunktionalität von SSH. Es wird häufig für den Austausch von Daten, einschließlich sensibler Informationen, zwischen Geschäftspartnern in einer Vielzahl von Branchen wie Finanzdienstleistungen, Gesundheitswesen, Einzelhandel und Werbung verwendet.

> **Note**
>
> SFTP-Server für Transfer Family arbeiten über Port 22. Für VPC-gehostete Endpunkte können die Server der SFTP Transfer Family auch über Port 2222 oder Port 22000 betrieben werden. Details hierzu finden Sie unter Erstellen Sie einen Server in einer virtuellen privaten Cloud.

Informationen finden Sie auch unter:

- Wir bieten ein AWS CDK Beispiel für die Erstellung eines SFTP Transfer Family Family-Servers. Das Beispiel verwendet TypeScript und ist GitHub hier verfügbar.
- Eine exemplarische Vorgehensweise zur Bereitstellung eines Transfer Family Family-Servers in einer VPC finden Sie unter Verwenden Sie die IP-Zulassungsliste, um Ihre AWS Transfer Family Server zu sichern.

So erstellen Sie einen SFTP-fähigen Server

1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/ und wählen Sie im Navigationsbereich Server und anschließend Server erstellen aus.
2. Wählen Sie unter Protokolle auswählen die Option SFTP und dann Weiter aus.
3. Wählen Sie unter Wählen Sie einen Identitätsanbieter aus den Identitätsanbieter aus, den Sie für die Verwaltung des Benutzerzugriffs verwenden möchten. Ihnen stehen folgende Optionen zur Verfügung:

   - Service verwaltet — Sie speichern Benutzeridentitäten und Schlüssel in AWS Transfer Family.
   - AWS Directory Service for Microsoft Active Directory— Sie stellen ein AWS Directory Service Verzeichnis für den Zugriff auf den Endpunkt bereit. Auf diese Weise können Sie die in Ihrem Active Directory gespeicherten Anmeldeinformationen verwenden, um Ihre Benutzer zu authentifizieren. Weitere Informationen zur Arbeit mit AWS Managed Microsoft AD Identitätsanbietern finden Sie unterVerwenden des AWS Directory Service-Identitätsanbieters.

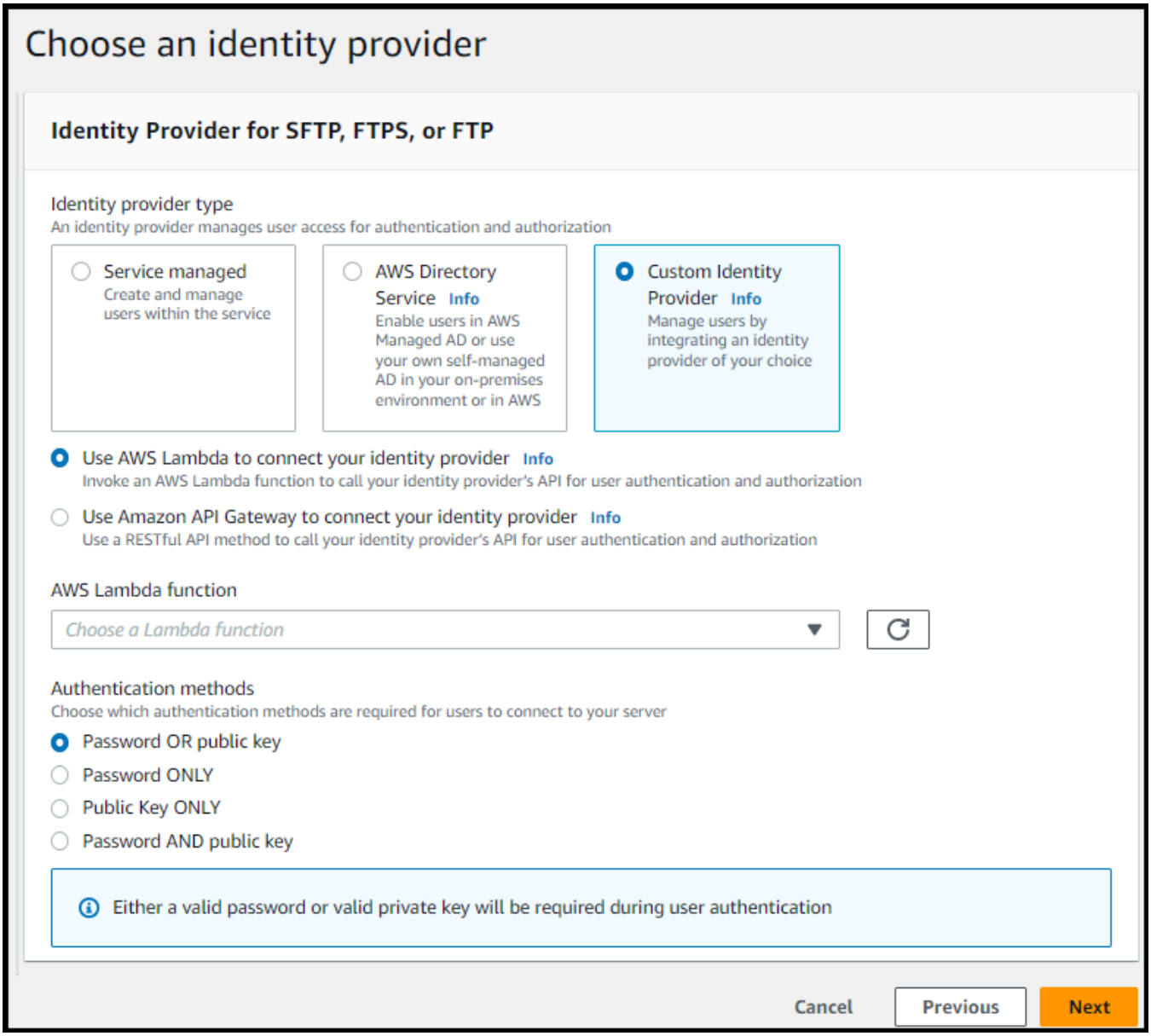
     > **Note**
     >
     > - Kontoübergreifende Verzeichnisse und gemeinsam genutzte Verzeichnisse werden für AWS Managed Microsoft AD nicht unterstützt.

- Um einen Server mit Directory Service als Identitätsanbieter einzurichten, müssen Sie einige AWS Directory Service Berechtigungen hinzufügen. Details hierzu finden Sie unter Bevor Sie mit der Verwendung von beginnen AWS Directory Service for Microsoft Active Directory.

- Benutzerdefinierter Identitätsanbieter — Wählen Sie eine der folgenden Optionen:
  - Verwenden Sie AWS Lambda , um Ihren Identitätsanbieter zu verbinden — Sie können einen vorhandenen Identitätsanbieter verwenden, der von einer Lambda-Funktion unterstützt wird. Sie geben den Namen der Lambda-Funktion an. Weitere Informationen finden Sie unter Wird AWS Lambda zur Integration Ihres Identitätsanbieters verwendet.
  - Verwenden Sie Amazon API Gateway, um Ihren Identitätsanbieter zu verbinden — Sie können eine API Gateway Gateway-Methode erstellen, die von einer Lambda-Funktion unterstützt wird, um sie als Identitätsanbieter zu verwenden. Sie geben eine Amazon API Gateway Gateway-URL und eine Aufruf-Rolle an. Weitere Informationen finden Sie unter Verwenden Sie Amazon API Gateway zur Integration Ihres Identitätsanbieters.

  Für beide Optionen können Sie auch angeben, wie Sie sich authentifizieren möchten.

  - Passwort ODER Schlüssel — Benutzer können sich entweder mit ihrem Passwort oder ihrem Schlüssel authentifizieren. Dies ist der Standardwert.
  - NUR Passwort — Benutzer müssen ihr Passwort angeben, um eine Verbindung herzustellen.
  - NUR Schlüssel — Benutzer müssen ihren privaten Schlüssel angeben, um eine Verbindung herzustellen.
  - Passwort UND Schlüssel — Benutzer müssen sowohl ihren privaten Schlüssel als auch ihr Passwort angeben, um eine Verbindung herzustellen. Der Server überprüft zuerst den Schlüssel. Wenn der Schlüssel gültig ist, fordert das System dann zur Eingabe eines Kennworts auf. Wenn der angegebene private Schlüssel nicht mit dem gespeicherten öffentlichen Schlüssel übereinstimmt, schlägt die Authentifizierung fehl.




4. Wählen Sie Weiter aus.
5. Gehen Sie unter Endpunkt auswählen wie folgt vor:

   a. Wählen Sie als Endpunkttyp den Typ Öffentlich zugänglicher Endpunkt aus. Informationen zu einem VPC-gehosteten Endpunkt finden Sie unterErstellen Sie einen Server in einer virtuellen privaten Cloud.

   b. (Optional) Wählen Sie für Benutzerdefinierter Hostname die Option Keine aus.

      Sie erhalten einen Server-Hostnamen, der von bereitgestellt wird. AWS Transfer Family Der Server-Host-Name hat das Format *serverId*`.server.transfer.`*regionId*`.amazonaws.com`.

Für einen benutzerdefinierten Hostnamen geben Sie einen benutzerdefinierten Alias für Ihren Serverendpunkt an. Weitere Informationen zum Arbeiten mit benutzerdefinierten Hostnamen finden Sie unter. Mit benutzerdefinierten Hostnamen arbeiten

c. (Optional) Aktivieren Sie für FIPS Enabled das Kontrollkästchen FIPS-fähiger Endpunkt, um sicherzustellen, dass der Endpunkt den Federal Information Processing Standards (FIPS) entspricht.

> **Note**
>
> FIPS-fähige Endpunkte sind nur in nordamerikanischen Regionen verfügbar. AWS Informationen zu den verfügbaren Regionen finden Sie unter AWS Transfer Family Endpunkte und Kontingente in der. Allgemeine AWS-Referenz Weitere Informationen zu FIPS finden Sie unter Federal Information Processing Standard (FIPS) 140-2.

d. Wählen Sie Weiter aus.

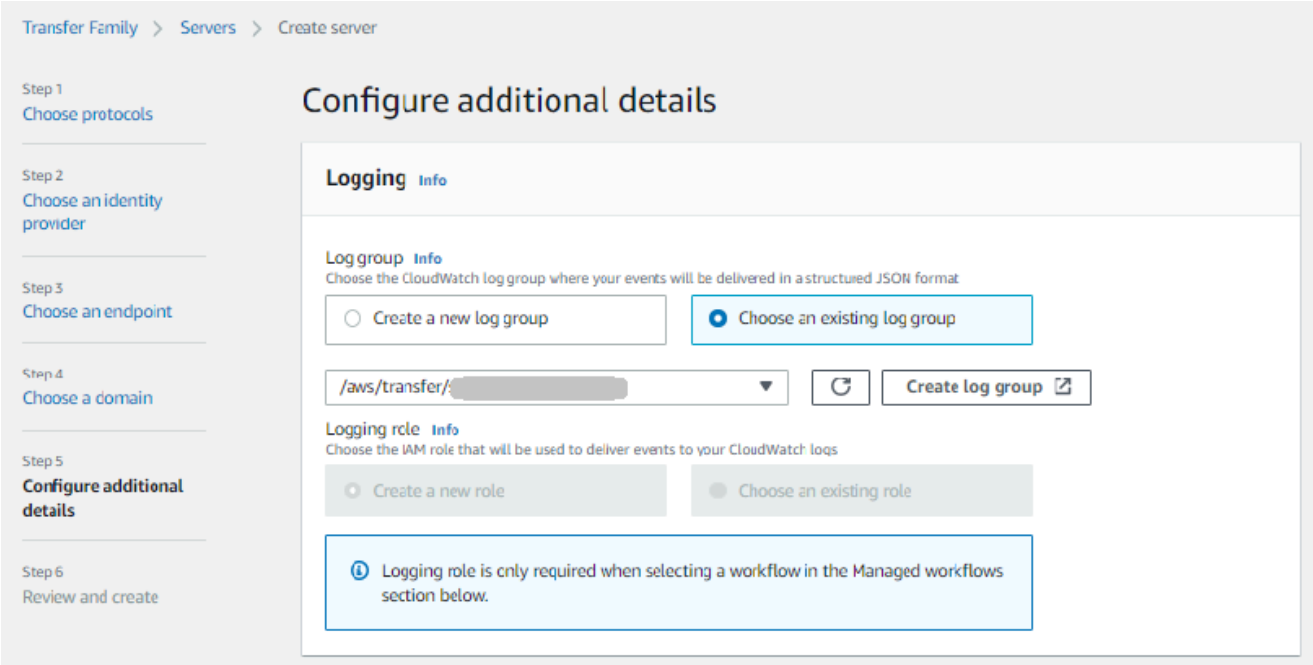
6. Wählen Sie auf der Seite Domain auswählen den AWS Speicherdienst aus, den Sie zum Speichern und Zugreifen auf Ihre Daten über das ausgewählte Protokoll verwenden möchten:

   - Wählen Sie Amazon S3, um Ihre Dateien als Objekte über das ausgewählte Protokoll zu speichern und darauf zuzugreifen.
   - Wählen Sie Amazon EFS, um Ihre Dateien in Ihrem Amazon EFS-Dateisystem über das ausgewählte Protokoll zu speichern und darauf zuzugreifen.

   Wählen Sie Weiter aus.

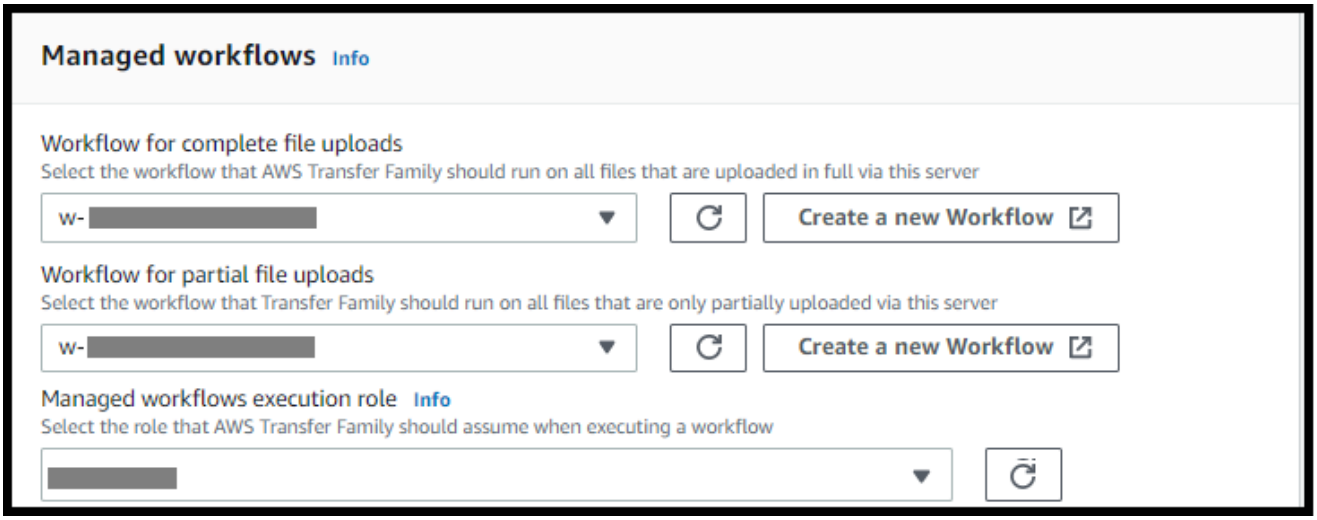
7. Gehen Sie unter Zusätzliche Details konfigurieren wie folgt vor:

   a. Geben Sie für die Protokollierung eine bestehende Protokollgruppe an oder erstellen Sie eine neue (Standardoption). Wenn Sie eine bestehende Protokollgruppe wählen, müssen Sie eine auswählen, die mit Ihrer verknüpft ist AWS-Konto.




Wenn Sie „Protokollgruppe erstellen“ wählen, wird in der CloudWatch Konsole (https://console.aws.amazon.com/cloudwatch/) die Seite „Protokollgruppe erstellen“ geöffnet. Einzelheiten finden Sie unter Protokollgruppe erstellen in CloudWatch Logs.

b. (Optional) Wählen Sie für verwaltete Workflows Workflow-IDs (und eine entsprechende Rolle) aus, die Transfer Family bei der Ausführung des Workflows annehmen soll. Sie können einen Workflow auswählen, der bei einem vollständigen Upload ausgeführt wird, und einen anderen, der bei einem teilweisen Upload ausgeführt werden soll. Weitere Informationen zur Verarbeitung Ihrer Dateien mithilfe verwalteter Workflows finden Sie unterAWS Transfer Family verwaltete Workflows.



c. Wählen Sie unter Optionen für kryptografische Algorithmen eine Sicherheitsrichtlinie aus, die die kryptografischen Algorithmen enthält, die für die Verwendung durch Ihren Server aktiviert sind. Unsere neueste Sicherheitsrichtlinie ist die Standardeinstellung: Einzelheiten finden Sie unter. Sicherheitsrichtlinien für AWS Transfer Family Server

d. (Optional) Geben Sie für Server Host Key einen privaten RSA-, ED25519- oder ECDSA-Schlüssel ein, der zur Identifizierung Ihres Servers verwendet wird, wenn Clients über SFTP eine Verbindung zu ihm herstellen. Sie können auch eine Beschreibung hinzufügen, um zwischen mehreren Hostschlüsseln zu unterscheiden.

   Nachdem Sie Ihren Server erstellt haben, können Sie weitere Hostschlüssel hinzufügen. Mehrere Hostschlüssel sind nützlich, wenn Sie Schlüssel rotieren möchten oder wenn Sie verschiedene Schlüsseltypen verwenden möchten, z. B. einen RSA-Schlüssel und auch einen ECDSA-Schlüssel.

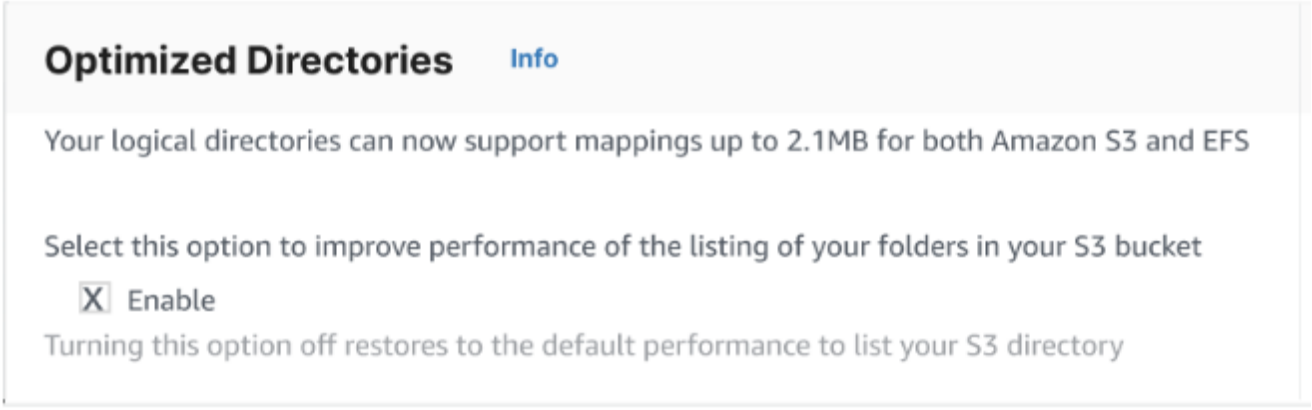
   > **Note**
   >
   > Der Abschnitt Server-Hostschlüssel wird nur für die Migration von Benutzern von einem vorhandenen SFTP-fähigen Server verwendet.

e. (Optional) Geben Sie für Tags für Schlüssel und Wert ein oder mehrere Tags als Schlüssel-Wert-Paare ein und wählen Sie dann Tag hinzufügen aus.

f. Wählen Sie Weiter aus.

g. Sie können die Leistung Ihrer Amazon S3 S3-Verzeichnisse optimieren. Nehmen wir zum Beispiel an, Sie gehen in Ihr Home-Verzeichnis und haben 10.000 Unterverzeichnisse. Mit anderen Worten, Ihr Amazon S3 S3-Bucket hat 10.000 Ordner. Wenn Sie in diesem Szenario den Befehl `ls` (list) ausführen, dauert der Listenvorgang zwischen sechs und acht Minuten. Wenn Sie Ihre Verzeichnisse optimieren, dauert dieser Vorgang jedoch nur wenige Sekunden.

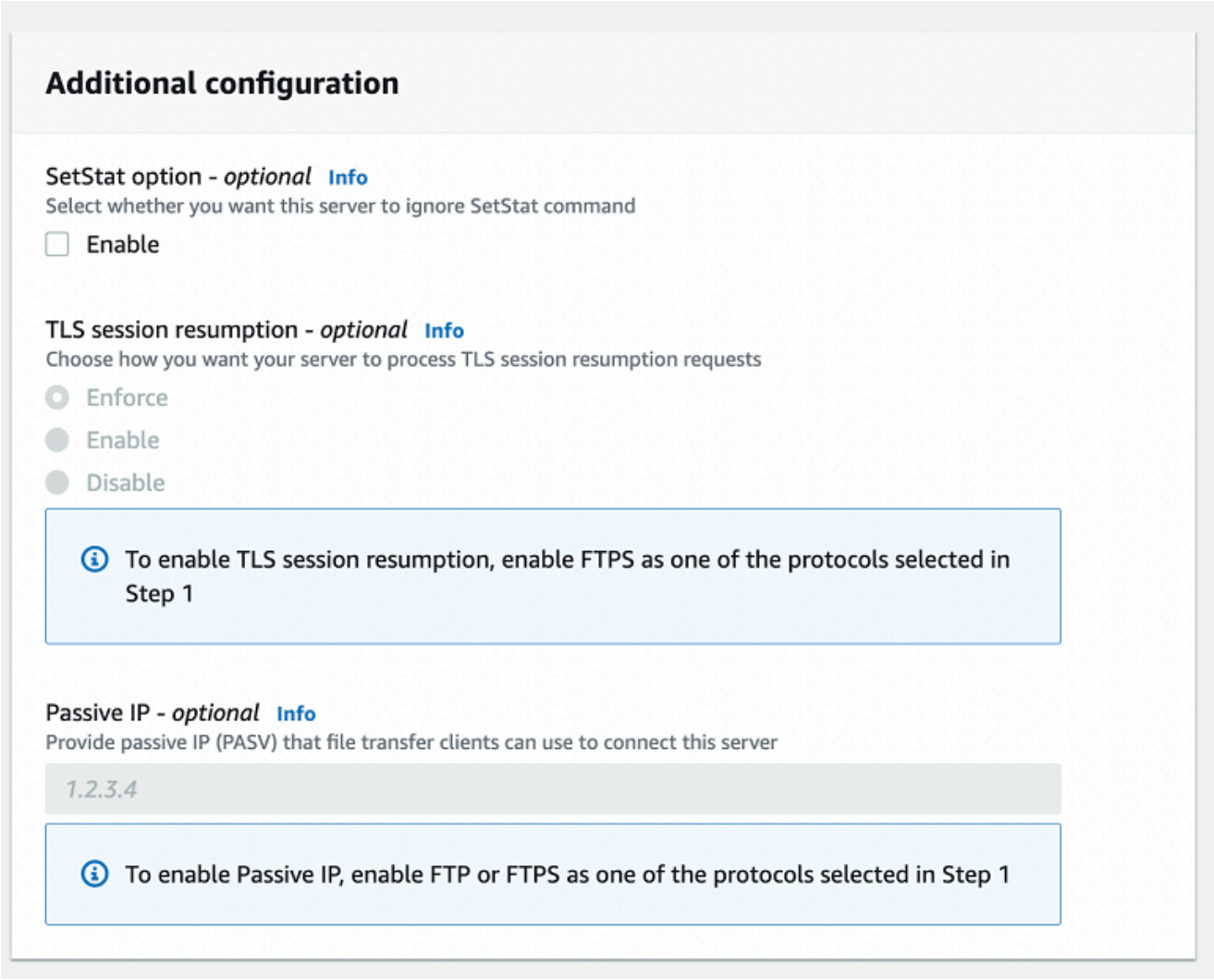
   Wenn Sie Ihren Server mit der Konsole erstellen, sind optimierte Verzeichnisse standardmäßig aktiviert. Wenn Sie Ihren Server mithilfe der API erstellen, ist dieses Verhalten standardmäßig nicht aktiviert.




h. (Optional) Konfigurieren Sie AWS Transfer Family Server so, dass Ihren Endbenutzern benutzerdefinierte Nachrichten wie Unternehmensrichtlinien oder Nutzungsbedingungen angezeigt werden. Geben Sie für Banner anzeigen in das Textfeld Banner für die Vorauthentifizierung die Textnachricht ein, die Sie Ihren Benutzern vor der Authentifizierung anzeigen möchten.

i. (Optional) Sie können die folgenden zusätzlichen Optionen konfigurieren.

- SetStat Option: Aktivieren Sie diese Option, um den Fehler zu ignorieren, der generiert wird, wenn ein Client versucht, eine Datei `SETSTAT` zu verwenden, die Sie in einen Amazon S3 S3-Bucket hochladen. Weitere Informationen finden Sie in der `SetStatOption` Dokumentation im ProtocolDetails.
- Wiederaufnahme der TLS-Sitzung: Diese Option ist nur verfügbar, wenn Sie FTPS als eines der Protokolle für diesen Server aktiviert haben.
- Passive IP: Diese Option ist nur verfügbar, wenn Sie FTPS oder FTP als eines der Protokolle für diesen Server aktiviert haben.




8. Überprüfen Sie unter Überprüfen und erstellen Ihre Auswahl.

   - Wenn Sie eine davon bearbeiten möchten, wählen Sie neben dem Schritt Bearbeiten aus.

     > **Note**
     >
     > Sie müssen jeden Schritt nach dem Schritt überprüfen, den Sie bearbeiten möchten.

   - Wenn Sie keine Änderungen vorgenommen haben, wählen Sie Server erstellen, um Ihren Server zu erstellen. Sie gelangen zur Seite Servers (Server) (siehe unten), auf der der neue Server aufgelistet ist.

Es kann einige Minuten dauern, bis sich der Status Ihres neuen Servers auf Online ändert. Zu diesem Zeitpunkt kann Ihr Server Dateioperationen ausführen, aber Sie müssen zuerst einen Benutzer erstellen. Einzelheiten zum Erstellen von Benutzern finden Sie unterVerwalten von Benutzern für Serverendpunkte.

## Erstellen Sie einen FTPS-fähigen Server

Das File Transfer Protocol over SSL (FTPS) ist eine Erweiterung von FTP. Es verwendet die kryptografischen Protokolle Transport Layer Security (TLS) und Secure Sockets Layer (SSL) zur Verschlüsselung des Datenverkehrs. FTPS ermöglicht die gleichzeitige oder unabhängige Verschlüsselung sowohl der Kontroll- als auch der Datenkanalverbindungen.

Um einen FTPS-fähigen Server zu erstellen

1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/ und wählen Sie im Navigationsbereich Server und anschließend Server erstellen aus.
2. Wählen Sie unter Protokolle auswählen die Option FTPS aus.

   Wählen Sie unter Serverzertifikat ein in AWS Certificate Manager (ACM) gespeichertes Zertifikat aus, das zur Identifizierung Ihres Servers verwendet wird, wenn Clients über FTPS eine Verbindung zu ihm herstellen, und wählen Sie dann Weiter aus.

   Informationen zum Anfordern eines neuen öffentlichen Zertifikats finden Sie unter Anfordern eines öffentlichen Zertifikats im AWS Certificate Manager Benutzerhandbuch.

   Informationen zum Importieren eines vorhandenen Zertifikats in ACM finden Sie unter Zertifikate in ACM importieren im AWS Certificate Manager Benutzerhandbuch.

   Informationen zum Anfordern eines privaten Zertifikats für die Verwendung von FTPS über private IP-Adressen finden Sie unter Anfordern eines privaten Zertifikats im AWS Certificate Manager Benutzerhandbuch.

   Zertifikate mit den folgenden kryptografischen Algorithmen und Schlüsselgrößen werden unterstützt:

   - 2048-Bit-RSA (RSA_2048)
   - 4096-Bit-RSA (RSA_4096)
   - Elliptic Prime Curve 256-Bit (EC_prime256v1)
   - Elliptic Prime Curve 384-Bit (EC_secp384r1)

- Elliptic Prime Curve 521-Bit (EC_secp521r1)

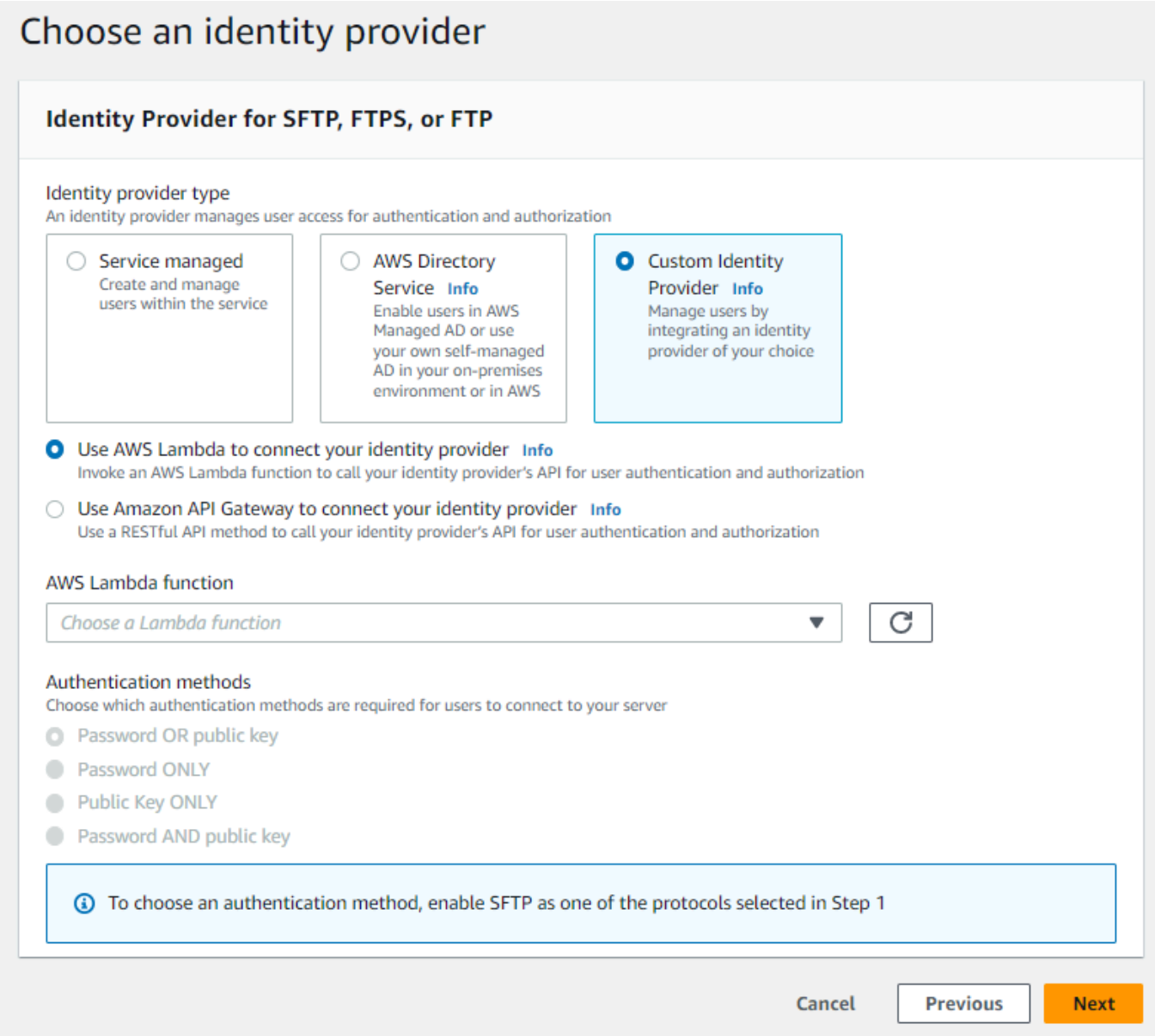
> **Note**
>
> Das Zertifikat muss ein gültiges SSL/TLS X.509 Version 3-Zertifikat mit angegebenem FQDN oder IP-Adresse sein und Informationen über den Aussteller enthalten.

3. Wählen Sie unter Identitätsanbieter auswählen den Identitätsanbieter aus, den Sie für die Verwaltung des Benutzerzugriffs verwenden möchten. Ihnen stehen folgende Optionen zur Verfügung:

   - AWS Directory Service for Microsoft Active Directory— Sie stellen ein AWS Directory Service Verzeichnis für den Zugriff auf den Endpunkt bereit. Auf diese Weise können Sie die in Ihrem Active Directory gespeicherten Anmeldeinformationen verwenden, um Ihre Benutzer zu authentifizieren. Weitere Informationen zur Arbeit mit AWS Managed Microsoft AD Identitätsanbietern finden Sie unterVerwenden des AWS Directory Service-Identitätsanbieters.

     > **Note**
     >
     > - Kontoübergreifende Verzeichnisse und gemeinsam genutzte Verzeichnisse werden für AWS Managed Microsoft AD nicht unterstützt.
     > - Um einen Server mit Directory Service als Identitätsanbieter einzurichten, müssen Sie einige AWS Directory Service Berechtigungen hinzufügen. Details hierzu finden Sie unter Bevor Sie mit der Verwendung von beginnen AWS Directory Service for Microsoft Active Directory.

   - Benutzerdefinierter Identitätsanbieter — Wählen Sie eine der folgenden Optionen:
     - Verwenden Sie AWS Lambda , um Ihren Identitätsanbieter zu verbinden — Sie können einen vorhandenen Identitätsanbieter verwenden, der von einer Lambda-Funktion unterstützt wird. Sie geben den Namen der Lambda-Funktion an. Weitere Informationen finden Sie unter Wird AWS Lambda zur Integration Ihres Identitätsanbieters verwendet.
     - Verwenden Sie Amazon API Gateway, um Ihren Identitätsanbieter zu verbinden — Sie können eine API Gateway Gateway-Methode erstellen, die von einer Lambda-Funktion unterstützt wird, um sie als Identitätsanbieter zu verwenden. Sie geben eine Amazon API Gateway Gateway-URL und eine Aufruf-Rolle an. Weitere Informationen finden Sie unter Verwenden Sie Amazon API Gateway zur Integration Ihres Identitätsanbieters.




4. Wählen Sie Weiter aus.
5. Gehen Sie unter Endpunkt auswählen wie folgt vor:

> **Note**
>
> FTPS-Server für Transfer Family arbeiten über Port 21 (Control Channel) und Port Range 8192—8200 (Datenkanal).

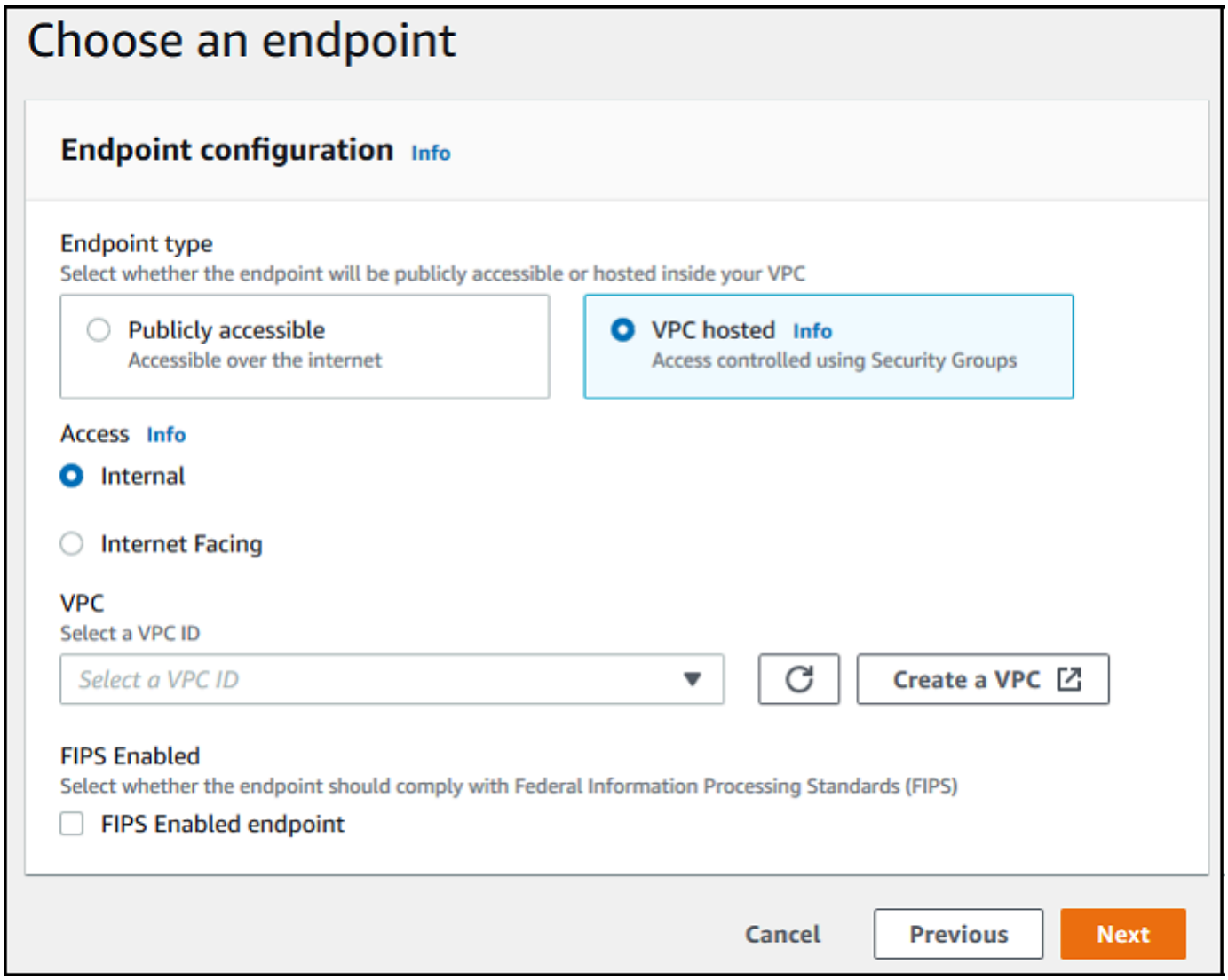
a. Wählen Sie als Endpunkttyp den VPC-gehosteten Endpunkttyp aus, um den Endpunkt Ihres Servers zu hosten. Informationen zur Einrichtung Ihres VPC-gehosteten Endpunkts finden Sie unterErstellen Sie einen Server in einer virtuellen privaten Cloud.

> **Note**
>
> Öffentlich zugängliche Endpunkte werden nicht unterstützt.

b. (Optional) Aktivieren Sie für FIPS Enabled das Kontrollkästchen FIPS-fähiger Endpunkt, um sicherzustellen, dass der Endpunkt den Federal Information Processing Standards (FIPS) entspricht.

> **Note**
>
> FIPS-fähige Endpunkte sind nur in nordamerikanischen Regionen verfügbar. AWS Informationen zu den verfügbaren Regionen finden Sie unter AWS Transfer Family Endpunkte und Kontingente in der. Allgemeine AWS-Referenz Weitere Informationen zu FIPS finden Sie unter Federal Information Processing Standard (FIPS) 140-2.

c. Wählen Sie Weiter aus.



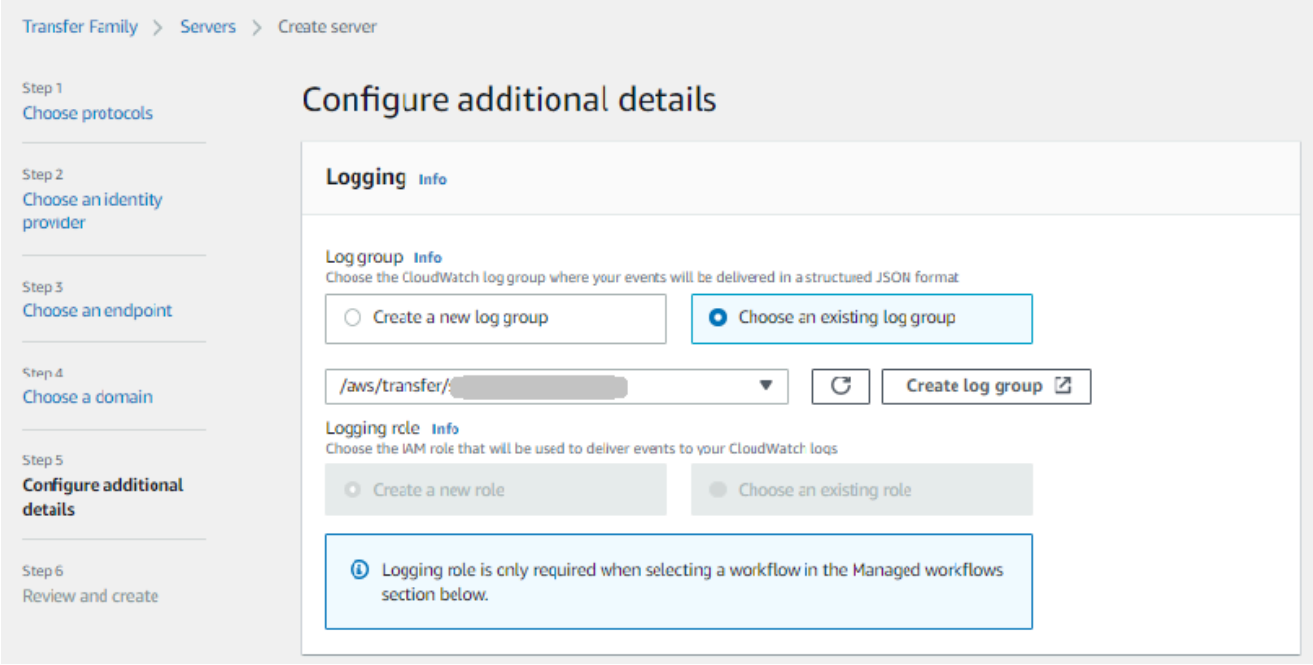

6. Wählen Sie auf der Seite „Domain auswählen“ den AWS Speicherdienst aus, den Sie zum Speichern und Zugreifen auf Ihre Daten über das ausgewählte Protokoll verwenden möchten:

   - Wählen Sie Amazon S3, um Ihre Dateien als Objekte über das ausgewählte Protokoll zu speichern und darauf zuzugreifen.
   - Wählen Sie Amazon EFS, um Ihre Dateien in Ihrem Amazon EFS-Dateisystem über das ausgewählte Protokoll zu speichern und darauf zuzugreifen.

   Wählen Sie Weiter aus.

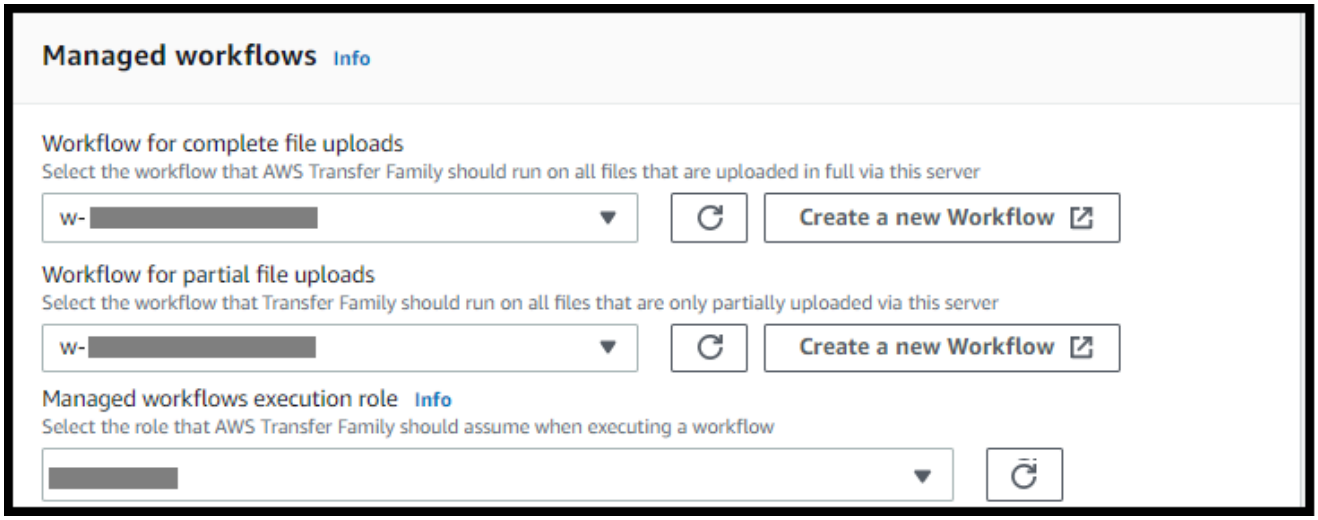
7. Gehen Sie unter Zusätzliche Details konfigurieren wie folgt vor:

a. Geben Sie für die Protokollierung eine bestehende Protokollgruppe an oder erstellen Sie eine neue (Standardoption).




Wenn Sie „Protokollgruppe erstellen“ wählen, wird in der CloudWatch Konsole (https://console.aws.amazon.com/cloudwatch/) die Seite „Protokollgruppe erstellen“ geöffnet. Einzelheiten finden Sie unter Protokollgruppe erstellen in CloudWatch Logs.

b. (Optional) Wählen Sie für verwaltete Workflows Workflow-IDs (und eine entsprechende Rolle) aus, die Transfer Family bei der Ausführung des Workflows annehmen soll. Sie können einen Workflow auswählen, der bei einem vollständigen Upload ausgeführt wird, und einen anderen, der bei einem teilweisen Upload ausgeführt werden soll. Weitere Informationen zur Verarbeitung Ihrer Dateien mithilfe verwalteter Workflows finden Sie unterAWS Transfer Family verwaltete Workflows.



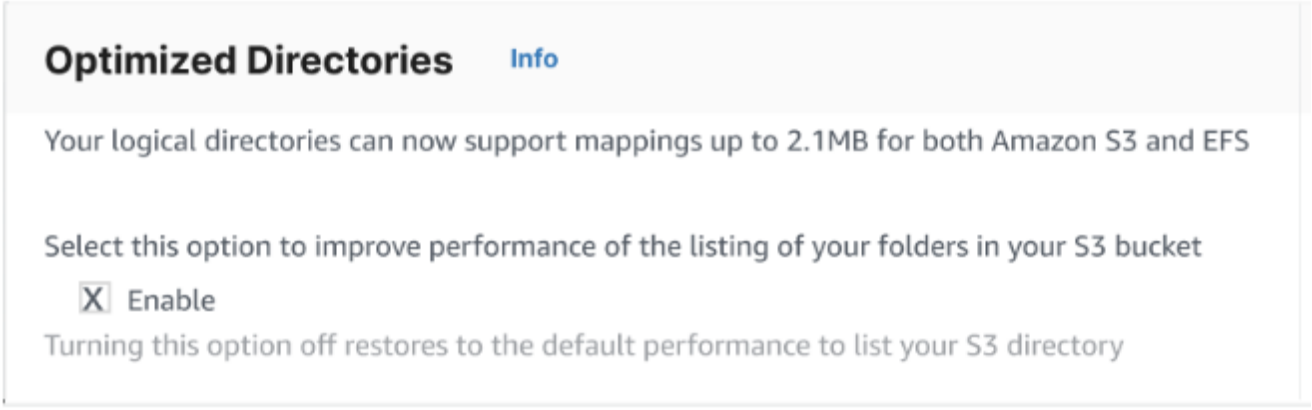

c. Wählen Sie unter Optionen für kryptografische Algorithmen eine Sicherheitsrichtlinie aus, die die kryptografischen Algorithmen enthält, die für die Verwendung durch Ihren Server aktiviert sind. Unsere neueste Sicherheitsrichtlinie ist die Standardeinstellung: Einzelheiten finden Sie unter. Sicherheitsrichtlinien für AWS Transfer Family Server

d. Lassen Sie das Feld Server-Hostschlüssel leer.

e. (Optional) Geben Sie für Tags für Schlüssel und Wert ein oder mehrere Tags als Schlüssel-Wert-Paare ein und wählen Sie dann Tag hinzufügen aus.

f. Sie können die Leistung Ihrer Amazon S3 S3-Verzeichnisse optimieren. Nehmen wir zum Beispiel an, Sie gehen in Ihr Home-Verzeichnis und haben 10.000 Unterverzeichnisse. Mit anderen Worten, Ihr Amazon S3 S3-Bucket hat 10.000 Ordner. Wenn Sie in diesem Szenario den Befehl `ls` (list) ausführen, dauert der Listenvorgang zwischen sechs und acht Minuten. Wenn Sie Ihre Verzeichnisse optimieren, dauert dieser Vorgang jedoch nur wenige Sekunden.

   Wenn Sie Ihren Server mit der Konsole erstellen, sind optimierte Verzeichnisse standardmäßig aktiviert. Wenn Sie Ihren Server mithilfe der API erstellen, ist dieses Verhalten standardmäßig nicht aktiviert.



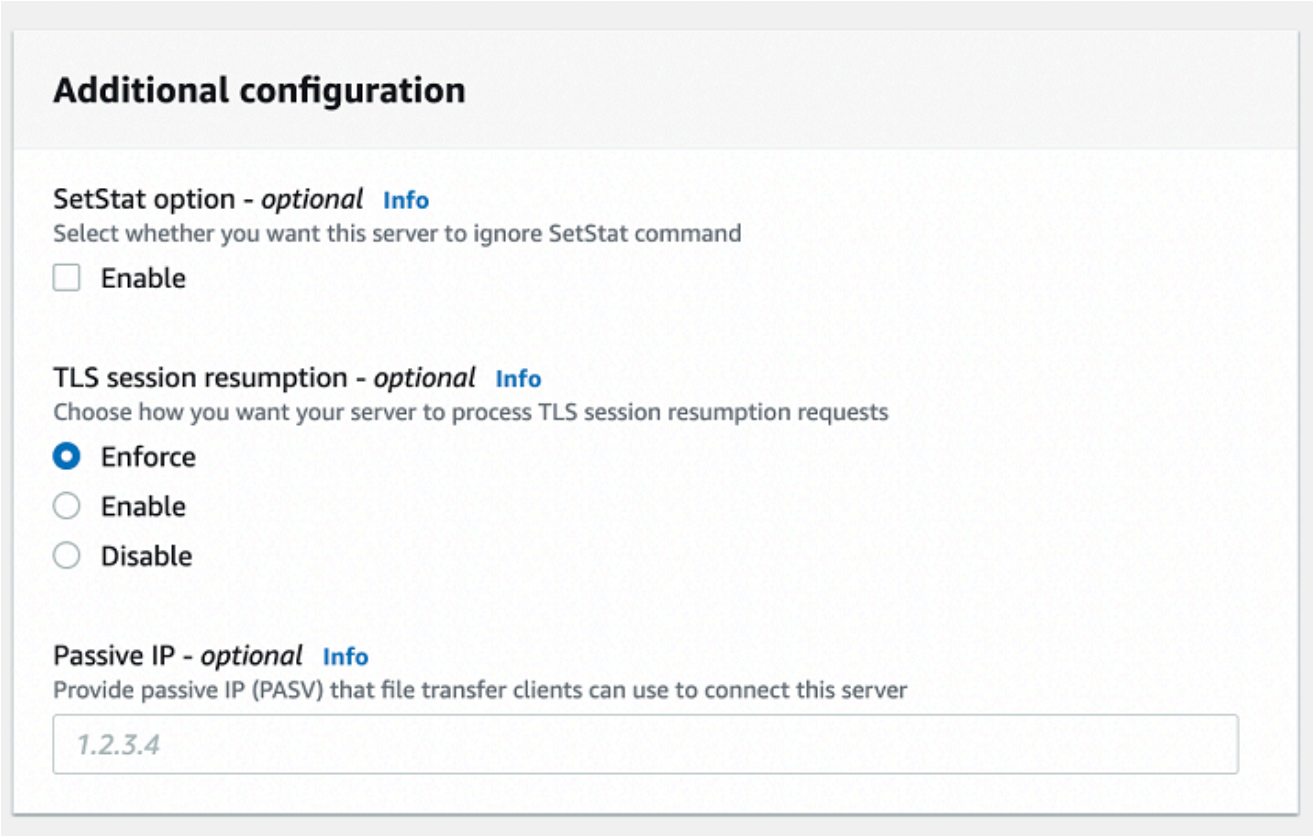

g. Wählen Sie Weiter aus.

h. (Optional) Sie können AWS Transfer Family Server so konfigurieren, dass Ihren Endbenutzern benutzerdefinierte Nachrichten wie Unternehmensrichtlinien oder Nutzungsbedingungen angezeigt werden. Sie können Benutzern, die sich erfolgreich authentifiziert haben, auch benutzerdefinierte Message of The Day (MOTD) anzeigen.

   Geben Sie für Banner anzeigen in das Textfeld Display-Banner vor der Authentifizierung die Textnachricht ein, die Sie Ihren Benutzern vor der Authentifizierung anzeigen möchten, und geben Sie in das Textfeld Display-Banner nach der Authentifizierung den Text ein, den Sie Ihren Benutzern nach erfolgreicher Authentifizierung anzeigen möchten.

i. (Optional) Sie können die folgenden zusätzlichen Optionen konfigurieren.

   - SetStat Option: Aktivieren Sie diese Option, um den Fehler zu ignorieren, der generiert wird, wenn ein Client versucht, eine Datei `SETSTAT` zu verwenden, die Sie in einen Amazon S3 S3-Bucket hochladen. Weitere Informationen finden Sie in der `SetStatOption` Dokumentation im ProtocolDetailsThema.
   - Wiederaufnahme der TLS-Sitzung: bietet einen Mechanismus zur Wiederaufnahme oder gemeinsamen Nutzung eines ausgehandelten geheimen Schlüssels zwischen der Kontroll- und Datenverbindung für eine FTPS-Sitzung. Weitere Informationen finden Sie in der `TlsSessionResumptionMode` Dokumentation zum Thema. ProtocolDetails
   - Passive IP: steht für den passiven Modus für FTP- und FTPS-Protokolle. Geben Sie eine einzelne IPv4-Adresse ein, z. B. die öffentliche IP-Adresse einer Firewall, eines Routers oder eines Load Balancers. Weitere Informationen finden Sie in der `PassiveIp` Dokumentation zum ProtocolDetailsThema.




8. Überprüfen Sie unter Überprüfen und erstellen Ihre Auswahl.

   - Wenn Sie eine davon bearbeiten möchten, wählen Sie neben dem Schritt Bearbeiten aus.

     > **Note**
     >
     > Sie müssen jeden Schritt nach dem Schritt überprüfen, den Sie bearbeiten möchten.

   - Wenn Sie keine Änderungen vorgenommen haben, wählen Sie Server erstellen, um Ihren Server zu erstellen. Sie gelangen zur Seite Servers (Server) (siehe unten), auf der der neue Server aufgelistet ist.

Es kann einige Minuten dauern, bis sich der Status Ihres neuen Servers auf Online ändert. Ab diesem Zeitpunkt kann der Server Dateioperationen für die Benutzer ausführen.

Nächste Schritte: Fahren Sie im nächsten Schritt mit Mit Anbietern benutzerdefinierter Identitäten arbeiten So richten Sie Benutzer ein.

# Erstellen Sie einen FTP-fähigen Server

Das File Transfer Protocol (FTP) ist ein Netzwerkprotokoll, das für die Übertragung von Daten verwendet wird. FTP verwendet einen separaten Kanal für die Steuerung und Datenübertragung. Der Steuerkanal ist geöffnet, bis er beendet wird oder ein Timeout bei Inaktivität eintritt. Der Datenkanal ist für die Dauer der Übertragung aktiv. FTP verwendet Klartext und unterstützt keine Verschlüsselung des Datenverkehrs.

> **Note**
>
> Wenn Sie FTP aktivieren, müssen Sie die interne Zugriffsoption für den VPC-gehosteten Endpunkt auswählen. Wenn Ihr Server Daten über das öffentliche Netzwerk übertragen lassen soll, müssen Sie sichere Protokolle wie SFTP oder FTPS verwenden.

Um einen FTP-fähigen Server zu erstellen

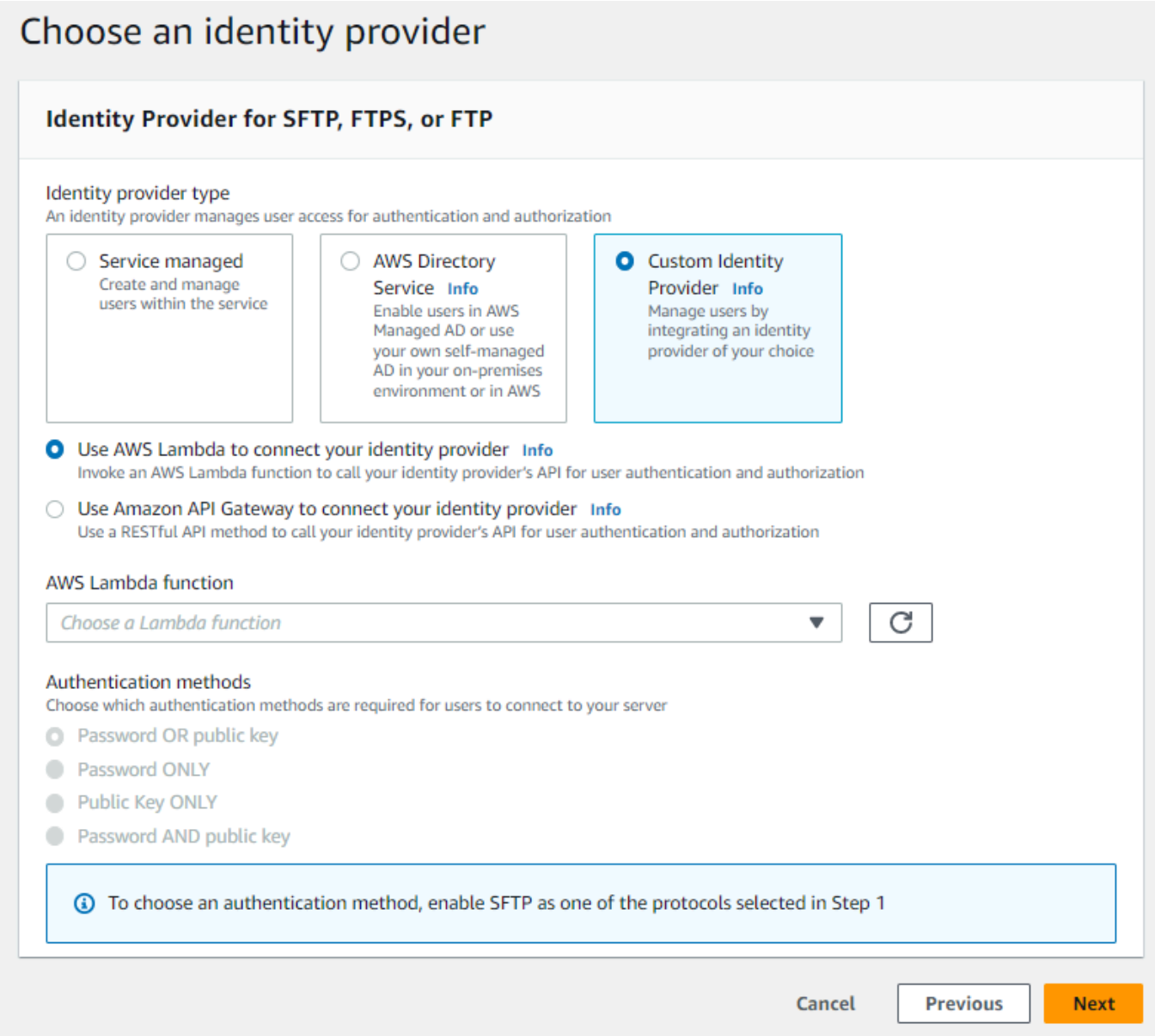
1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/ und wählen Sie im Navigationsbereich Server und anschließend Server erstellen aus.
2. Wählen Sie unter Protokolle auswählen die Option FTP und dann Weiter aus.
3. Wählen Sie unter Wählen Sie einen Identitätsanbieter aus den Identitätsanbieter aus, den Sie für die Verwaltung des Benutzerzugriffs verwenden möchten. Ihnen stehen folgende Optionen zur Verfügung:

   - AWS Directory Service for Microsoft Active Directory— Sie stellen ein AWS Directory Service Verzeichnis für den Zugriff auf den Endpunkt bereit. Auf diese Weise können Sie die in Ihrem Active Directory gespeicherten Anmeldeinformationen verwenden, um Ihre Benutzer zu authentifizieren. Weitere Informationen zur Arbeit mit AWS Managed Microsoft AD Identitätsanbietern finden Sie unterVerwenden des AWS Directory Service-Identitätsanbieters.

     > **Note**
     >
     > - Kontoübergreifende Verzeichnisse und gemeinsam genutzte Verzeichnisse werden für AWS Managed Microsoft AD nicht unterstützt.
     > - Um einen Server mit Directory Service als Identitätsanbieter einzurichten, müssen Sie einige AWS Directory Service Berechtigungen hinzufügen. Details hierzu finden

Sie unter Bevor Sie mit der Verwendung von beginnen AWS Directory Service for Microsoft Active Directory.

- Benutzerdefinierter Identitätsanbieter — Wählen Sie eine der folgenden Optionen:
  - Verwenden Sie AWS Lambda , um Ihren Identitätsanbieter zu verbinden — Sie können einen vorhandenen Identitätsanbieter verwenden, der von einer Lambda-Funktion unterstützt wird. Sie geben den Namen der Lambda-Funktion an. Weitere Informationen finden Sie unter Wird AWS Lambda zur Integration Ihres Identitätsanbieters verwendet.
  - Verwenden Sie Amazon API Gateway, um Ihren Identitätsanbieter zu verbinden — Sie können eine API Gateway Gateway-Methode erstellen, die von einer Lambda-Funktion unterstützt wird, um sie als Identitätsanbieter zu verwenden. Sie geben eine Amazon API Gateway Gateway-URL und eine Aufruf-Rolle an. Weitere Informationen finden Sie unter Verwenden Sie Amazon API Gateway zur Integration Ihres Identitätsanbieters.




4. Wählen Sie Weiter aus.
5. Gehen Sie unter Endpunkt auswählen wie folgt vor:

> **Note**
>
> FTP-Server für Transfer Family arbeiten über Port 21 (Steuerkanal) und Portbereich 8192—8200 (Datenkanal).

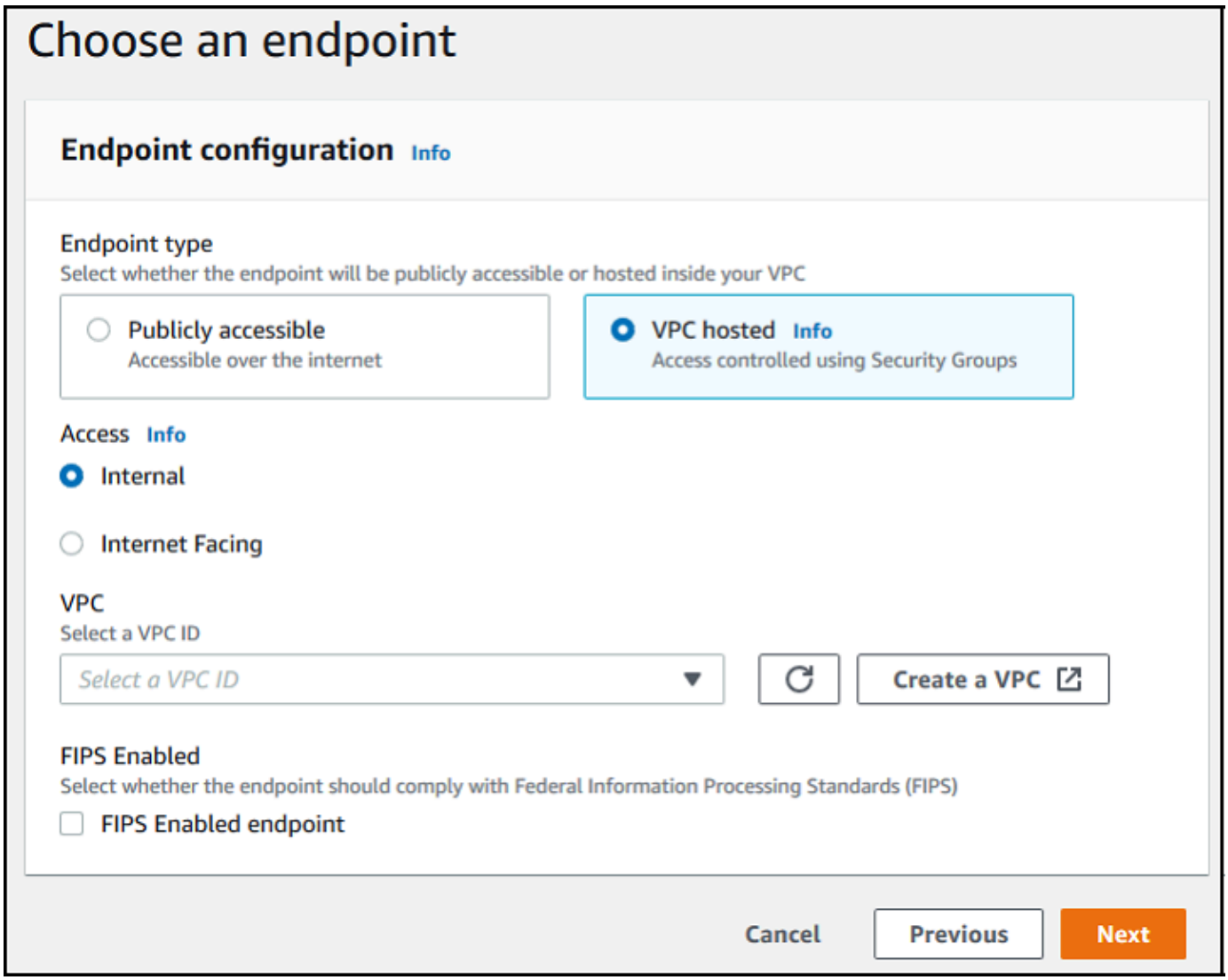
a. Wählen Sie als Endpunkttyp die Option VPC hostet, um den Endpunkt Ihres Servers zu hosten. Informationen zur Einrichtung Ihres VPC-gehosteten Endpunkts finden Sie unterErstellen Sie einen Server in einer virtuellen privaten Cloud.

> **Note**
>
> Öffentlich zugängliche Endpunkte werden nicht unterstützt.

b. Lassen Sie für FIPS Enabled das Kontrollkästchen FIPS-fähiger Endpunkt deaktiviert.

> **Note**
>
> FIPS-fähige Endpunkte werden für FTP-Server nicht unterstützt.

c. Wählen Sie Weiter aus.



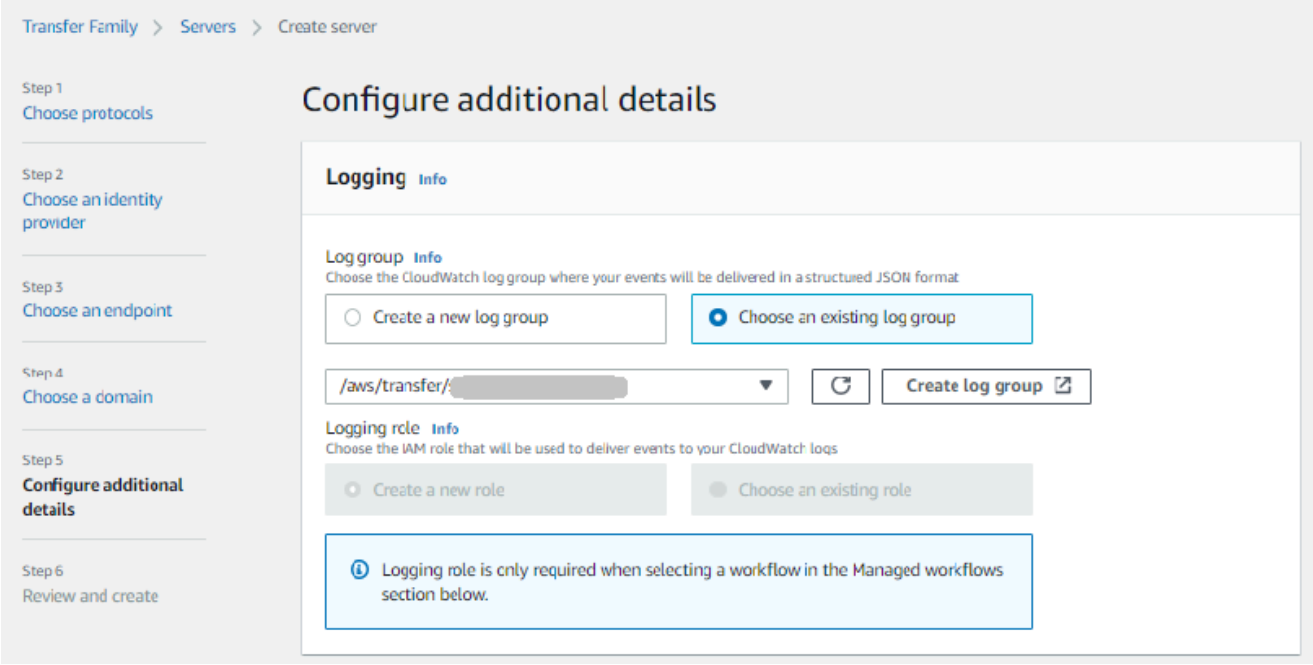

6. Wählen Sie auf der Seite „Domain auswählen“ den AWS Speicherdienst aus, den Sie zum Speichern und Zugreifen auf Ihre Daten über das ausgewählte Protokoll verwenden möchten.

   - Wählen Sie Amazon S3, um Ihre Dateien als Objekte über das ausgewählte Protokoll zu speichern und darauf zuzugreifen.
   - Wählen Sie Amazon EFS, um Ihre Dateien in Ihrem Amazon EFS-Dateisystem über das ausgewählte Protokoll zu speichern und darauf zuzugreifen.

   Wählen Sie Weiter aus.

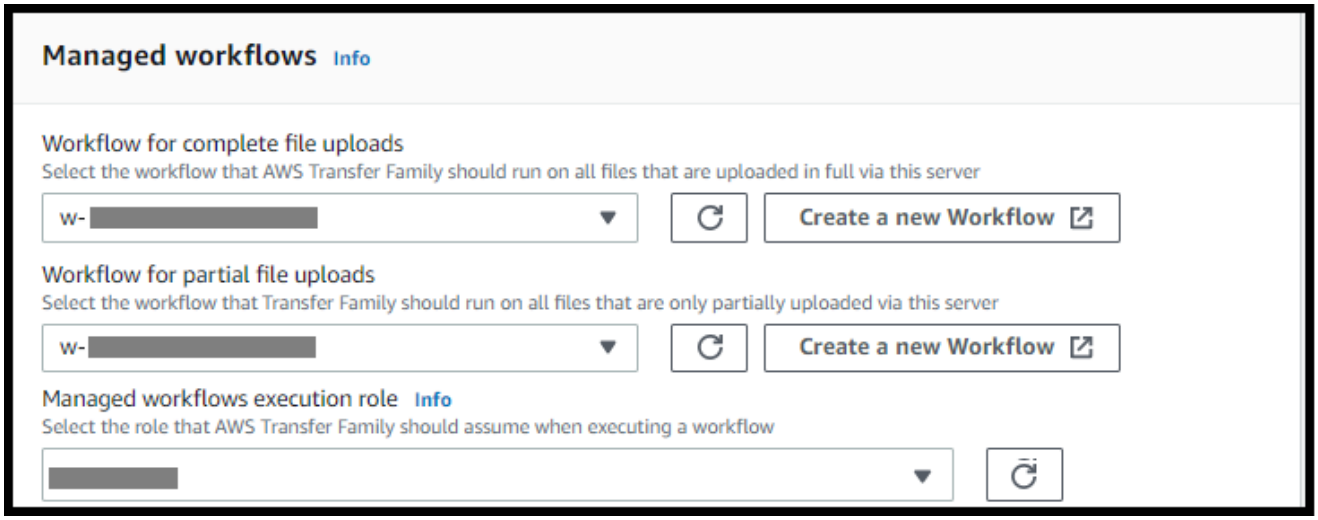
7. Gehen Sie unter Zusätzliche Details konfigurieren wie folgt vor:

a. Geben Sie für die Protokollierung eine bestehende Protokollgruppe an oder erstellen Sie eine neue (Standardoption).




Wenn Sie „Protokollgruppe erstellen“ wählen, wird in der CloudWatch Konsole (https://console.aws.amazon.com/cloudwatch/) die Seite „Protokollgruppe erstellen“ geöffnet. Einzelheiten finden Sie unter Protokollgruppe erstellen in CloudWatch Logs.

b. (Optional) Wählen Sie für verwaltete Workflows Workflow-IDs (und eine entsprechende Rolle) aus, die Transfer Family bei der Ausführung des Workflows annehmen soll. Sie können einen Workflow auswählen, der bei einem vollständigen Upload ausgeführt wird, und einen anderen, der bei einem teilweisen Upload ausgeführt werden soll. Weitere Informationen zur Verarbeitung Ihrer Dateien mithilfe verwalteter Workflows finden Sie unterAWS Transfer Family verwaltete Workflows.




c. Wählen Sie unter Optionen für kryptografische Algorithmen eine Sicherheitsrichtlinie aus, die die kryptografischen Algorithmen enthält, die für die Verwendung durch Ihren Server aktiviert sind.

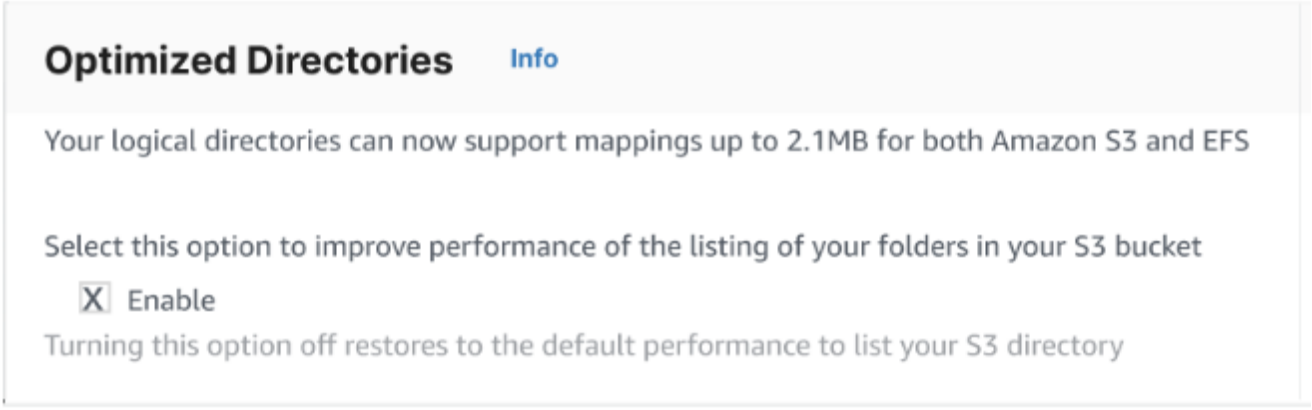
> **Note**
>
> Transfer Family weist Ihrem FTP-Server die neueste Sicherheitsrichtlinie zu. Da das FTP-Protokoll jedoch keine Verschlüsselung verwendet, verwenden FTP-Server keinen der Sicherheitsrichtlinien-Algorithmen. Sofern Ihr Server nicht auch das FTPS- oder SFTP-Protokoll verwendet, bleibt die Sicherheitsrichtlinie ungenutzt.

d. Lassen Sie das Feld Server-Hostschlüssel leer.

e. (Optional) Geben Sie für Tags für Schlüssel und Wert ein oder mehrere Tags als Schlüssel-Wert-Paare ein und wählen Sie dann Tag hinzufügen aus.

f. Sie können die Leistung Ihrer Amazon S3 S3-Verzeichnisse optimieren. Nehmen wir zum Beispiel an, Sie gehen in Ihr Home-Verzeichnis und haben 10.000 Unterverzeichnisse. Mit anderen Worten, Ihr Amazon S3 S3-Bucket hat 10.000 Ordner. Wenn Sie in diesem Szenario den Befehl `ls` (list) ausführen, dauert der Listenvorgang zwischen sechs und acht Minuten. Wenn Sie Ihre Verzeichnisse optimieren, dauert dieser Vorgang jedoch nur wenige Sekunden.

   Wenn Sie Ihren Server mit der Konsole erstellen, sind optimierte Verzeichnisse standardmäßig aktiviert. Wenn Sie Ihren Server mithilfe der API erstellen, ist dieses Verhalten standardmäßig nicht aktiviert.



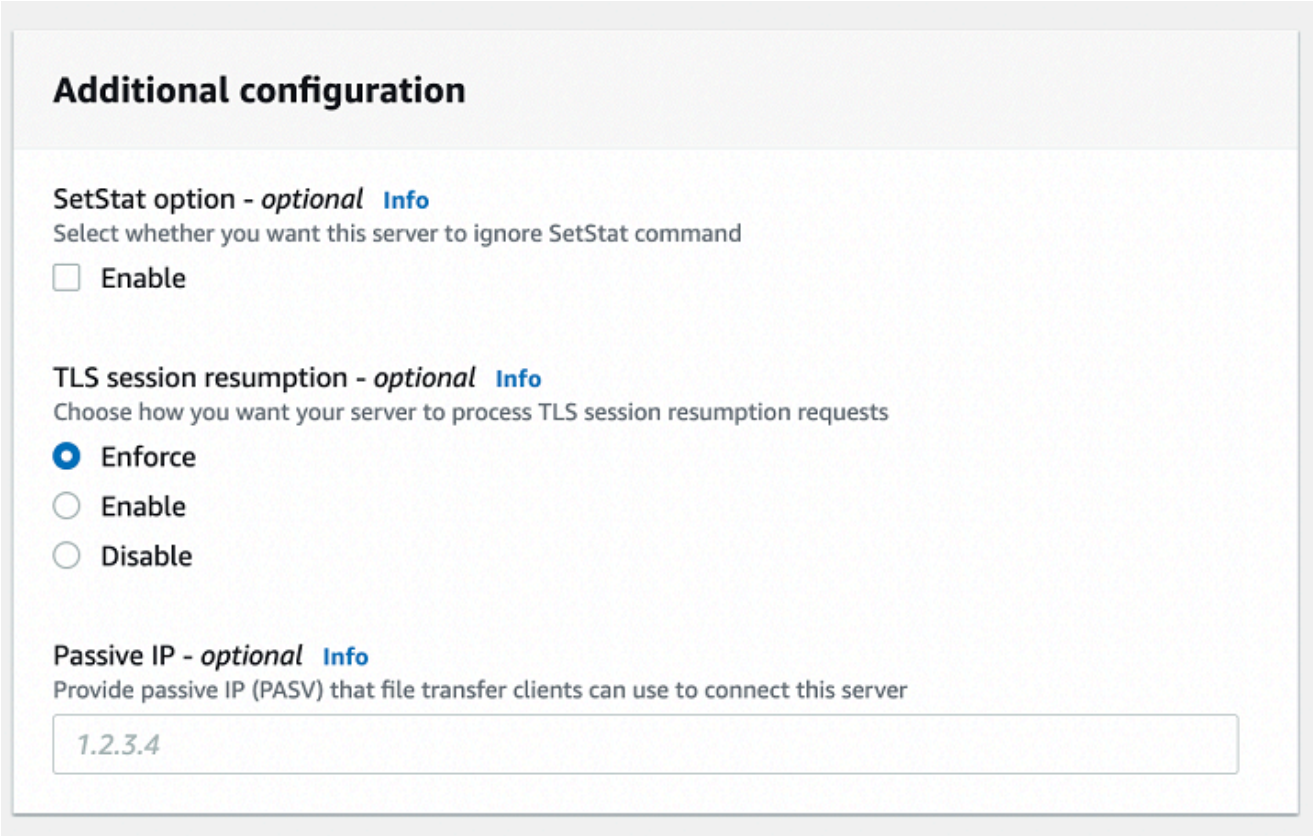

g. Wählen Sie Weiter aus.

h. (Optional) Sie können AWS Transfer Family Server so konfigurieren, dass Ihren Endbenutzern benutzerdefinierte Nachrichten wie Unternehmensrichtlinien oder Nutzungsbedingungen angezeigt werden. Sie können Benutzern, die sich erfolgreich authentifiziert haben, auch benutzerdefinierte Message of The Day (MOTD) anzeigen.

   Geben Sie für Banner anzeigen in das Textfeld Display-Banner vor der Authentifizierung die Textnachricht ein, die Sie Ihren Benutzern vor der Authentifizierung anzeigen möchten, und geben Sie in das Textfeld Display-Banner nach der Authentifizierung den Text ein, den Sie Ihren Benutzern nach erfolgreicher Authentifizierung anzeigen möchten.

i. (Optional) Sie können die folgenden zusätzlichen Optionen konfigurieren.

   - SetStat Option: Aktivieren Sie diese Option, um den Fehler zu ignorieren, der generiert wird, wenn ein Client versucht, eine Datei `SETSTAT` zu verwenden, die Sie in einen Amazon S3 S3-Bucket hochladen. Weitere Informationen finden Sie in der `SetStatOption` Dokumentation im ProtocolDetailsThema.
   - Wiederaufnahme der TLS-Sitzung: bietet einen Mechanismus zur Wiederaufnahme oder gemeinsamen Nutzung eines ausgehandelten geheimen Schlüssels zwischen der Kontroll- und Datenverbindung für eine FTPS-Sitzung. Weitere Informationen finden Sie in der `TlsSessionResumptionMode` Dokumentation zum Thema. ProtocolDetails
   - Passive IP: steht für den passiven Modus für FTP- und FTPS-Protokolle. Geben Sie eine einzelne IPv4-Adresse ein, z. B. die öffentliche IP-Adresse einer Firewall, eines Routers oder eines Load Balancers. Weitere Informationen finden Sie in der `PassiveIp` Dokumentation zum ProtocolDetailsThema.




8. Überprüfen Sie unter Überprüfen und erstellen Ihre Auswahl.

   - Wenn Sie eine davon bearbeiten möchten, wählen Sie neben dem Schritt Bearbeiten aus.

     > **Note**
     >
     > Sie müssen jeden Schritt nach dem Schritt überprüfen, den Sie bearbeiten möchten.

   - Wenn Sie keine Änderungen vorgenommen haben, wählen Sie Server erstellen, um Ihren Server zu erstellen. Sie gelangen zur Seite Servers (Server) (siehe unten), auf der der neue Server aufgelistet ist.

Es kann einige Minuten dauern, bis sich der Status Ihres neuen Servers auf Online ändert. Ab diesem Zeitpunkt kann der Server Dateioperationen für die Benutzer ausführen.

Nächste Schritte — Fahren Sie im nächsten Schritt mit Mit Anbietern benutzerdefinierter Identitäten arbeiten So richten Sie Benutzer ein.

# Erstellen Sie einen Server in einer virtuellen privaten Cloud

Sie können den Endpunkt Ihres Servers in einer Virtual Private Cloud (VPC) hosten, um Daten zu und von einem Amazon S3-Bucket oder Amazon EFS-Dateisystem zu übertragen, ohne das öffentliche Internet nutzen zu müssen.

> **Note**
>
> Nach dem 19. Mai 2021 können Sie mit `EndpointType=VPC_ENDPOINT` Ihrem Konto keinen Server mehr erstellen, wenn Ihr AWS Konto dies nicht bereits vor dem 19. Mai 2021 getan hat. Wenn du am oder vor dem 21. Februar 2021 bereits Server mit `EndpointType=VPC_ENDPOINT` deinem AWS Konto erstellt hast, bist du davon nicht betroffen. Verwenden Sie nach diesem Datum `EndpointType` **=`VPC`**. Weitere Informationen finden Sie unter the section called “Einstellung der Verwendung von VPC_ENDPOINT”.

Wenn Sie Amazon Virtual Private Cloud (Amazon VPC) zum Hosten Ihrer AWS Ressourcen verwenden, können Sie eine private Verbindung zwischen Ihrer VPC und einem Server herstellen. Sie können diesen Server dann verwenden, um Daten über Ihren Client zu und von Ihrem Amazon S3 S3-Bucket zu übertragen, ohne öffentliche IP-Adressen zu verwenden oder ein Internet-Gateway zu benötigen.

Mit Amazon VPC können Sie AWS Ressourcen in einem benutzerdefinierten virtuellen Netzwerk starten. Mit einer VPC können Sie Netzwerkeinstellungen, wie IP-Adressbereich, Subnetze, Routing-Tabellen und Netzwerk-Gateways, steuern. Weitere Informationen zu VPCs finden Sie unter Was ist Amazon VPC? im Amazon VPC-Benutzerhandbuch.

In den nächsten Abschnitten finden Sie Anweisungen zum Erstellen und Verbinden Ihrer VPC mit einem Server. Im Überblick gehen Sie dazu wie folgt vor:

1. Richten Sie einen Server mit einem VPC-Endpunkt ein.
2. Connect Sie mithilfe eines Clients, der sich in Ihrer VPC befindet, über den VPC-Endpunkt eine Verbindung zu Ihrem Server her. Auf diese Weise können Sie Daten, die in Ihrem Amazon S3 S3-Bucket gespeichert sind, mithilfe von über Ihren Client übertragen AWS Transfer Family. Sie können diese Übertragung durchführen, obwohl das Netzwerk vom öffentlichen Internet getrennt ist.
3. Wenn Sie sich dafür entscheiden, den Endpunkt Ihres Servers mit dem Internet zu verbinden, können Sie Ihrem Endpunkt außerdem Elastic IP-Adressen zuordnen. Auf diese Weise können

Clients außerhalb Ihrer VPC eine Verbindung zu Ihrem Server herstellen. Sie können VPC-Sicherheitsgruppen verwenden, um den Zugriff auf authentifizierte Benutzer zu kontrollieren, deren Anfragen nur von zulässigen Adressen stammen.

Themen


- Erstellen Sie einen Serverendpunkt, auf den nur innerhalb Ihrer VPC zugegriffen werden kann
- Erstellen Sie einen mit dem Internet verbundenen Endpunkt für Ihren Server
- Ändern Sie den Endpunkttyp für Ihren Server
- Einstellung der Verwendung von VPC_ENDPOINT
- Aktualisierung des AWS Transfer Family Serverendpunkttyps von VPC_ENDPOINT auf VPC


## Erstellen Sie einen Serverendpunkt, auf den nur innerhalb Ihrer VPC zugegriffen werden kann

Im folgenden Verfahren erstellen Sie einen Serverendpunkt, auf den nur Ressourcen innerhalb Ihrer VPC zugreifen können.

So erstellen Sie einen Serverendpunkt in einer VPC

1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Wählen Sie im Navigationsbereich Server und anschließend Server erstellen aus.
3. Wählen Sie unter Protokolle auswählen ein oder mehrere Protokolle aus, und klicken Sie dann auf Weiter. Weitere Informationen zu Protokollen finden Sie unterSchritt 2: Erstellen Sie einen SFTP-fähigen Server.
4. Wählen Sie unter Einen Identitätsanbieter auswählen die Option Dienst verwaltet, um Benutzeridentitäten und Schlüssel zu speichern AWS Transfer Family, und klicken Sie dann auf Weiter.

> **Note**
>
> Bei diesem Verfahren wird die vom Dienst verwaltete Option verwendet. Wenn Sie Benutzerdefiniert wählen, geben Sie einen Amazon API Gateway Gateway-Endpunkt und eine AWS Identity and Access Management (IAM) -Rolle für den Zugriff auf den Endpunkt an. Auf diese Weise können Sie Ihren Verzeichnisdienst integrieren, um Ihre Benutzer zu authentifizieren und zu autorisieren. Weitere Informationen zur Verwendung

benutzerdefinierter Identitätsanbieter siehe Mit Anbietern benutzerdefinierter Identitäten arbeiten.

5. Gehen Sie unter Endpunkt auswählen wie folgt vor:

> **Note**
>
> FTP- und FTPS-Server für Transfer Family arbeiten über Port 21 (Steuerkanal) und Portbereich 8192-8200 (Datenkanal).

a. Wählen Sie als Endpunkttyp den VPC-gehosteten Endpunkttyp aus, um den Endpunkt Ihres Servers zu hosten.

b. Wählen Sie für Zugriff die Option Intern aus, damit Ihr Endpunkt nur für Clients zugänglich ist, die die privaten IP-Adressen des Endpunkts verwenden.

> **Note**
>
> Einzelheiten zur Option Internetzugriff finden Sie unterErstellen Sie einen mit dem Internet verbundenen Endpunkt für Ihren Server. Ein Server, der in einer VPC nur für den internen Zugriff erstellt wurde, unterstützt keine benutzerdefinierten Hostnamen.

c. Wählen Sie für VPC eine vorhandene VPC-ID oder wählen Sie Create a VPC, um eine neue VPC zu erstellen.

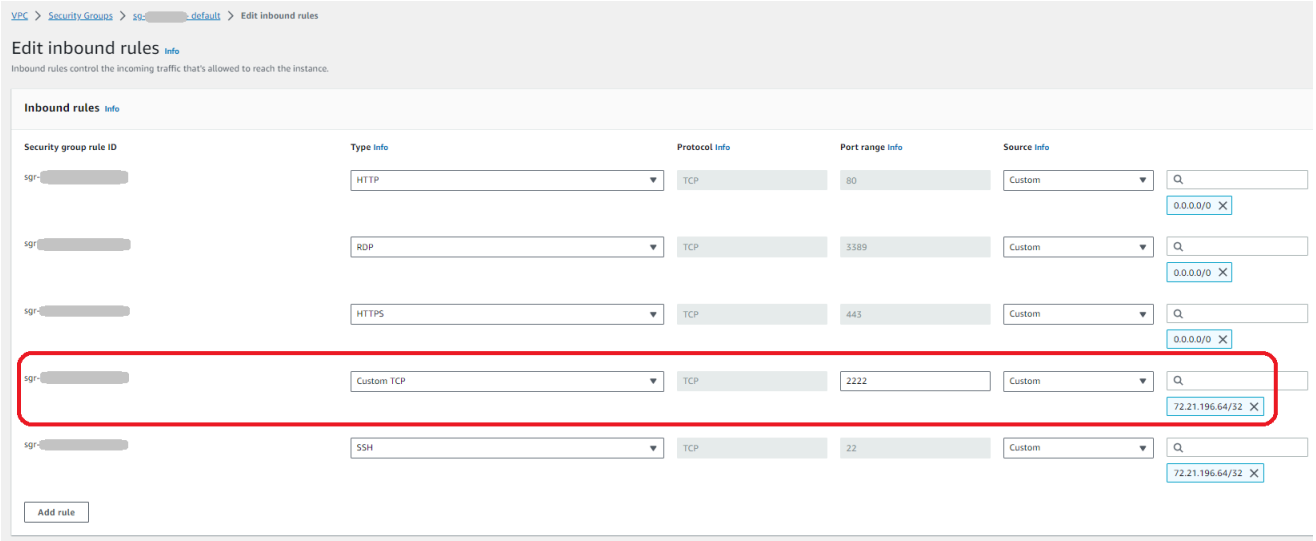
d. Wählen Sie im Abschnitt Availability Zones bis zu drei Availability Zones und zugehörige Subnetze aus.

e. Wählen Sie im Abschnitt Sicherheitsgruppen eine oder mehrere vorhandene Sicherheitsgruppen-IDs aus, oder wählen Sie Sicherheitsgruppe erstellen, um eine neue Sicherheitsgruppe zu erstellen. Weitere Informationen zu Sicherheitsgruppen finden Sie unter Sicherheitsgruppen für Ihre VPC im Amazon Virtual Private Cloud Cloud-Benutzerhandbuch. Informationen zum Erstellen einer Sicherheitsgruppe finden Sie unter Erstellen einer Sicherheitsgruppe im Amazon Virtual Private Cloud Cloud-Benutzerhandbuch.

> **Note**
>
> Ihre VPC verfügt automatisch über eine Standardsicherheitsgruppe. Wenn Sie beim Starten des Servers keine andere Sicherheitsgruppe oder Gruppen angeben, ordnen wir Ihrem Server die Standardsicherheitsgruppe zu.

Für die eingehenden Regeln für die Sicherheitsgruppe können Sie den SSH-Verkehr so konfigurieren, dass er Port 22, 2222, 22000 oder eine beliebige Kombination verwendet. Port 22 ist standardmäßig konfiguriert. Um Port 2222 oder Port 22000 zu verwenden, fügen Sie Ihrer Sicherheitsgruppe eine Regel für eingehenden Datenverkehr hinzu. Wählen Sie für den Typ Benutzerdefiniertes TCP aus, geben Sie dann entweder **`2222`** oder **`22000`** als Portbereich ein und geben Sie für die Quelle denselben CIDR-Bereich ein, den Sie für Ihre SSH-Port-22-Regel haben.

> **Note**
>
> Sie können Port 2223 auch für Clients verwenden, die TCP-Pickback-ACKs oder die Fähigkeit benötigen, dass das letzte Ack des TCP-3-Wege-Handshakes auch Daten enthält.
> Manche Client-Software ist möglicherweise nicht mit Port 2223 kompatibel, z. B. ein Client, bei dem der Server die SFTP-Identifikationszeichenfolge senden muss, bevor der Client dies tut.




f. (Optional) Aktivieren Sie für FIPS Enabled das Kontrollkästchen FIPS-fähiger Endpunkt, um sicherzustellen, dass der Endpunkt den Federal Information Processing Standards (FIPS) entspricht.

> **Note**
>
> FIPS-fähige Endpunkte sind nur in nordamerikanischen Regionen verfügbar. AWS Informationen zu den verfügbaren Regionen finden Sie unter AWS Transfer Family Endpunkte und Kontingente in der. Allgemeine AWS-Referenz Weitere Informationen zu FIPS finden Sie unter Federal Information Processing Standard (FIPS) 140-2.

g. Wählen Sie Weiter aus.

6. Gehen Sie unter Zusätzliche Details konfigurieren wie folgt vor:

   a. Wählen Sie für die CloudWatch Protokollierung eine der folgenden Optionen, um die CloudWatch Amazon-Protokollierung Ihrer Benutzeraktivitäten zu aktivieren:

   - Erstellen Sie eine neue Rolle, damit Transfer Family die IAM-Rolle automatisch erstellen kann, sofern Sie über die erforderlichen Berechtigungen zum Erstellen einer neuen Rolle verfügen. Die erstellte IAM-Rolle wird aufgerufen. `AWSTransferLoggingAccess`
   - Wählen Sie eine vorhandene Rolle aus, um eine bestehende IAM-Rolle aus Ihrem Konto auszuwählen. Wählen Sie unter Logging-Rolle die Rolle aus. Diese IAM-Rolle sollte

eine Vertrauensrichtlinie mit der Einstellung Service auf `transfer.amazonaws.com` enthalten.

Weitere Informationen zur CloudWatch Protokollierung finden Sie unterKonfigurieren Sie die CloudWatch Protokollierungsrolle.

> **Note**
>
> - Sie können keine Endbenutzeraktivitäten in anzeigen, CloudWatch wenn Sie keine Protokollierungsrolle angeben.
> - Wenn Sie keine CloudWatch Protokollierungsrolle einrichten möchten, wählen Sie Bestehende Rolle auswählen, aber wählen Sie keine Protokollierungsrolle aus.

b. Wählen Sie unter Optionen für kryptografische Algorithmen eine Sicherheitsrichtlinie aus, die die kryptografischen Algorithmen enthält, die für die Verwendung durch Ihren Server aktiviert sind.

> **Note**
>
> Standardmäßig ist die `TransferSecurityPolicy-2020-06` Sicherheitsrichtlinie an Ihren Server angehängt, sofern Sie keine andere auswählen.

Weitere Informationen zu Sicherheitsrichtlinien finden Sie unter Sicherheitsrichtlinien für AWS Transfer Family Server.

c. (Optional: Dieser Abschnitt ist nur für die Migration von Benutzern von einem vorhandenen SFTP-fähigen Server vorgesehen.) Geben Sie als Server-Hostschlüssel einen privaten RSA-, ED25519- oder ECDSA-Schlüssel ein, der zur Identifizierung Ihres Servers verwendet wird, wenn Clients über SFTP eine Verbindung zu ihm herstellen.

d. (Optional) Geben Sie für Tags für Schlüssel und Wert ein oder mehrere Tags als Schlüssel-Wert-Paare ein und wählen Sie dann Tag hinzufügen aus.

e. Wählen Sie Weiter aus.

7. Überprüfen Sie unter Überprüfen und erstellen Ihre Auswahl. Wenn Sie:

- Wenn Sie eine davon bearbeiten möchten, wählen Sie neben dem Schritt Bearbeiten aus.

> **Note**
>
> Sie müssen jeden Schritt nach dem Schritt, den Sie bearbeiten möchten, erneut überprüfen.

- Haben Sie keine Änderungen vorgenommen, wählen Sie Server erstellen, um Ihren Server zu erstellen. Sie gelangen zur Seite Servers (Server) (siehe unten), auf der der neue Server aufgelistet ist.

Es kann einige Minuten dauern, bis sich der Status Ihres neuen Servers auf Online ändert. Zu diesem Zeitpunkt kann Ihr Server Dateioperationen ausführen, aber Sie müssen zuerst einen Benutzer erstellen. Einzelheiten zum Erstellen von Benutzern finden Sie unterVerwalten von Benutzern für Serverendpunkte.

## Erstellen Sie einen mit dem Internet verbundenen Endpunkt für Ihren Server

Im folgenden Verfahren erstellen Sie einen Serverendpunkt. Auf diesen Endpunkt können nur Clients über das Internet zugreifen, deren Quell-IP-Adressen in der Standardsicherheitsgruppe Ihrer VPC zulässig sind. Indem Sie Elastic IP-Adressen verwenden, um Ihren Endpunkt mit dem Internet zu verbinden, können Ihre Kunden die Elastic IP-Adresse außerdem verwenden, um den Zugriff auf Ihren Endpunkt in ihren Firewalls zu ermöglichen.

> **Note**
>
> Nur SFTP und FTPS können auf einem mit dem Internet verbundenen VPC-gehosteten Endpunkt verwendet werden.

Um einen mit dem Internet verbundenen Endpunkt zu erstellen

1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Wählen Sie im Navigationsbereich Server und anschließend Server erstellen aus.
3. Wählen Sie unter Protokolle auswählen ein oder mehrere Protokolle aus, und klicken Sie dann auf Weiter. Weitere Informationen zu Protokollen finden Sie unterSchritt 2: Erstellen Sie einen SFTP-fähigen Server.

4. Wählen Sie unter Einen Identitätsanbieter auswählen die Option Dienst verwaltet, um Benutzeridentitäten und Schlüssel zu speichern AWS Transfer Family, und klicken Sie dann auf Weiter.

> **Note**
>
> Bei diesem Verfahren wird die vom Dienst verwaltete Option verwendet. Wenn Sie Benutzerdefiniert wählen, geben Sie einen Amazon API Gateway Gateway-Endpunkt und eine AWS Identity and Access Management (IAM) -Rolle für den Zugriff auf den Endpunkt an. Auf diese Weise können Sie Ihren Verzeichnisdienst integrieren, um Ihre Benutzer zu authentifizieren und zu autorisieren. Weitere Informationen zur Verwendung benutzerdefinierter Identitätsanbieter siehe Mit Anbietern benutzerdefinierter Identitäten arbeiten.

5. Gehen Sie unter Endpunkt auswählen wie folgt vor:

   a. Wählen Sie als Endpunkttyp den VPC-gehosteten Endpunkttyp aus, um den Endpunkt Ihres Servers zu hosten.

   b. Wählen Sie für Access die Option Internet Facing, um Ihren Endpunkt für Kunden über das Internet zugänglich zu machen.

   > **Note**
   >
   > Wenn Sie sich für Internet Facing entscheiden, können Sie in jedem Subnetz oder Subnetzen eine bestehende Elastic IP-Adresse auswählen. Oder Sie können die VPC-Konsole (https://console.aws.amazon.com/vpc/) aufrufen, um eine oder mehrere neue Elastic IP-Adressen zuzuweisen. Diese Adressen können entweder Ihnen AWS oder Ihnen gehören. Sie können Elastic IP-Adressen, die bereits verwendet werden, nicht mit Ihrem Endpunkt verknüpfen.

   c. (Optional) Wählen Sie für Benutzerdefinierter Hostname eine der folgenden Optionen:

   > **Note**
   >
   > Kunden, die eine direkte Verbindung über die Elastic IP-Adresse herstellen oder einen Hostnamen-Datensatz in Commercial Route 53 erstellen AWS GovCloud (US) müssen, der auf ihre EIP verweist. Weitere Informationen zur Verwendung von Route 53 für GovCloud Endgeräte finden Sie unter Einrichten von Amazon

Route 53 mit Ihren AWS GovCloud (US) Ressourcen im AWS GovCloud (US) Benutzerhandbuch.

- Amazon Route 53 DNS-Alias — wenn der Hostname, den Sie verwenden möchten, bei Route 53 registriert ist. Sie können dann den Hostnamen eingeben.
- Anderes DNS — wenn der Hostname, den Sie verwenden möchten, bei einem anderen DNS-Anbieter registriert ist. Sie können dann den Hostnamen eingeben.
- Keine — um den Endpunkt des Servers zu verwenden und keinen benutzerdefinierten Hostnamen zu verwenden. Der Server-Host-Name hat das Format *server-id*`.server.transfer.`*region*`.amazonaws.com`.

> **Note**
>
> Für Kunden in AWS GovCloud (US) wird durch die Auswahl von None kein Hostname in diesem Format erstellt.

Weitere Informationen zum Arbeiten mit benutzerdefinierten Hostnamen finden Sie unter Mit benutzerdefinierten Hostnamen arbeiten

d. Wählen Sie für VPC eine vorhandene VPC-ID oder wählen Sie Create a VPC, um eine neue VPC zu erstellen.

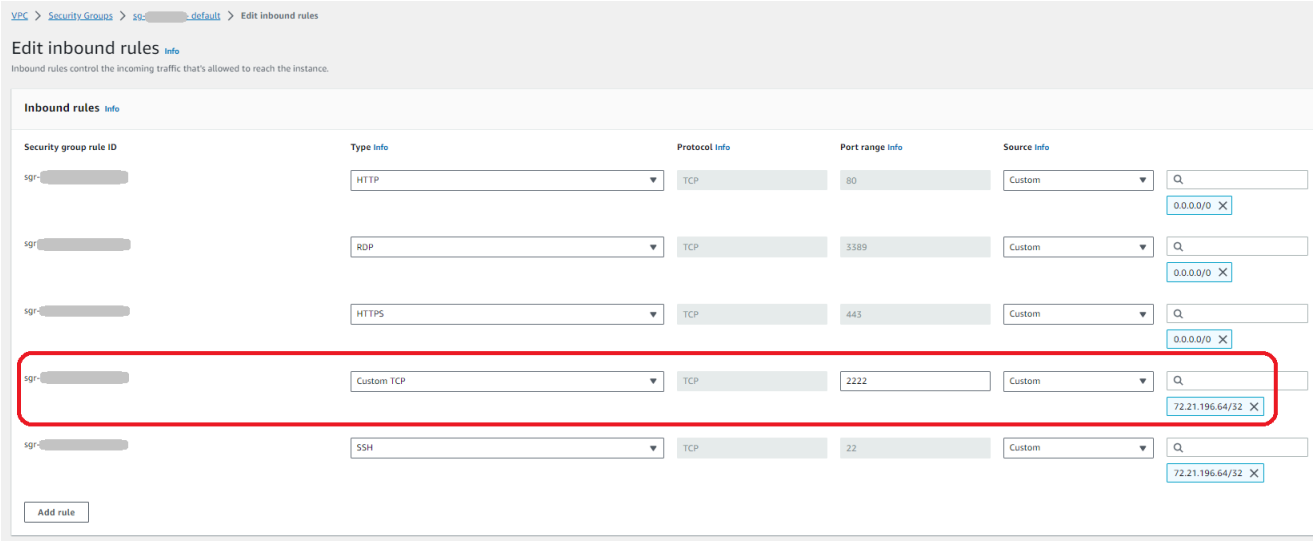
e. Wählen Sie im Abschnitt Availability Zones bis zu drei Availability Zones und zugehörige Subnetze aus. Wählen Sie für IPv4-Adressen eine Elastic IP-Adresse für jedes Subnetz aus. Dies ist die IP-Adresse, die Ihre Clients verwenden können, um den Zugriff auf Ihren Endpunkt in ihren Firewalls zu ermöglichen.

f. Wählen Sie im Abschnitt Sicherheitsgruppen eine oder mehrere vorhandene Sicherheitsgruppen-IDs aus, oder wählen Sie Sicherheitsgruppe erstellen, um eine neue Sicherheitsgruppe zu erstellen. Weitere Informationen zu Sicherheitsgruppen finden Sie unter Sicherheitsgruppen für Ihre VPC im Amazon Virtual Private Cloud Cloud-Benutzerhandbuch. Informationen zum Erstellen einer Sicherheitsgruppe finden Sie unter Erstellen einer Sicherheitsgruppe im Amazon Virtual Private Cloud Cloud-Benutzerhandbuch.

> **Note**
>
> Ihre VPC verfügt automatisch über eine Standardsicherheitsgruppe. Wenn Sie beim Starten des Servers keine andere Sicherheitsgruppe oder Gruppen angeben, ordnen wir Ihrem Server die Standardsicherheitsgruppe zu.

Für die eingehenden Regeln für die Sicherheitsgruppe können Sie den SSH-Verkehr so konfigurieren, dass er Port 22, 2222, 22000 oder eine beliebige Kombination verwendet. Port 22 ist standardmäßig konfiguriert. Um Port 2222 oder Port 22000 zu verwenden, fügen Sie Ihrer Sicherheitsgruppe eine Regel für eingehenden Datenverkehr hinzu. Wählen Sie für den Typ Benutzerdefiniertes TCP aus, geben Sie dann entweder **`2222`** oder **`22000`** als Portbereich ein und geben Sie für die Quelle denselben CIDR-Bereich ein, den Sie für Ihre SSH-Port-22-Regel haben.

> **Note**
>
> Sie können Port 2223 auch für Clients verwenden, die TCP-Pickback-ACKs oder die Fähigkeit benötigen, dass das letzte Ack des TCP-3-Wege-Handshakes auch Daten enthält.
> Manche Client-Software ist möglicherweise nicht mit Port 2223 kompatibel, z. B. ein Client, bei dem der Server die SFTP-Identifikationszeichenfolge senden muss, bevor der Client dies tut.




g. (Optional) Aktivieren Sie für FIPS Enabled das Kontrollkästchen FIPS-fähiger Endpunkt, um sicherzustellen, dass der Endpunkt den Federal Information Processing Standards (FIPS) entspricht.

> **Note**
>
> FIPS-fähige Endpunkte sind nur in nordamerikanischen Regionen verfügbar. AWS Informationen zu den verfügbaren Regionen finden Sie unter AWS Transfer Family Endpunkte und Kontingente in der. Allgemeine AWS-Referenz Weitere Informationen zu FIPS finden Sie unter Federal Information Processing Standard (FIPS) 140-2.

h. Wählen Sie Weiter aus.

6. Gehen Sie unter Zusätzliche Details konfigurieren wie folgt vor:

   a. Wählen Sie für die CloudWatch Protokollierung eine der folgenden Optionen, um die CloudWatch Amazon-Protokollierung Ihrer Benutzeraktivitäten zu aktivieren:

      - Erstellen Sie eine neue Rolle, damit Transfer Family die IAM-Rolle automatisch erstellen kann, sofern Sie über die erforderlichen Berechtigungen zum Erstellen einer neuen Rolle verfügen. Die erstellte IAM-Rolle wird aufgerufen. `AWSTransferLoggingAccess`
      - Wählen Sie eine vorhandene Rolle aus, um eine bestehende IAM-Rolle aus Ihrem Konto auszuwählen. Wählen Sie unter Logging-Rolle die Rolle aus. Diese IAM-Rolle sollte

eine Vertrauensrichtlinie mit der Einstellung Service auf `transfer.amazonaws.com` enthalten.

Weitere Informationen zur CloudWatch Protokollierung finden Sie unterKonfigurieren Sie die CloudWatch Protokollierungsrolle.

> **Note**
>
> - Sie können keine Endbenutzeraktivitäten in anzeigen, CloudWatch wenn Sie keine Protokollierungsrolle angeben.
> - Wenn Sie keine CloudWatch Protokollierungsrolle einrichten möchten, wählen Sie Bestehende Rolle auswählen, aber wählen Sie keine Protokollierungsrolle aus.

b. Wählen Sie unter Optionen für kryptografische Algorithmen eine Sicherheitsrichtlinie aus, die die kryptografischen Algorithmen enthält, die für die Verwendung durch Ihren Server aktiviert sind.

> **Note**
>
> Standardmäßig ist die `TransferSecurityPolicy-2020-06` Sicherheitsrichtlinie an Ihren Server angehängt, sofern Sie keine andere auswählen.

Weitere Informationen zu Sicherheitsrichtlinien finden Sie unter Sicherheitsrichtlinien für AWS Transfer Family Server.

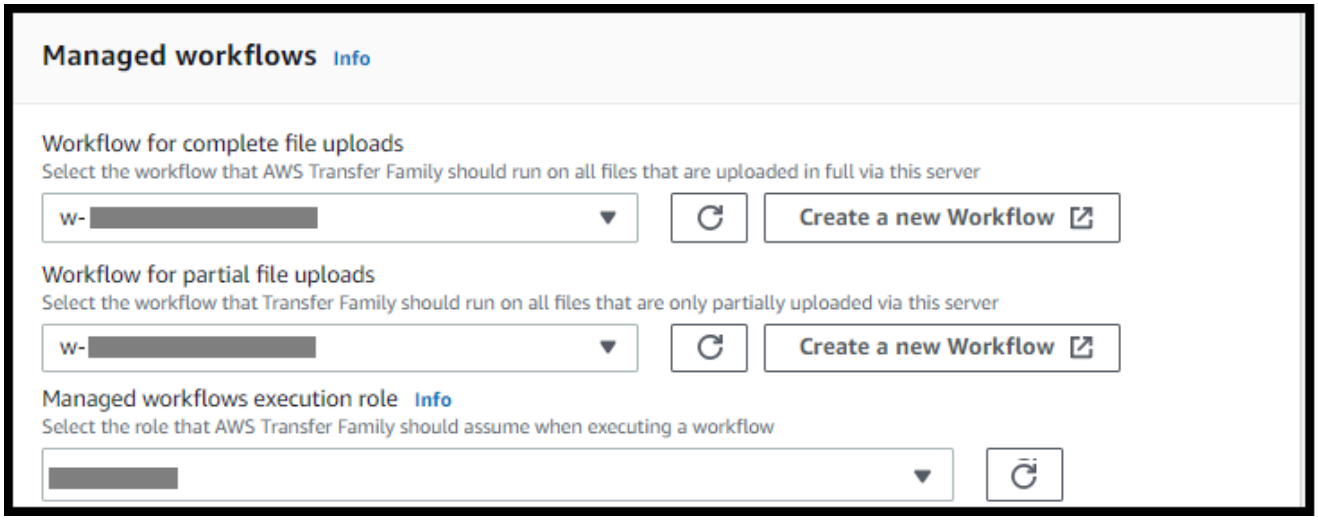
c. (Optional: Dieser Abschnitt ist nur für die Migration von Benutzern von einem vorhandenen SFTP-fähigen Server vorgesehen.) Geben Sie als Server-Hostschlüssel einen privaten RSA-, ED25519- oder ECDSA-Schlüssel ein, der zur Identifizierung Ihres Servers verwendet wird, wenn Clients über SFTP eine Verbindung zu ihm herstellen.

d. (Optional) Geben Sie für Tags für Schlüssel und Wert ein oder mehrere Tags als Schlüssel-Wert-Paare ein und wählen Sie dann Tag hinzufügen aus.

e. Wählen Sie Weiter aus.

f. (Optional) Wählen Sie für verwaltete Workflows Workflow-IDs (und eine entsprechende Rolle) aus, die Transfer Family bei der Ausführung des Workflows annehmen soll. Sie können einen Workflow auswählen, der bei einem vollständigen Upload ausgeführt wird,

und einen anderen, der bei einem teilweisen Upload ausgeführt werden soll. Weitere Informationen zur Verarbeitung Ihrer Dateien mithilfe verwalteter Workflows finden Sie unterAWS Transfer Family verwaltete Workflows.




7. Überprüfen Sie unter Überprüfen und erstellen Ihre Auswahl. Wenn Sie:

   - Wenn Sie eine davon bearbeiten möchten, wählen Sie neben dem Schritt Bearbeiten aus.

     > **Note**
     >
     > Sie müssen jeden Schritt nach dem Schritt, den Sie bearbeiten möchten, erneut überprüfen.

   - Haben Sie keine Änderungen vorgenommen, wählen Sie Server erstellen, um Ihren Server zu erstellen. Sie gelangen zur Seite Servers (Server) (siehe unten), auf der der neue Server aufgelistet ist.

Sie können die Server-ID auswählen, um die detaillierten Einstellungen des Servers zu sehen, den Sie gerade erstellt haben. Nachdem die Spalte Öffentliche IPv4-Adresse gefüllt wurde, wurden die von Ihnen angegebenen Elastic IP-Adressen erfolgreich mit dem Endpunkt Ihres Servers verknüpft.

> **Note**
>
> Wenn Ihr Server in einer VPC online ist, können nur die Subnetze geändert werden, und zwar nur über die UpdateServerAPI. Sie müssen den Server anhalten, um die Elastic-IP-Adressen des Serverendpunkts hinzuzufügen oder zu ändern.

## Ändern Sie den Endpunkttyp für Ihren Server

Wenn Sie bereits über einen Server verfügen, auf den über das Internet zugegriffen werden kann (d. h. einen öffentlichen Endpunkttyp hat), können Sie seinen Endpunkt in einen VPC-Endpunkt ändern.

> **Note**
>
> Wenn Sie einen vorhandenen Server in einer VPC als angezeigt haben, empfehlen wir IhnenVPC_ENDPOINT, ihn auf den neuen VPC-Endpunkttyp zu ändern. Mit diesem neuen Endpunkttyp müssen Sie keinen Network Load Balancer (NLB) mehr verwenden, um Elastic IP-Adressen mit dem Endpunkt Ihres Servers zu verknüpfen. Sie können auch VPC-Sicherheitsgruppen verwenden, um den Zugriff auf den Endpunkt Ihres Servers einzuschränken. Sie können den VPC_ENDPOINT Endpunkttyp jedoch weiterhin nach Bedarf verwenden.

Beim folgenden Verfahren wird davon ausgegangen, dass Sie über einen Server verfügen, der entweder den aktuellen öffentlichen Endpunkttyp oder den älteren VPC_ENDPOINT Typ verwendet.

Um den Endpunkttyp für Ihren Server zu ändern

1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Klicken Sie im Navigationsbereich auf Servers (Server).
3. Aktivieren Sie das Kontrollkästchen des Servers, für den Sie den Endpunkttyp ändern möchten.

   > **Important**
   >
   > Sie müssen den Server anhalten, bevor Sie dessen Endpunkttyp ändern können.

4. Wählen Sie für Actions (Aktionen) die Option Stop (Stopp).
5. Wählen Sie im daraufhin angezeigten Bestätigungsdialogfeld die Option Stopp aus, um zu bestätigen, dass Sie den Server beenden möchten.

   > **Note**
   >
   > Bevor Sie mit dem nächsten Schritt fortfahren, warten Sie unter Endpunktdetails, bis sich der Status des Servers in Offline ändert. Dies kann einige Minuten dauern.

Möglicherweise müssen Sie auf der Seite Server die Option Aktualisieren auswählen, um die Statusänderung zu sehen.
Sie können keine Änderungen vornehmen, bis der Server offline ist.

6. Wählen Sie unter Endpunktdetails die Option Bearbeiten aus.
7. Gehen Sie unter Endpunktkonfiguration bearbeiten wie folgt vor:

   a. Wählen Sie für Endpunkttyp bearbeiten die Option VPC hosted aus.

   b. Wählen Sie für Access eine der folgenden Optionen aus:

   - Intern, um Ihren Endpunkt nur für Clients zugänglich zu machen, die die privaten IP-Adressen des Endpunkts verwenden.
   - Internet Facing, um Ihren Endpunkt für Kunden über das öffentliche Internet zugänglich zu machen.

   > **Note**
   >
   > Wenn Sie sich für Internet Facing entscheiden, können Sie in jedem Subnetz oder Subnetzen eine bestehende Elastic IP-Adresse auswählen. Oder Sie können die VPC-Konsole (https://console.aws.amazon.com/vpc/) aufrufen, um eine oder mehrere neue Elastic IP-Adressen zuzuweisen. Diese Adressen können entweder Ihnen AWS oder Ihnen gehören. Sie können Elastic IP-Adressen, die bereits verwendet werden, nicht mit Ihrem Endpunkt verknüpfen.

   c. (Optional nur für Internetzugriff) Wählen Sie für Benutzerdefinierter Hostname eine der folgenden Optionen:

   - Amazon Route 53 DNS-Alias — wenn der Hostname, den Sie verwenden möchten, bei Route 53 registriert ist. Sie können dann den Hostnamen eingeben.
   - Anderes DNS — wenn der Hostname, den Sie verwenden möchten, bei einem anderen DNS-Anbieter registriert ist. Sie können dann den Hostnamen eingeben.
   - Keine — um den Endpunkt des Servers zu verwenden und keinen benutzerdefinierten Hostnamen zu verwenden. Der Server-Host-Name hat das Format `serverId.server.transfer.regionId.amazonaws.com`.

     Weitere Informationen zum Arbeiten mit benutzerdefinierten Hostnamen finden Sie unter. Mit benutzerdefinierten Hostnamen arbeiten

d. Wählen Sie für VPC eine vorhandene VPC-ID aus, oder wählen Sie Create a VPC, um eine neue VPC zu erstellen.

e. Wählen Sie im Abschnitt Availability Zones bis zu drei Availability Zones und zugehörige Subnetze aus. Wenn Internet Facing ausgewählt ist, wählen Sie auch eine Elastic IP-Adresse für jedes Subnetz aus.

> **Note**
>
> Wenn Sie maximal drei Availability Zones wünschen, aber nicht genügend verfügbar sind, erstellen Sie sie in der VPC-Konsole (https://console.aws.amazon.com/vpc/). Wenn Sie die Subnetze oder Elastic IP-Adressen ändern, dauert die Aktualisierung des Servers einige Minuten. Sie können Ihre Änderungen erst speichern, wenn das Server-Update abgeschlossen ist.

f. Wählen Sie Speichern.

8. Wählen Sie unter Aktionen die Option Start und warten Sie, bis sich der Status des Servers auf Online ändert. Dies kann einige Minuten dauern.

> **Note**
>
> Wenn Sie einen öffentlichen Endpunkttyp in einen VPC-Endpunkttyp geändert haben, beachten Sie, dass der Endpunkttyp für Ihren Server in VPC geändert wurde.

Die Standardsicherheitsgruppe ist an den Endpunkt angehängt. Informationen zum Ändern oder Hinzufügen zusätzlicher Sicherheitsgruppen finden Sie unter Sicherheitsgruppen erstellen.

## Einstellung der Verwendung von VPC_ENDPOINT

AWS Transfer Family stellt die Möglichkeit ein, Server `EndpointType=VPC_ENDPOINT` für neue Konten zu erstellen. AWS Ab dem 19. Mai 2021 können AWS Konten, die keine AWS Transfer Family Server mit dem Endpunkttyp von `VPC_ENDPOINT` besitzen, keine neuen Server mit `EndpointType=VPC_ENDPOINT` erstellen. Wenn Sie bereits Server besitzen, die den `VPC_ENDPOINT` Endpunkttyp verwenden, empfehlen wir Ihnen, `EndpointType=VPC` so bald wie möglich mit der Nutzung zu beginnen. Einzelheiten finden Sie unter Aktualisieren Sie Ihren AWS Transfer Family Serverendpunkttyp von VPC_ENDPOINT auf VPC.

Wir haben den neuen VPC Endpunkttyp Anfang 2020 eingeführt. Weitere Informationen finden Sie unter AWS Transfer Family Für SFTP werden VPC-Sicherheitsgruppen und Elastic IP-Adressen unterstützt. Dieser neue Endpunkt bietet mehr Funktionen und ist kostengünstiger und es fallen keine PrivateLink Gebühren an. Weitere Informationen finden Sie unter AWS PrivateLink Preise.

Dieser Endpunkttyp entspricht funktionell dem vorherigen Endpunkttyp (`VPC_ENDPOINT`). Sie können Elastic IP-Adressen direkt an den Endpunkt anhängen, um ihn mit dem Internet zu verbinden, und Sicherheitsgruppen für die Quell-IP-Filterung verwenden. Weitere Informationen finden Sie im Blogbeitrag Use IP Allow List to secure your AWS Transfer Family for SFTP servers.

Sie können diesen Endpunkt auch in einer gemeinsam genutzten VPC-Umgebung hosten. Weitere Informationen finden Sie unter Unterstützt AWS Transfer Family jetzt Shared Services-VPC-Umgebungen.

Zusätzlich zu SFTP können Sie die VPC verwenden, um FTPS und FTP `EndpointType` zu aktivieren. Wir haben nicht vor, diese Funktionen und FTPS/FTP-Unterstützung hinzuzufügen. `EndpointType=VPC_ENDPOINT` Wir haben diesen Endpunkttyp auch als Option aus der Konsole entfernt. AWS Transfer Family

Sie können den Endpunkttyp für Ihren Server mithilfe der Transfer Family Family-Konsole AWS CLI, API, SDKs oder AWS CloudFormationändern. Informationen zum Ändern des Endpunkttyps Ihres Servers finden Sie unterAktualisierung des AWS Transfer Family Serverendpunkttyps von VPC_ENDPOINT auf VPC.

Wenn Sie Fragen haben, wenden Sie sich an AWS Support Ihr AWS Account-Team.

> **Note**
>
> Wir haben nicht vor, diese Funktionen und FTPS- oder FTP-Unterstützung zu EndpointType =VPC_ENDPOINT hinzuzufügen. Wir bieten es nicht mehr als Option auf der Konsole an. AWS Transfer Family

Wenn du weitere Fragen hast, kannst du uns über AWS Support oder dein Account-Team kontaktieren.

# Aktualisierung des AWS Transfer Family Serverendpunkttyps von VPC_ENDPOINT auf VPC

Sie können die AWS Management Console AWS CloudFormation, oder die Transfer Family Family-API verwenden, um die Daten `EndpointType` von bis eines Servers `VPC_ENDPOINT` zu aktualisieren`VPC`. In den folgenden Abschnitten finden Sie detaillierte Verfahren und Beispiele für die Verwendung jeder dieser Methoden zur Aktualisierung eines Serverendpunkttyps. Wenn Sie Server in mehreren AWS Regionen und in mehreren AWS Konten haben, können Sie das Beispielskript im folgenden Abschnitt mit Änderungen verwenden, um Server zu identifizieren, die den `VPC_ENDPOINT` Typ verwenden, den Sie aktualisieren müssen.

Themen


- Identifizieren von Servern anhand des VPC_ENDPOINT Endpunkttyps
- Aktualisierung des Serverendpunkt-Typs mit dem AWS Management Console
- Den Serverendpunkttyp aktualisieren mit AWS CloudFormation
- Den Server EndpointType mithilfe der API aktualisieren


## Identifizieren von Servern anhand des **`VPC_ENDPOINT`** Endpunkttyps

Mithilfe von können Sie feststellen, welche Server das `VPC_ENDPOINT` verwenden AWS Management Console.

### Um Server zu identifizieren, die den **`VPC_ENDPOINT`** Endpunkttyp mithilfe der Konsole verwenden

1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Wählen Sie im Navigationsbereich Server aus, um die Liste der Server in Ihrem Konto in dieser Region anzuzeigen.
3. Sortieren Sie die Liste der Server nach dem Endpunkttyp, um zu sehen, dass alle Server diese verwenden`VPC_ENDPOINT`.

### Um Server zu identifizieren, die mehrere **`VPC_ENDPOINT`**AWS Regionen und Konten verwenden

Wenn Sie Server in mehreren AWS Regionen und in mehreren AWS Konten haben, können Sie das folgende Beispielskript mit Änderungen verwenden, um Server zu identifizieren, die den `VPC_ENDPOINT` Endpunkttyp verwenden. Das Beispielskript verwendet die ListServers API-Aufrufe von Amazon EC2 DescribeRegionsund Transfer Family, um eine Liste der Server-IDs und Regionen

all Ihrer Server abzurufen, die Sie verwendenVPC_ENDPOINT. Wenn Sie viele AWS Konten haben, können Sie Ihre Konten mithilfe einer IAM-Rolle mit schreibgeschütztem Auditor-Zugriff durchsuchen, wenn Sie sich mithilfe von Sitzungsprofilen bei Ihrem Identitätsanbieter authentifizieren.

1. Es folgt ein einfaches Beispiel.

```
import boto3

profile = input("Enter the name of the AWS account you'll be working in: ")
session = boto3.Session(profile_name=profile)

ec2 = session.client("ec2")

regions = ec2.describe_regions()

for region in regions['Regions']:
    region_name = region['RegionName']
    if region_name=='ap-northeast-3': #https://github.com/boto/boto3/issues/1943
        continue
    transfer = session.client("transfer", region_name=region_name)
    servers = transfer.list_servers()
    for server in servers['Servers']:
       if server['EndpointType']=='VPC_ENDPOINT':
           print(server['ServerId'], region_name)
```

2. Nachdem Sie die Liste der zu aktualisierenden Server erstellt haben, können Sie eine der in den folgenden Abschnitten beschriebenen Methoden verwenden, um den Server `EndpointType` zu aktualisierenVPC.

Aktualisierung des Serverendpunkt-Typs mit dem AWS Management Console

1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Klicken Sie im Navigationsbereich auf Servers (Server).
3. Aktivieren Sie das Kontrollkästchen des Servers, für den Sie den Endpunkttyp ändern möchten.

> **Important**
>
> Sie müssen den Server anhalten, bevor Sie dessen Endpunkttyp ändern können.

4. Wählen Sie für Actions (Aktionen) die Option Stop (Stopp).

5. Wählen Sie im daraufhin angezeigten Bestätigungsdialogfeld die Option Stopp aus, um zu bestätigen, dass Sie den Server beenden möchten.

   > **Note**
   >
   > Bevor Sie mit dem nächsten Schritt fortfahren, warten Sie, bis sich der Status des Servers auf Offline ändert. Dies kann einige Minuten dauern. Möglicherweise müssen Sie auf der Seite Server die Option Aktualisieren auswählen, um die Statusänderung zu sehen.

6. Nachdem sich der Status auf Offline geändert hat, wählen Sie den Server aus, auf dem die Serverdetailseite angezeigt werden soll.
7. Wählen Sie im Abschnitt Endpunktdetails die Option Bearbeiten aus.
8. Wählen Sie VPC Hosted für den Endpoint-Typ aus.
9. Wählen Sie Speichern.
10. Wählen Sie für Aktionen die Option Start und warten Sie, bis sich der Status des Servers auf Online ändert. Dies kann einige Minuten dauern.

### Den Serverendpunkttyp aktualisieren mit AWS CloudFormation

In diesem Abschnitt wird beschrieben, wie AWS CloudFormation Sie das auf einem Server aktualisieren `EndpointType` können`VPC`. Verwenden Sie dieses Verfahren für Transfer Family Family-Server, mit denen Sie bereitgestellt haben AWS CloudFormation. In diesem Beispiel wird die ursprüngliche AWS CloudFormation Vorlage, die für die Bereitstellung des Transfer Family Family-Servers verwendet wurde, wie folgt dargestellt:

```
AWS TemplateFormatVersion: '2010-09-09'
Description: 'Create AWS Transfer Server with VPC_ENDPOINT endpoint type'
Parameters:
  SecurityGroupId:
    Type: AWS::EC2::SecurityGroup::Id
  SubnetIds:
    Type: List<AWS::EC2::Subnet::Id>
  VpcId:
    Type: AWS::EC2::VPC::Id
Resources:
  TransferServer:
    Type: AWS::Transfer::Server
    Properties:
```

```
      Domain: S3
      EndpointDetails:
        VpcEndpointId: !Ref VPCEndpoint
      EndpointType: VPC_ENDPOINT
      IdentityProviderType: SERVICE_MANAGED
      Protocols:
        - SFTP
  VPCEndpoint:
    Type: AWS::EC2::VPCEndpoint
    Properties:
      ServiceName: com.amazonaws.us-east-1.transfer.server
      SecurityGroupIds:
        - !Ref SecurityGroupId
      SubnetIds:
        - !Select [0, !Ref SubnetIds]
        - !Select [1, !Ref SubnetIds]
        - !Select [2, !Ref SubnetIds]
      VpcEndpointType: Interface
      VpcId: !Ref VpcId
```

Die Vorlage wurde mit den folgenden Änderungen aktualisiert:

- Das `EndpointType` wurde geändert in`VPC`.
- Die `AWS::EC2::VPCEndpoint` Ressource wurde entfernt.
- Die `SecurityGroupIdSubnetIds`, und `VpcId` wurden in den `EndpointDetails` Abschnitt der `AWS::Transfer::Server` Ressource verschoben,
- Die `VpcEndpointId` Eigenschaft von `EndpointDetails` wurde entfernt.

Die aktualisierte Vorlage sieht wie folgt aus:

```
AWS TemplateFormatVersion: '2010-09-09'
Description: 'Create AWS Transfer Server with VPC endpoint type'
Parameters:
  SecurityGroupId:
    Type: AWS::EC2::SecurityGroup::Id
  SubnetIds:
    Type: List<AWS::EC2::Subnet::Id>
  VpcId:
    Type: AWS::EC2::VPC::Id
Resources:
  TransferServer:
```

```
    Type: AWS::Transfer::Server
    Properties:
      Domain: S3
      EndpointDetails:
        SecurityGroupIds:
          - !Ref SecurityGroupId
        SubnetIds:
          - !Select [0, !Ref SubnetIds]
          - !Select [1, !Ref SubnetIds]
          - !Select [2, !Ref SubnetIds]
        VpcId: !Ref VpcId
      EndpointType: VPC
      IdentityProviderType: SERVICE_MANAGED
      Protocols:
        - SFTP
```

Um den Endpunkttyp von Transfer Family Family-Servern zu aktualisieren, die bereitgestellt werden mit AWS CloudFormation

1. Stoppen Sie den Server, den Sie aktualisieren möchten, indem Sie die folgenden Schritte ausführen.

   a. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.

   b. Klicken Sie im Navigationsbereich auf Servers (Server).

   c. Aktivieren Sie das Kontrollkästchen des Servers, für den Sie den Endpunkttyp ändern möchten.

   > **Important**
   >
   > Sie müssen den Server anhalten, bevor Sie dessen Endpunkttyp ändern können.

   d. Wählen Sie für Actions (Aktionen) die Option Stop (Stopp).

   e. Wählen Sie im daraufhin angezeigten Bestätigungsdialogfeld die Option Stopp aus, um zu bestätigen, dass Sie den Server beenden möchten.

   > **Note**
   >
   > Bevor Sie mit dem nächsten Schritt fortfahren, warten Sie, bis sich der Status des Servers auf Offline ändert. Dies kann einige Minuten dauern. Möglicherweise

müssen Sie auf der Seite Server die Option Aktualisieren auswählen, um die Statusänderung zu sehen.

2. Aktualisieren Sie den CloudFormation Stack

   a. Öffnen Sie die AWS CloudFormation Konsole unter https://console.aws.amazon.com/cloudformation.

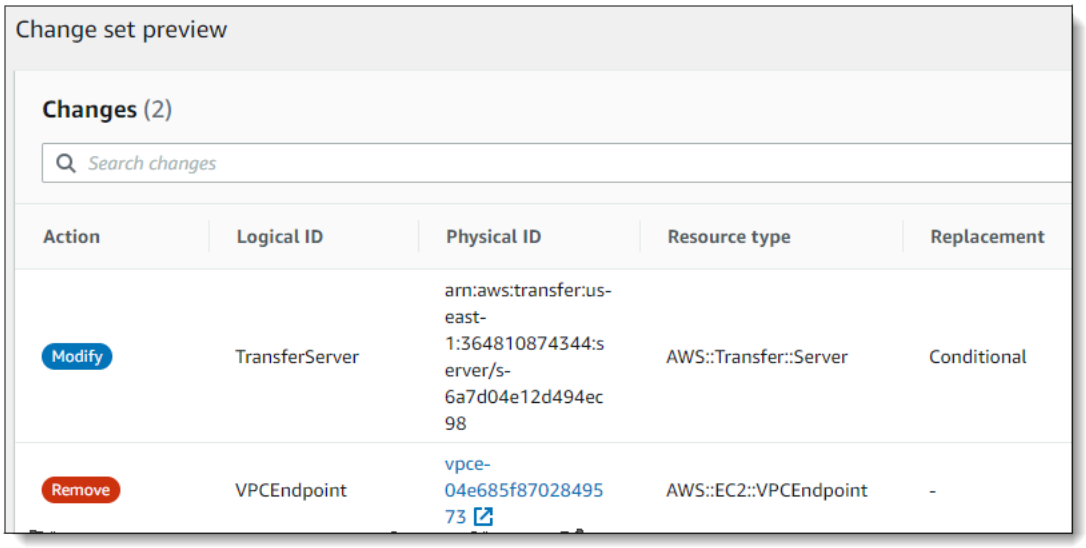
   b. Wählen Sie den Stack aus, der zur Erstellung des Transfer Family Family-Servers verwendet wurde.

   c. Wählen Sie Aktualisieren.

   d. Wählen Sie Aktuelle Vorlage ersetzen

   e. Laden Sie die neue Vorlage hoch. CloudFormation Mithilfe von Änderungssätzen können Sie verstehen, wie sich Vorlagenänderungen auf laufende Ressourcen auswirken, bevor Sie sie implementieren. In diesem Beispiel wird die Transfer-Serverressource geändert und die VPCEnPoint-Ressource wird entfernt. Der VPC-Endpunktserver erstellt in Ihrem Namen einen VPC-Endpunkt und ersetzt die ursprüngliche `VPCEndpoint` Ressource.

      Nach dem Hochladen der neuen Vorlage sieht der Änderungssatz etwa wie folgt aus:




   f. Aktualisieren Sie den Stack.

3. Sobald das Stack-Update abgeschlossen ist, navigieren Sie zur Transfer Family Family-Verwaltungskonsole unter https://console.aws.amazon.com/transfer/.

4. Starten Sie den Server neu. Wählen Sie den Server aus, auf dem Sie aktualisiert haben AWS CloudFormation, und wählen Sie dann im Menü Aktionen die Option Start.

Den Server EndpointType mithilfe der API aktualisieren

Sie können den Befehl describe-server oder den AWS CLI UpdateServer API-Befehl verwenden. Das folgende Beispielskript stoppt den Transfer Family Family-Server, aktualisiert den EndpointType, entfernt den VPC_ENDPOINT und startet den Server.

```
import boto3
import time

profile = input("Enter the name of the AWS account you'll be working in: ")
region_name = input("Enter the AWS Region you're working in: ")
server_id = input("Enter the AWS Transfer Server Id: ")

session = boto3.Session(profile_name=profile)

ec2 = session.client("ec2", region_name=region_name)
transfer = session.client("transfer", region_name=region_name)

group_ids=[]

transfer_description = transfer.describe_server(ServerId=server_id)
if transfer_description['Server']['EndpointType']=='VPC_ENDPOINT':
    transfer_vpc_endpoint = transfer_description['Server']['EndpointDetails']
['VpcEndpointId']
    transfer_vpc_endpoint_descriptions =
 ec2.describe_vpc_endpoints(VpcEndpointIds=[transfer_vpc_endpoint])
    for transfer_vpc_endpoint_description in
 transfer_vpc_endpoint_descriptions['VpcEndpoints']:
        subnet_ids=transfer_vpc_endpoint_description['SubnetIds']
        group_id_list=transfer_vpc_endpoint_description['Groups']
        vpc_id=transfer_vpc_endpoint_description['VpcId']
        for group_id in group_id_list:
             group_ids.append(group_id['GroupId'])
   if transfer_description['Server']['State']=='ONLINE':
        transfer_stop = transfer.stop_server(ServerId=server_id)
        print(transfer_stop)
        time.sleep(300) #safe
        transfer_update =
 transfer.update_server(ServerId=server_id,EndpointType='VPC',EndpointDetails={'SecurityGroupId
        print(transfer_update)
        time.sleep(10)
        transfer_start = transfer.start_server(ServerId=server_id)
        print(transfer_start)
```

```
        delete_vpc_endpoint =
ec2.delete_vpc_endpoints(VpcEndpointIds=[transfer_vpc_endpoint])
```

## Mit benutzerdefinierten Hostnamen arbeiten

Ihr Server-Hostname ist der Hostname, den Ihre Benutzer in ihren Clients eingeben, wenn sie eine Verbindung zu Ihrem Server herstellen. Sie können eine benutzerdefinierte Domain verwenden, die Sie für Ihren Server-Hostnamen registriert haben, wenn Sie damit arbeiten. AWS Transfer Family Sie könnten beispielsweise einen benutzerdefinierten Hostnamen wie verwenden. `mysftpserver.mysubdomain.domain.com`

Um Traffic von Ihrer registrierten benutzerdefinierten Domain zu Ihrem Serverendpunkt umzuleiten, können Sie Amazon Route 53 oder einen beliebigen Domain Name System (DNS) -Anbieter verwenden. Route 53 ist der DNS-Dienst, der AWS Transfer Family nativ unterstützt wird.

Themen

- Verwenden Sie Amazon Route 53 als Ihren DNS-Anbieter
- Verwenden Sie andere DNS-Anbieter
- Benutzerdefinierte Hostnamen für Server, die nicht von der Konsole erstellt wurden

Auf der Konsole können Sie eine der folgenden Optionen für die Einrichtung eines benutzerdefinierten Hostnamens wählen:

- Amazon Route 53 DNS-Alias — wenn der Hostname, den Sie verwenden möchten, bei Route 53 registriert ist. Sie können dann den Hostnamen eingeben.
- Anderes DNS — wenn der Hostname, den Sie verwenden möchten, bei einem anderen DNS-Anbieter registriert ist. Sie können dann den Hostnamen eingeben.
- Keine — um den Endpunkt des Servers zu verwenden und keinen benutzerdefinierten Hostnamen zu verwenden.

Sie legen diese Option fest, wenn Sie einen neuen Server erstellen oder die Konfiguration eines vorhandenen Servers bearbeiten. Weitere Hinweise zum Erstellen eines neuen Servers finden Sie unterSchritt 2: Erstellen Sie einen SFTP-fähigen Server. Weitere Hinweise zum Bearbeiten der Konfiguration eines vorhandenen Servers finden Sie unterServerdetails bearbeiten.

Weitere Informationen zur Verwendung Ihrer eigenen Domain für den Server-Hostnamen und zur Verwendung AWS Transfer Family von Route 53 finden Sie in den folgenden Abschnitten.

## Verwenden Sie Amazon Route 53 als Ihren DNS-Anbieter

Wenn Sie einen Server erstellen, können Sie Amazon Route 53 als Ihren DNS-Anbieter verwenden. Bevor Sie eine Domain mit Route 53 verwenden, registrieren Sie die Domain. Weitere Informationen finden Sie unter So funktioniert die Domainregistrierung im Amazon Route 53-Entwicklerhandbuch.

Wenn Sie Route 53 verwenden, um DNS-Routing für Ihren Server bereitzustellen, AWS Transfer Family verwendet es den benutzerdefinierten Hostnamen, den Sie eingegeben haben, um die Hosting-Zone zu extrahieren. Beim AWS Transfer Family Extrahieren einer Hosting-Zone können drei Dinge passieren:

1. Wenn Route 53 neu für Sie ist und Sie noch keine Hosting-Zone haben, AWS Transfer Family fügt Sie eine neue Hosting-Zone und einen `CNAME` Datensatz hinzu. Der Wert dieses `CNAME` Datensatzes ist der Endpunkt-Hostname für Ihren Server. Ein CNAME ist ein alternativer Domänenname.
2. Wenn Sie eine gehostete Zone in Route 53 ohne `CNAME` Datensätze haben, AWS Transfer Family fügt der Hosting-Zone einen `CNAME` Datensatz hinzu.
3. Wenn der Service erkennt, dass bereits ein `CNAME`-Datensatz in der gehosteten Zone vorhanden ist, wird Ihnen eine Fehlermeldung angezeigt, der angibt, dass bereits ein `CNAME`-Datensatz vorhanden ist. Ändern Sie in diesem Fall den Wert des `CNAME` Datensatzes in den Hostnamen Ihres Servers.

Weitere Informationen zu gehosteten Zonen in Route 53 finden Sie unter Gehostete Zone im Amazon Route 53 Developer Guide.

## Verwenden Sie andere DNS-Anbieter

Wenn Sie einen Server erstellen, können Sie auch andere DNS-Anbieter als Amazon Route 53 verwenden. Wenn Sie einen alternativen DNS-Anbieter verwenden, müssen Sie sicherstellen, dass der Datenverkehr aus Ihrer Domäne zum -Server-Endpunkt geleitet wird.

Stellen Sie dazu Ihre Domain auf den Endpunkt-Hostnamen für den Server ein. Ein Endpunkt-Hostname sieht in der Konsole wie folgt aus:

```
serverid.server.transfer.region.amazonaws.com
```

> **Note**
>
> Wenn Ihr Server über einen VPC-Endpunkt verfügt, unterscheidet sich das Format für den Hostnamen von dem oben beschriebenen. Um Ihren VPC-Endpunkt zu finden, wählen Sie die VPC auf der Detailseite des Servers und dann die VPC-Endpunkt-ID im VPC-Dashboard aus. Der Endpunkt ist der erste DNS-Name der aufgelisteten Namen.

## Benutzerdefinierte Hostnamen für Server, die nicht von der Konsole erstellt wurden

Wenn Sie einen Server mit AWS Cloud Development Kit (AWS CDK) AWS CloudFormation, oder über die CLI erstellen, müssen Sie ein Tag hinzufügen, wenn dieser Server einen benutzerdefinierten Hostnamen haben soll. Wenn Sie mithilfe der Konsole einen Transfer Family Family-Server erstellen, erfolgt das Tagging automatisch.

> **Note**
>
> Sie müssen auch einen DNS-Eintrag erstellen, um den Verkehr von Ihrer Domain zu Ihrem Serverendpunkt umzuleiten. Einzelheiten finden Sie unter Arbeiten mit Datensätzen im Amazon Route 53-Entwicklerhandbuch.

Verwenden Sie die folgenden Schlüssel für Ihren benutzerdefinierten Hostnamen:

- Fügen Sie hinzu`transfer:customHostname`, um den benutzerdefinierten Hostnamen in der Konsole anzuzeigen.
- Wenn Sie Route 53 als Ihren DNS-Anbieter verwenden, fügen Sie hinzu`transfer:route53HostedZoneId`. Dieses Tag verknüpft den benutzerdefinierten Hostnamen mit Ihrer Route 53 Hosted Zone ID.

Geben Sie den folgenden CLI-Befehl ein, um den benutzerdefinierten Hostnamen hinzuzufügen.

```
aws transfer tag-resource --arn arn:aws:transfer:region:AWS-Konto:server/server-ID --
tags Key=transfer:customHostname,Value="custom-host-name"
```

Beispielsweise:

```
aws transfer tag-resource --arn arn:aws:transfer:us-east-1:111122223333:server/
s-1234567890abcdef0 --tags Key=transfer:customHostname,Value="abc.example.com"
```

Wenn Sie Route 53 verwenden, geben Sie den folgenden Befehl ein, um Ihren benutzerdefinierten Hostnamen mit Ihrer Route 53 Hosted Zone ID zu verknüpfen.

```
aws transfer tag-resource --arn server-ARN:server/server-ID --tags
 Key=transfer:route53HostedZoneId,Value=HOSTED-ZONE-ID
```

Beispielsweise:

```
aws transfer tag-resource --arn arn:aws:transfer:us-east-1:111122223333:server/
s-1234567890abcdef0 --tags Key=transfer:route53HostedZoneId,Value=ABCDE1111222233334444
```

Gehen Sie von den Beispielwerten aus dem vorherigen Befehl aus und führen Sie den folgenden Befehl aus, um Ihre Tags anzuzeigen:

```
aws transfer list-tags-for-resource --arn arn:aws:transfer:us-
east-1:111122223333:server/s-1234567890abcdef0
```

```
"Tags": [
   {
      "Key": "transfer:route53HostedZoneId",
      "Value": "/hostedzone/ABCDE1111222233334444"
   },
   {
      "Key": "transfer:customHostname",
      "Value": "abc.example.com"
   }
 ]
```

> **Note**
>
> Ihre öffentlichen, gehosteten Zonen und ihre IDs sind auf Amazon Route 53 verfügbar. Melden Sie sich bei der Route 53-Konsole an AWS Management Console und öffnen Sie sie unter https://console.aws.amazon.com/route53/.

# Übertragung von Dateien über einen Serverendpunkt mit einem Client

Sie übertragen Dateien über den AWS Transfer Family Dienst, indem Sie den Übertragungsvorgang in einem Client angeben. AWS Transfer Family unterstützt die folgenden Clients:

- Wir unterstützen Version 3 des SFTP-Protokolls.
- OpenSSH (macOS und Linux)

> **Note**
>
> Dieser Client funktioniert nur mit Servern, die für Secure Shell (SSH) File Transfer Protocol (SFTP) aktiviert sind.

- WinSCP (nur Microsoft Windows)
- Cyberduck (Windows, macOS und Linux)
- FileZilla (Windows, macOS und Linux)

Die folgenden Einschränkungen gelten für jeden Client:

- Die maximale Anzahl gleichzeitiger, gemultiplexter SFTP-Sitzungen pro Verbindung beträgt 10.
- Es gibt zwei Timeout-Werte für SFTP/FTP/FTPS-Verbindungen. Bei Verbindungen im Leerlauf beträgt der Timeout-Wert 1800 Sekunden (30 Minuten). Wenn nach Ablauf des Zeitraums keine Aktivität mehr stattfindet, wird die Verbindung zum Client möglicherweise unterbrochen. Es gibt auch ein Timeout von 300 Sekunden (5 Minuten), wenn ein Client überhaupt nicht reagiert.
- Amazon S3 und Amazon EFS (aufgrund des NFSv4-Protokolls) erfordern, dass Dateinamen in UTF-8-Kodierung vorliegen. Die Verwendung einer anderen Kodierung kann zu unerwarteten Ergebnissen führen. Informationen zu Amazon S3 finden Sie unter Richtlinien zur Benennung von Objektschlüsseln.
- Für das File Transfer Protocol over SSL (FTPS) wird nur der Modus Explizit unterstützt. Der implizite Modus wird nicht unterstützt.
- Für File Transfer Protocol (FTP) und FTPS wird nur der passive Modus unterstützt.
- Für FTP und FTPS wird nur der STREAM-Modus unterstützt.
- Für FTP und FTPS wird nur der Bild-/Binärmodus unterstützt.

- Für FTP und FTPS ist TLS — PROT C (ungeschützt) TLS für die Datenverbindung die Standardeinstellung, aber PROT C wird im FTPS-Protokoll nicht unterstützt. AWS Transfer Family Für FTPS müssen Sie also PROT P ausgeben, damit Ihr Datenvorgang akzeptiert wird.
- Wenn Sie Amazon S3 für den Speicher Ihres Servers verwenden und Ihr Client die Option enthält, mehrere Verbindungen für eine einzelne Übertragung zu verwenden, stellen Sie sicher, dass Sie die Option deaktivieren. Andernfalls können große Datei-Uploads auf unvorhersehbare Weise fehlschlagen. Beachten Sie, dass EFS mehrere Verbindungen für eine einzige Übertragung unterstützt, wenn Sie Amazon EFS als Speicher-Backend verwenden.

Im Folgenden finden Sie eine Liste der verfügbaren Befehle für FTP und FTPS:

| Verfügbare Befehle | | | | | |
|---|---|---|---|---|---|
| ARBEIT | KUNSTSTÜCK | AM MEISTEN | PASS | RETR | GESCHICHTE |
| AUTH | LANG | MKD | PASV | RMD | STOU |
| CUP | LIST | MODE | PBSZ | RNFR | STRU |
| CWD | MDTM | NLST | HAFEN | ENTO | SYSTEM |
| DELE | MFMT | NEIN | PWD | SIZE | TYPE |
| EPSV | MLSD | WÄHLT | QUIT | STAT | USER |

> **Note**
>
> APPE wird nicht unterstützt.

Für SFTP werden die folgenden Operationen derzeit nicht für Benutzer unterstützt, die das logische Home-Verzeichnis auf Servern verwenden, die Amazon Elastic File System (Amazon EFS) verwenden.

| SFTP-Befehle werden nicht unterstützt | | | |
|---|---|---|---|
| SSH_FXP_READLINK | SSH_FXP_SYMLINK | SSH_FXP_STAT, wenn die angeforderte Datei ein Symlink ist | SSH_FXP_REALPATH, wenn der angeforderte Pfad irgendwelche Symlink-Komponenten enthält |

Generieren Sie ein öffentlich-privates key pair

Bevor Sie eine Datei übertragen können, müssen Sie über ein öffentlich-privates key pair verfügen. Wenn Sie noch kein key pair generiert haben, finden Sie weitere Informationen unterGenerieren Sie SSH-Schlüssel für vom Service verwaltete Benutzer.

Themen

- Verfügbare SFTP/FTPS/FTP-Befehle
- Finden Sie Ihren Amazon VPC-Endpunkt
- Vermeiden Sie Fehler setstat
- OpenSSH verwenden
- Verwenden Sie WinSCP
- Verwenden Sie Cyberduck
- Verwenden FileZilla
- Verwenden Sie einen Perl-Client
- Verarbeitung nach dem Upload

## Verfügbare SFTP/FTPS/FTP-Befehle

In der folgenden Tabelle werden die verfügbaren Befehle für die Protokolle SFTP AWS Transfer Family, FTPS und FTP beschrieben.

> **Note**
>
> In der Tabelle werden Dateien und Verzeichnisse für Amazon S3 erwähnt, das nur Buckets und Objekte unterstützt: Es gibt keine Hierarchie. Sie können jedoch Präfixe in

Objektschlüsselnamen verwenden, um eine Hierarchie zu implizieren und Ihre Daten ähnlich wie Ordner zu organisieren. Dieses Verhalten wird unter Arbeiten mit Objektmetadaten im Amazon Simple Storage Service-Benutzerhandbuch beschrieben.

SFTP/FTPS/FTP-Befehle

| Befehl | Amazon S3 | Amazon EFS |
|---|---|---|
| `cd` | Unterstützt | Unterstützt |
| `chgrp` | Nicht unterstützt | Unterstützt (oder nur) `root` `owner` |
| `chmod` | Nicht unterstützt | Wird unterstützt (`root`nur) |
| `chmtime` | Nicht unterstützt | Unterstützt |
| `chown` | Nicht unterstützt | Unterstützt (`root`nur) |
| `get` | Unterstützt | Unterstützt (einschließlich der Auflösung symbolischer Links) |
| `ln -s` | Nicht unterstützt | Unterstützt |
| `ls/dir` | Unterstützt | Unterstützt |
| `mkdir` | Unterstützt | Unterstützt |
| `put` | Unterstützt | Unterstützt |
| `pwd` | Unterstützt | Unterstützt |
| `rename` | Wird nur für Dateien unterstützt | Unterstützt<br>**Note**<br>Ein Umbenennen, das eine bestehende Datei oder ein vorhandenes Verzeichnis überschre |

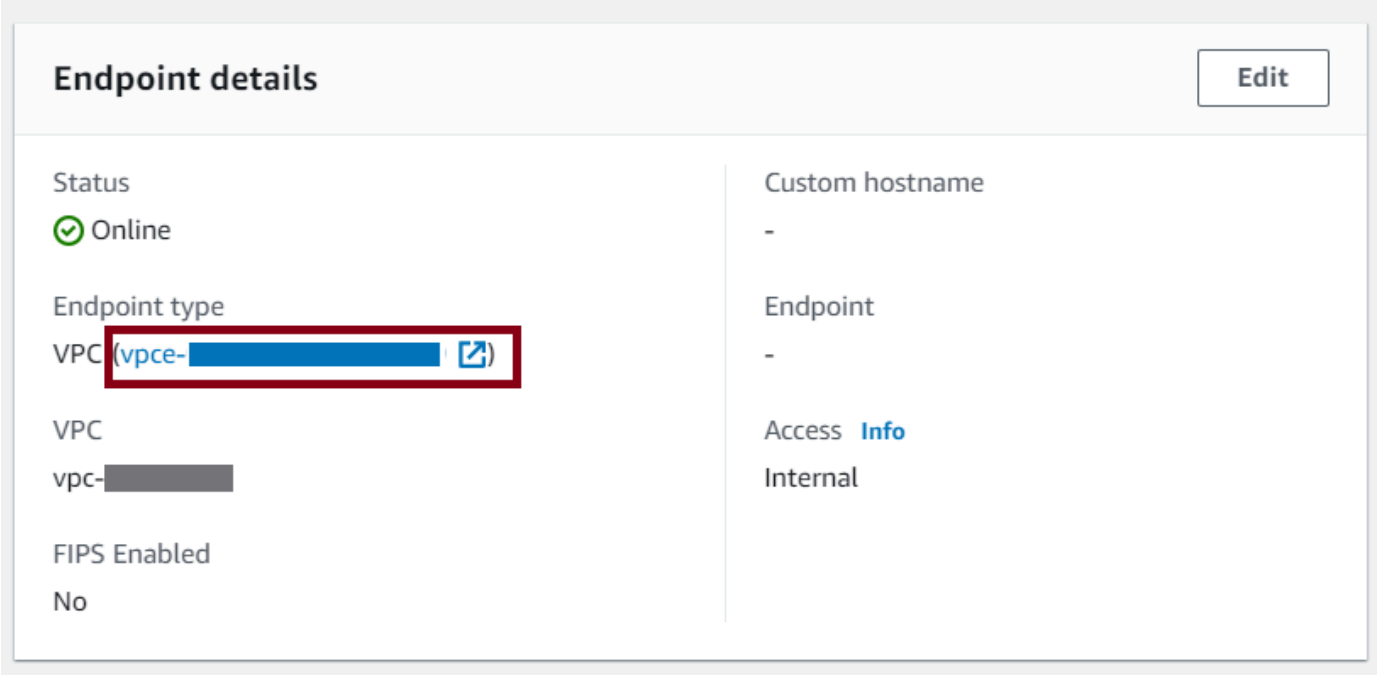
| Befehl | Amazon S3 | Amazon EFS |
| --- | --- | --- |
| | | iben würde, wird nicht unterstützt. |
| `rm` | Unterstützt | Unterstützt |
| `rmdir` | Unterstützt (nur leere Verzeichnisse) | Unterstützt |
| `version` | Unterstützt | Unterstützt |

## Finden Sie Ihren Amazon VPC-Endpunkt

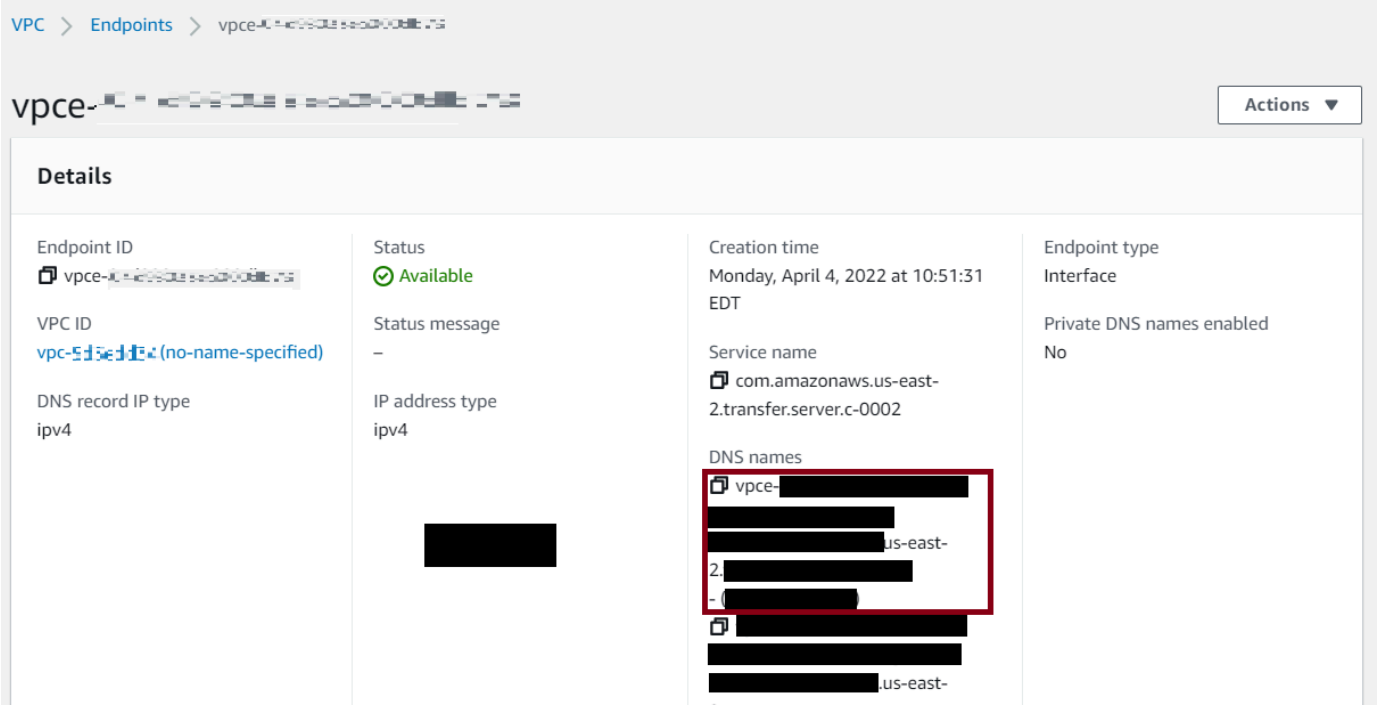
Wenn der Endpunkttyp für Ihren Transfer Family Family-Server VPC ist, ist es nicht einfach, den Endpunkt zu identifizieren, der für die Übertragung von Dateien verwendet werden soll. Gehen Sie in diesem Fall wie folgt vor, um Ihren Amazon VPC-Endpunkt zu finden.

Finden Sie Ihren Amazon VPC-Endpunkt

1. Navigieren Sie zur Detailseite Ihres Servers.
2. Wählen Sie im Bereich Endpunktdetails die VPC aus.



3. Wählen Sie im Amazon VPC-Dashboard die VPC-Endpunkt-ID aus.
4. In der Liste der DNS-Namen ist Ihr Serverendpunkt der erste, der aufgeführt ist.




## Vermeiden Sie Fehler `setstat`

Einige SFTP-Dateiübertragungsclients können versuchen, die Attribute von Remotedateien, einschließlich Zeitstempel und Berechtigungen, mithilfe von Befehlen wie SETSTAT beim Hochladen der Datei zu ändern. Diese Befehle sind jedoch nicht mit Objekt-Speichersystemen wie Amazon S3 kompatibel. Aufgrund dieser Inkompatibilität können Datei-Uploads von diesen Clients zu Fehlern führen, selbst wenn die Datei anderweitig erfolgreich hochgeladen wurde.

- Verwenden Sie beim Aufrufen der `UpdateServer` API `CreateServer` oder die `ProtocolDetails` Option, um den Fehler `SetStatOption` zu ignorieren, der generiert wird, wenn der Client versucht, SETSTAT für eine Datei zu verwenden, die Sie in einen S3-Bucket hochladen.
- Setzen Sie den Wert auf `ENABLE_NO_OP`, damit der Transfer-Family-Server den SETSTAT-Befehl ignoriert und Dateien hochlädt, ohne Änderungen an Ihrem SFTP-Client vornehmen zu müssen.
- Beachten Sie, dass die `SetStatOption` `ENABLE_NO_OP` Einstellung zwar den Fehler ignoriert, aber einen Protokolleintrag in CloudWatch Logs generiert, sodass Sie feststellen können, wann der Client einen SETSTAT-Aufruf durchführt.

Die API-Details für diese Option finden Sie unter. ProtocolDetails

## OpenSSH verwenden

Unten wird beschrieben, wie Dateien in der Befehlszeile mit OpenSSH übertragen werden.

> **Note**
>
> Dieser Client funktioniert nur mit einem SFTP-fähigen Server.

Um Dateien AWS Transfer Family mit dem OpenSSH-Befehlszeilenprogramm zu übertragen

1. Öffnen Sie unter Linux, macOS oder Windows ein Befehlsterminal.
2. Geben Sie an der Eingabeaufforderung den folgenden Befehl ein:

   ```
   sftp -i transfer-key sftp_user@service_endpoint
   ```

   Im vorherigen Befehl *sftp_user* ist dies der Benutzername und *transfer-key* der private SSH-Schlüssel. Hier *service_endpoint* ist der Endpunkt des Servers, wie er in der AWS Transfer Family Konsole für den ausgewählten Server angezeigt wird.

   > **Note**
   >
   > Dieser Befehl verwendet Einstellungen, die in der `ssh_config` Standarddatei enthalten sind. Sofern Sie diese Datei nicht zuvor bearbeitet haben, verwendet SFTP Port 22. Sie können einen anderen Port angeben (z. B. 2222), indem Sie dem Befehl wie folgt ein **`-P`** Flag hinzufügen.
   >
   > ```
   > sftp -P 2222 -i transfer-key sftp_user@service_endpoint
   > ```
   >
   > Wenn Sie immer Port 2222 oder Port 22000 verwenden möchten, können Sie alternativ Ihren Standardport in Ihrer Datei aktualisieren. `ssh_config`

   Eine `sftp`-Eingabeaufforderung sollte angezeigt werden.
3. (Optional) Um das Home-Verzeichnis des Benutzers anzuzeigen, geben Sie an der Eingabeaufforderung den folgenden Befehl ein: `sftp`

```
pwd
```

4. Verwenden Sie den `put` Befehl, um eine Datei von Ihrem Dateisystem auf den Transfer Family Family-Server hochzuladen. Um beispielsweise hochzuladen `hello.txt` (vorausgesetzt, die Datei befindet sich in Ihrem aktuellen Verzeichnis auf Ihrem Dateisystem), führen Sie an der `sftp` Eingabeaufforderung den folgenden Befehl aus:

   ```
   put hello.txt
   ```

   Eine Meldung ähnlich der folgenden wird angezeigt, die darauf hinweist, dass die Dateiübertragung im Gange ist oder abgeschlossen ist.

   ```
   Uploading hello.txt to /my-bucket/home/sftp_user/hello.txt
   ```

   ```
   hello.txt 100% 127 0.1KB/s 00:00
   ```

> **Note**
>
> Nachdem Ihr Server erstellt wurde, kann es einige Minuten dauern, bis der Hostname des Serverendpunkts vom DNS-Dienst in Ihrer Umgebung aufgelöst werden kann.

## Verwenden Sie WinSCP

Unten wird beschrieben, wie Dateien in der Befehlszeile mit WinSCP übertragen werden.

> **Note**
>
> Wenn Sie WinSCP 5.19 verwenden, können Sie mit Ihren AWS Anmeldeinformationen direkt eine Verbindung zu Amazon S3 herstellen und Dateien hochladen/herunterladen. Weitere Informationen finden Sie unter Verbindung zum Amazon S3 S3-Service herstellen.

Um Dateien AWS Transfer Family mit WinSCP zu übertragen

1. Öffnen Sie den WinSCP-Client.
2. Wählen Sie im Anmeldedialogfeld für Dateiprotokoll ein Protokoll aus: SFTP oder FTP.

   Wenn Sie FTP für Verschlüsselung ausgewählt haben, wählen Sie eine der folgenden Optionen:

- Keine Verschlüsselung für FTP
- Explizite TLS/SSL-Verschlüsselung für FTPS

3. Geben Sie in das Feld Host name (Host-Name) den Server-Endpunkt ein. Der Serverendpunkt befindet sich auf der Seite mit den Serverdetails. Weitere Informationen finden Sie unter SFTP-, FTPS- und FTP-Serverdetails anzeigen.

> **Note**
>
> Wenn Ihr Server einen VPC-Endpunkt verwendet, finden Sie weitere Informationen unterFinden Sie Ihren Amazon VPC-Endpunkt.

4. Geben Sie als Portnummer Folgendes ein:

- **22**für SFTP
- **21**für FTP/FTPS

5. Geben Sie unter Benutzername den Namen des Benutzers ein, den Sie für Ihren spezifischen Identitätsanbieter erstellt haben.

> **Note**
>
> Der Benutzername sollte einer der Benutzer sein, die Sie für Ihren Identitätsanbieter erstellt oder konfiguriert haben. AWS Transfer Family bietet die folgenden Identitätsanbieter:
>
> - Arbeiten mit serviceverwalteten Benutzern
> - Verwenden des AWS Directory Service-Identitätsanbieters
> - Mit Anbietern benutzerdefinierter Identitäten arbeiten

6. Wählen Sie „Erweitert“, um das Dialogfeld „Erweiterte Seiteneinstellungen“ zu öffnen. Wählen Sie im Abschnitt SSH die Option Authentifizierung aus.
7. Suchen Sie unter Datei mit privatem Schlüssel nach der privaten SSH-Schlüsseldatei aus Ihrem Dateisystem und wählen Sie sie aus.

> **Note**
>
> Wenn WinSCP anbietet, Ihren privaten SSH-Schlüssel in das PPK-Format zu konvertieren, wählen Sie OK.

8. Wählen Sie OK aus, um zum Dialogfeld Login (Anmelden) zurückzukehren. Wählen Sie dann Save (Speichern) aus.
9. Wählen Sie im Dialogfeld Sitzung als Site speichern die Option OK, um die Verbindungseinrichtung abzuschließen.
10. Wählen Sie im Anmeldedialogfeld die Option Tools und anschließend Einstellungen aus.
11. Wählen Sie im Dialogfeld „Einstellungen“ für „Übertragung“ die Option „Endurance“.

    Wählen Sie für die Option „Wiederaufnahme der Übertragung/Übertragung auf temporären Dateinamen aktivieren“ die Option „Deaktivieren“.

    > **Note**
    >
    > Wenn Sie diese Option aktiviert lassen, erhöht dies die Upload-Kosten und verringert die Upload-Leistung erheblich. Dies kann auch dazu führen, dass das Hochladen großer Dateien fehlschlägt.

12. Wählen Sie für Übertragung die Option Hintergrund und deaktivieren Sie das Kontrollkästchen Mehrere Verbindungen für eine einzige Übertragung verwenden.

    > **Note**
    >
    > Wenn Sie diese Option aktiviert lassen, können große Dateiuploads auf unvorhersehbare Weise fehlschlagen. Beispielsweise können verwaiste mehrteilige Uploads erstellt werden, für die Amazon S3 S3-Gebühren anfallen. Es kann auch zu einer unbemerkten Beschädigung von Daten kommen.

13. Führen Sie Ihre Dateiübertragung durch.

    Sie können drag-and-drop Methoden verwenden, um Dateien zwischen dem Ziel- und dem Quellfenster zu kopieren. Sie können die Symbolleistensymbole verwenden, um die Eigenschaften von Dateien in WinSCP hochzuladen, herunterzuladen, zu löschen, zu bearbeiten oder zu ändern.

> **Note**
>
> Dieser Hinweis gilt nicht, wenn Sie Amazon EFS als Speicher verwenden.
> Befehle, die versuchen, Attribute von Remote-Dateien, einschließlich Zeitstempeln, zu ändern, sind nicht mit Objektspeichersystemen wie Amazon S3 kompatibel. Wenn Sie Amazon S3 als Speicher verwenden, stellen Sie daher sicher, dass Sie die WinSCP-Zeitstempeleinstellungen deaktivieren (oder `SetStatOption` wie unter beschrieben verwendenVermeiden Sie Fehler `setstat`), bevor Sie Dateiübertragungen durchführen. Deaktivieren Sie dazu im Dialogfeld mit den WinSCP-Übertragungseinstellungen die Upload-Option Berechtigungen festlegen und die allgemeine Option Zeitstempel beibehalten.

## Verwenden Sie Cyberduck

Unten wird beschrieben, wie Dateien in der Befehlszeile mit Cyberduck übertragen werden.

Um Dateien AWS Transfer Family mit Cyberduck zu übertragen

1. Öffnen Sie den Cyberduck-Client.
2. Wählen Sie Verbindung öffnen.
3. Wählen Sie im Dialogfeld „Verbindung öffnen“ ein Protokoll aus: SFTP (SSH File Transfer Protocol), FTP-SSL (Explicit AUTH TLS) oder FTP (File Transfer Protocol).
4. Geben Sie unter Server Ihren Serverendpunkt ein. Der Serverendpunkt befindet sich auf der Seite mit den Serverdetails. Weitere Informationen finden Sie unter SFTP-, FTPS- und FTP-Serverdetails anzeigen.

   > **Note**
   >
   > Wenn Ihr Server einen VPC-Endpunkt verwendet, finden Sie weitere Informationen unterFinden Sie Ihren Amazon VPC-Endpunkt.

5. Geben Sie als Portnummer Folgendes ein:

   - **22**für SFTP
   - **21**für FTP/FTPS

6. Geben Sie in das Feld Username (Benutzername) den Namen des Benutzers ein, den Sie in Verwalten von Benutzern für Serverendpunkte erstellt haben.

7. Wenn SFTP ausgewählt ist, wählen Sie für SSH Private Key den privaten SSH-Schlüssel oder geben Sie ihn ein.
8. Wählen Sie Connect aus.
9. Führen Sie Ihre Dateiübertragung durch.

   Führen Sie nun – abhängig von der Position der Dateien – einen der folgenden Schritte durch:

   - Wählen Sie in Ihrem lokalen Verzeichnis (der Quelle) die Dateien aus, die Sie übertragen möchten, und ziehen Sie sie per Drag & Drop in das Amazon S3 S3-Verzeichnis (das Ziel).
   - Wählen Sie im Amazon S3 S3-Verzeichnis (der Quelle) die Dateien aus, die Sie übertragen möchten, und ziehen Sie sie per Drag & Drop in Ihr lokales Verzeichnis (das Ziel).

## Verwenden FileZilla

Verwenden Sie die folgenden Anweisungen, um Dateien mit zu übertragen FileZilla.

So richten Sie FileZilla eine Dateiübertragung ein

1. Öffnen Sie den FileZilla Client.
2. Wählen Sie „Datei“ und anschließend „Site-Manager“.
3. Wählen Sie im Dialogfeld „Site-Manager“ die Option „Neue Site“ aus.
4. Wählen Sie auf der Registerkarte Allgemein für Protokoll ein Protokoll aus: SFTP oder FTP.

   Wenn Sie FTP für Verschlüsselung ausgewählt haben, wählen Sie eine der folgenden Optionen:

   - Verwenden Sie nur einfaches FTP (unsicher) — für FTP
   - Verwenden Sie explizites FTP über TLS, falls verfügbar — für FTPS
5. Geben Sie als Hostname das Protokoll ein, das Sie verwenden, gefolgt von Ihrem Serverendpunkt. Der Serverendpunkt befindet sich auf der Seite mit den Serverdetails. Weitere Informationen finden Sie unter SFTP-, FTPS- und FTP-Serverdetails anzeigen.

   > **Note**
   >
   > Wenn Ihr Server einen VPC-Endpunkt verwendet, finden Sie weitere Informationen unterFinden Sie Ihren Amazon VPC-Endpunkt.

- Wenn Sie SFTP verwenden, geben Sie Folgendes ein: `sftp://`*`hostname`*
- Wenn Sie FTPS verwenden, geben Sie Folgendes ein: `ftps://`*`hostname`*

Stellen Sie sicher, dass Sie den *`Hostnamen`* durch Ihren tatsächlichen Serverendpunkt ersetzen.

6. Geben Sie als Portnummer Folgendes ein:

   - **22**für SFTP
   - **21**für FTP/FTPS

7. Wenn SFTP ausgewählt ist, wählen Sie als Anmeldetyp die Option Schlüsseldatei aus.

   Wählen Sie unter Schlüsseldatei den privaten SSH-Schlüssel aus, oder geben Sie ihn ein.

8. Geben Sie unter Benutzer den Namen des Benutzers ein, den Sie erstellt haben. Verwalten von Benutzern für Serverendpunkte
9. Wählen Sie Connect aus.
10. Führen Sie Ihre Dateiübertragung durch.

> **Note**
>
> Wenn Sie eine laufende Dateiübertragung unterbrechen, wird AWS Transfer Family möglicherweise ein Teilobjekt in Ihren Amazon S3 S3-Bucket geschrieben. Wenn Sie einen Upload unterbrechen, überprüfen Sie, ob die Dateigröße im Amazon S3 S3-Bucket der Dateigröße des Quellobjekts entspricht, bevor Sie fortfahren.

## Verwenden Sie einen Perl-Client

Wenn Sie den NET::SFTP::Foreign Perl-Client verwenden, müssen Sie den `queue_size` Wert auf setzen. 1 Beispielsweise:

```
my $sftp = Net::SFTP::Foreign->new('user@s-12345.server.transfer.us-east-2.amazonaws.com', queue_size => 1);
```

> **Note**
>
> Diese Problemumgehung ist für Versionen `Net::SFTP::Foreign` vor 1.92.02 erforderlich.

# Verarbeitung nach dem Upload

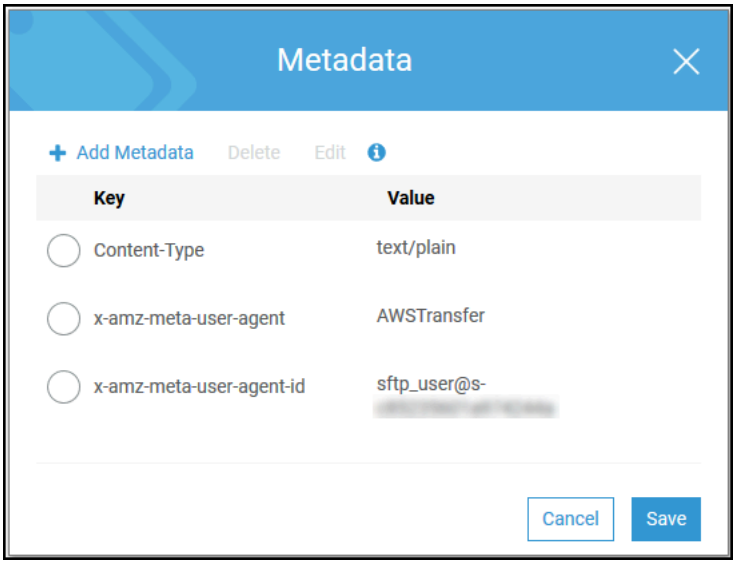
Sie können Verarbeitungsinformationen nach dem Upload anzeigen, einschließlich Amazon S3 S3-Objektmetadaten und Ereignisbenachrichtigungen.

Themen


- Amazon S3 S3-Objektmetadaten
- Amazon-S3-Ereignis-Benachrichtigungen


## Amazon S3 S3-Objektmetadaten

Als Teil der Metadaten Ihres Objekts sehen Sie einen Schlüssel namens `x-amz-meta-user-agent` Wessen Wert ist `AWSTransfer` und `x-amz-meta-user-agent-id` Wessen Wert ist`username@server-id`. Der `username` ist der Transfer Family Family-Benutzer, der die Datei hochgeladen hat, und `server-id` ist der Server, der für den Upload verwendet wurde. Auf diese Informationen kann mithilfe der HeadObjectOperation für das S3-Objekt in Ihrer Lambda-Funktion zugegriffen werden.




## Amazon-S3-Ereignis-Benachrichtigungen

Wenn ein Objekt mithilfe von Transfer Family in Ihren S3-Bucket hochgeladen `RoleSessionName` wird, ist es im Feld Requester in der S3-Ereignisbenachrichtigungsstruktur als `[AWS:Role Unique Identifier]/username.sessionid@server-id` enthalten. Im Folgenden finden Sie beispielsweise den Inhalt eines Beispielfeldes „Requester“ aus einem S3-Zugriffsprotokoll für eine Datei, die in den S3-Bucket kopiert wurde.

```
arn:aws:sts::AWS-Account-ID:assumed-role/IamRoleName/
username.sessionid@server-id
```

Im obigen Feld Requester wird die aufgerufene IAM-Rolle angezeigt. `IamRoleName` Weitere Informationen zur Konfiguration von S3-Ereignisbenachrichtigungen finden Sie unter Konfiguration von Amazon S3 S3-Ereignisbenachrichtigungen im Amazon Simple Storage Service Developer Guide. Weitere Informationen zu eindeutigen AWS Identity and Access Management (IAM-) Rollenbezeichnern finden Sie unter Eindeutige Identifikatoren im AWS Identity and Access Management Benutzerhandbuch.

## Verwalten von Benutzern für Serverendpunkte

In den folgenden Abschnitten finden Sie Informationen zum Hinzufügen von Benutzern mithilfe von AWS Directory Service for Microsoft Active Directory oder eines benutzerdefinierten AWS Transfer FamilyIdentitätsanbieters.

Wenn Sie einen serviceverwalteten Identitätstyp verwenden, fügen Sie Ihrem Server mit aktiviertem Dateiübertragungsprotokoll Benutzer hinzu. Wenn Sie dies tun, muss jeder Benutzername auf Ihrem Server eindeutig sein.

Sie speichern außerdem in den Eigenschaften der Benutzer den öffentlichen Secure Shell (SSH)-Schlüssel des jeweiligen Benutzers. Dies ist für die schlüsselbasierte Authentifizierung erforderlich, die dieses Verfahren verwendet. Der private Schlüssel wird lokal auf dem Computer Ihres Benutzers gespeichert. Wenn Ihr Benutzer mithilfe eines Clients eine Authentifizierungsanforderung an Ihren Server sendet, bestätigt Ihr Server zunächst, dass der Benutzer Zugriff auf den zugehörigen privaten SSH-Schlüssel hat. Der Server authentifiziert den Benutzer dann erfolgreich.

Darüber hinaus geben Sie das Stammverzeichnis oder das Zielverzeichnis eines Benutzers an und weisen ihm eine AWS Identity and Access Management (IAM)-Rolle zu. Optional können Sie eine Sitzungsrichtlinie bereitstellen, um den Benutzerzugriff nur auf das Stammverzeichnis Ihres Amazon S3-Buckets zu beschränken.

> **Important**
>
> AWS Transfer Family blockiert Benutzernamen, die 1 oder 2 Zeichen lang sind, von der Authentifizierung auf SFTP-Servern. Darüber hinaus blockieren wir auch den `root`Benutzernamen.

Der Grund dafür ist das große Volumen an böswilligen Anmeldeversuchen durch Passwortgeräte.

### Amazon EFS im Vergleich zu Amazon S3

Merkmale jeder Speicheroption:

- So beschränken Sie den Zugriff: Amazon S3 unterstützt Sitzungsrichtlinien; Amazon EFS unterstützt POSIX-Benutzer-, Gruppen- und sekundäre Gruppen-IDs
- Beide unterstützen öffentliche/private Schlüssel
- Beide unterstützen Basisverzeichnisse
- Beide unterstützen logische Verzeichnisse

> **Note**
>
> Für Amazon S3 erfolgt der Großteil der Unterstützung für logische Verzeichnisse über API/CLI. Sie können das Kontrollkästchen Eingeschränkt in der Konsole verwenden, um einen Benutzer für sein Home-Verzeichnis zu sperren, aber Sie können keine virtuelle Verzeichnisstruktur angeben.

### Logische Verzeichnisse

Wenn Sie logische Verzeichniswerte für Ihren Benutzer angeben, hängt der von Ihnen verwendete Parameter vom Benutzertyp ab.

- Geben Sie für serviceverwaltete Benutzer logische Verzeichniswerte in an`HomeDirectoryMappings`.
- Geben Sie für benutzerdefinierte Identitätsanbieter-Benutzer logische Verzeichniswerte in an`HomeDirectoryDetails`.

### Themen


- Arbeiten mit serviceverwalteten Benutzern
- Verwenden des AWS Directory Service-Identitätsanbieters
- Mit Anbietern benutzerdefinierter Identitäten arbeiten

# Arbeiten mit serviceverwalteten Benutzern

Sie können Ihrem Server je nach Domäneneinstellung des Servers entweder serviceverwaltete Amazon S3- oder Amazon-EFS-Benutzer hinzufügen. Weitere Informationen finden Sie unter Konfiguration eines SFTP-, FTPS- oder FTP-Serverendpunkts.

Informationen zum programmgesteuerten Hinzufügen eines serviceverwalteten Benutzers finden Sie im Beispiel für die CreateUser-API.

> **Note**
>
> Für serviceverwaltete Benutzer gibt es ein Limit von 2 000 logischen Verzeichniseinträgen. Informationen zur Verwendung logischer Verzeichnisse finden Sie unter Verwendung logischer Verzeichnisse zur Vereinfachung Ihrer Transfer Family Family-Verzeichnisstrukturen.

Themen


- Hinzufügen von serviceverwalteten Amazon S3-Benutzern
- Hinzufügen von serviceverwalteten Amazon-EFS-Benutzern
- Verwalten von serviceverwalteten Benutzern


## Hinzufügen von serviceverwalteten Amazon S3-Benutzern

> **Note**
>
> Wenn Sie einen kontoübergreifenden Amazon S3-Bucket konfigurieren möchten, führen Sie die in diesem Knowledge-Center-Artikel genannten Schritte aus: Wie konfiguriere ich meinen AWS Transfer Family Server für die Verwendung eines Amazon-Simple-Storage-Service-Buckets, der sich in einem anderen AWS Konto befindet?.

So fügen Sie Ihrem Server einen serviceverwalteten Amazon S3-Benutzer hinzu

1. Öffnen Sie die -AWS Transfer FamilyKonsole unter https://console.aws.amazon.com/transfer/ und wählen Sie dann im Navigationsbereich Server aus.

2. Aktivieren Sie auf der Seite Server das Kontrollkästchen des Servers, dem Sie einen Benutzer hinzufügen möchten.
3. Wählen Sie Benutzer hinzufügen.
4. Geben Sie im Abschnitt Benutzerkonfiguration für Benutzername den Benutzernamen ein. Dieser Benutzername muss mindestens 3 und maximal 100 Zeichen lang sein. Sie können die folgenden Zeichen im Benutzernamen verwenden: a–z, A–Z, 0–9, Unterstrich „_“, Bindestrich „-“, Punkt „.“ und At-Zeichen „@“. Der Benutzername darf nicht mit einem Bindestrich '-', Punkt '.' oder dem At-Zeichen "@" beginnen.
5. Wählen Sie für Zugriff die zuvor erstellte IAM-Rolle aus, die Zugriff auf Ihren Amazon S3-Bucket bietet.

   Sie haben diese IAM-Rolle in der Prozedur Erstellen Sie eine IAM-Rolle und -Richtlinie erstellt. Diese IAM-Rolle enthält eine IAM-Richtlinie, die Zugriff auf Ihren Amazon S3-Bucket gewährt. Sie enthält zudem eine Vertrauensbeziehung zum AWS Transfer Family-Service, die in einer anderen IAM-Richtlinie definiert ist. Wenn Sie eine differenzierte Zugriffskontrolle für Ihre Benutzer benötigen, lesen Sie den Blogbeitrag Verbessern der Datenzugriffskontrolle mit AWS Transfer Family und Amazon S3.
6. (Optional) Wählen Sie für Richtlinie eine der folgenden Optionen aus:

   - Keine
   - Bestehende Richtlinie
   - Wählen Sie eine Richtlinie aus IAM aus: Mit können Sie eine vorhandene Sitzungsrichtlinie auswählen. Wählen Sie Anzeigen, um ein JSON-Objekt anzuzeigen, das die Details der Richtlinie enthält.
   - Richtlinie automatisch generieren basierend auf dem Basisordner : generiert eine Sitzungsrichtlinie für Sie. Wählen Sie Anzeigen, um ein JSON-Objekt anzuzeigen, das die Details der Richtlinie enthält.

   > **Note**
   >
   > Wenn Sie Richtlinie basierend auf dem Basisordner automatisch generieren auswählen, wählen Sie für diesen Benutzer nicht Eingeschränkt aus.

Weitere Informationen zu Sitzungsrichtlinien finden Sie unter Erstellen Sie eine IAM-Rolle und -Richtlinie. Weitere Informationen zum Erstellen einer Sitzungsrichtlinie finden Sie unter Sitzungsrichtlinie für einen Amazon S3 S3-Bucket erstellen.

7. Wählen Sie für Home directory den Amazon S3-Bucket aus, in dem die Daten gespeichert werden sollen, die mit übertragen werden sollenAWS Transfer Family. Geben Sie den Pfad zu dem home Verzeichnis ein, in dem Ihr Benutzer landet, wenn er sich mit seinem Client anmeldet.

   Wenn Sie diesen Parameter leer lassen, wird das `root` Verzeichnis Ihres Amazon S3-Buckets verwendet. Stellen Sie in diesem Fall sicher, dass die IAM-Rolle Zugriff auf dieses `root`-Verzeichnis gewährt.

   > **Note**
   >
   > Wir empfehlen Ihnen, einen Verzeichnispfad zu wählen, der den Benutzernamen des Benutzers enthält, sodass Sie eine Sitzungsrichtlinie effektiv verwenden können. Die Sitzungsrichtlinie beschränkt den Benutzerzugriff im Amazon S3-Bucket auf das home Verzeichnis dieses Benutzers.

8. (Optional) Aktivieren Sie für Eingeschränkt das Kontrollkästchen, damit Ihre Benutzer nicht auf etwas außerhalb dieses Ordners zugreifen können und den Amazon S3-Bucket oder -Ordnernamen nicht sehen können.

   > **Note**
   >
   > Es sollte ausreichend sein, dem Benutzer ein Stammverzeichnis zuzuweisen und den Benutzer auf dieses Stammverzeichnis zu beschränken, um den Zugriff des Benutzers auf den angegebenen Ordner zu beschränken. Wenn Sie weitere Kontrollen anwenden müssen, verwenden Sie eine Sitzungsrichtlinie.
   > Wenn Sie Eingeschränkt für diesen Benutzer auswählen, können Sie Richtlinie basierend auf dem Basisordner nicht automatisch generieren auswählen, da der Basisordner kein definierter Wert für Eingeschränkte Benutzer ist.

9. Geben Sie für den öffentlichen SSH-Schlüssel den öffentlichen SSH-Schlüsselteil des SSH-Schlüsselpaars ein.

   Der Schlüssel wird vom Service validiert, bevor Sie den neuen Benutzer hinzufügen können.

> **Note**
>
> Anleitungen zum Generieren eines SSH-Schlüsselpaars siehe Generieren Sie SSH-Schlüssel für vom Service verwaltete Benutzer.

10. (Optional) Geben Sie für Schlüssel und Wert ein oder mehrere Tags als Schlüssel-Wert-Paare ein und wählen Sie Tag hinzufügen aus.
11. Wählen Sie Add (Hinzufügen), um den neuen Benutzer dem ausgewählten Server hinzuzufügen.

    Der neue Benutzer wird im Abschnitt Benutzer der Seite Serverdetails angezeigt.

Nächste Schritte – Fahren Sie für den nächsten Schritt mit fortÜbertragung von Dateien über einen Serverendpunkt mit einem Client.

## Hinzufügen von serviceverwalteten Amazon-EFS-Benutzern

Amazon EFS verwendet das POSIX-Dateiberechtigungsmodell (Portable Operating System Interface), um den Dateibesitz darzustellen.

- Weitere Informationen zur Eigentümerschaft von Amazon-EFS-Dateien finden Sie unter Amazon-EFS-Dateieigentümerschaft.
- Weitere Informationen zum Einrichten von Verzeichnissen für Ihre EFS-Benutzer finden Sie unter Amazon EFS-Benutzer für Transfer Family einrichten.

So fügen Sie Ihrem Server einen serviceverwalteten Amazon-EFS-Benutzer hinzu

1. Öffnen Sie die -AWS Transfer FamilyKonsole unter https://console.aws.amazon.com/transfer/ und wählen Sie dann im Navigationsbereich Server aus.
2. Wählen Sie auf der Seite Server den Amazon-EFS-Server aus, dem Sie einen Benutzer hinzufügen möchten.
3. Wählen Sie Benutzer hinzufügen, um die Seite Benutzer hinzufügen anzuzeigen.
4. Verwenden Sie im Abschnitt Benutzerkonfiguration die folgenden Einstellungen.

    a. Der Benutzername muss mindestens 3 und maximal 100 Zeichen lang sein. Sie können die folgenden Zeichen im Benutzernamen verwenden: a–z, A–Z, 0–9, Unterstrich „_“, Bindestrich

„-“, Punkt „.“ und At-Zeichen „@“. Der Benutzername darf nicht mit einem Bindestrich '-', Punkt '.' oder dem At-Zeichen "@" beginnen.

b. Beachten Sie für Benutzer-ID und Gruppen-ID Folgendes:

- Für den ersten Benutzer, den Sie erstellen, empfehlen wir Ihnen, sowohl für die Gruppen-ID als auch **`0`** für die Benutzer-ID einen Wert von einzugeben. Dadurch erhalten die Benutzeradministratorrechte für Amazon EFS .
- Geben Sie für weitere Benutzer die POSIX-Benutzer-ID und die Gruppen-ID des Benutzers ein. Diese IDs werden für alle vom Benutzer ausgeführten Amazon Elastic File System-Operationen verwendet.
- Verwenden Sie für Benutzer-ID und Gruppen-ID keine führenden Nullen. Beispielsweise **`12345`** ist akzeptabel, **`012345`** ist nicht.

c. (Optional) Geben Sie für sekundäre Gruppen-IDs eine oder mehrere zusätzliche POSIX-Gruppen-IDs für jeden Benutzer ein, getrennt durch Kommas.

d. Wählen Sie für Zugriff die IAM-Rolle aus, die:

- Gewährt dem Benutzer nur Zugriff auf die Amazon-EFS-Ressourcen (Dateisysteme), auf die er zugreifen soll.
- Definiert, welche Dateisystemoperationen der Benutzer ausführen kann und welche nicht.

Wir empfehlen Ihnen, die IAM-Rolle für die Amazon-EFS-Dateisystemauswahl mit Mount-Zugriff und Lese-/Schreibberechtigungen zu verwenden. Beispielsweise gewährt die Kombination der folgenden beiden AWS verwalteten Richtlinien, obwohl sie ziemlich offen sind, Ihrem Benutzer die erforderlichen Berechtigungen:

- AmazonElasticFileSystemClientFullAccess
- AWSTransferConsoleFullAccess

Weitere Informationen finden Sie im Blogbeitrag AWS Transfer Family Support für Amazon Elastic File System .

e. Gehen Sie für Home directory wie folgt vor:

- Wählen Sie das Amazon-EFS-Dateisystem aus, das Sie zum Speichern der zu übertragenden Daten mit verwenden möchtenAWS Transfer Family.

- Entscheiden Sie, ob das Stammverzeichnis auf Eingeschränkt gesetzt werden soll. Das Festlegen des Stammverzeichnisses auf Eingeschränkt hat die folgenden Auswirkungen:
  - Amazon-EFS-Benutzer können nicht auf Dateien oder Verzeichnisse außerhalb dieses Ordners zugreifen.
  - Amazon-EFS-Benutzer können den Amazon-EFS-Dateisystemnamen (fs-xxxxxxx ) nicht sehen.

  > **Note**
  >
  > Wenn Sie die Option Eingeschränkt auswählen, werden Symlinks für Amazon-EFS-Benutzer nicht aufgelöst.

- (Optional) Geben Sie den Pfad zum Stammverzeichnis ein, in dem sich Benutzer befinden sollen, wenn sie sich mit ihrem Client anmelden.

  Wenn Sie kein Stammverzeichnis angeben, wird das Stammverzeichnis Ihres Amazon-EFS-Dateisystems verwendet. Stellen Sie in diesem Fall sicher, dass Ihre IAM-Rolle Zugriff auf dieses Stammverzeichnis gewährt.

5. Geben Sie für den öffentlichen SSH-Schlüssel den öffentlichen SSH-Schlüsselteil des SSH-Schlüsselpaars ein.

   Der Schlüssel wird vom Service validiert, bevor Sie den neuen Benutzer hinzufügen können.

   > **Note**
   >
   > Anleitungen zum Generieren eines SSH-Schlüsselpaars siehe Generieren Sie SSH-Schlüssel für vom Service verwaltete Benutzer.

6. (Optional) Geben Sie alle Tags für den Benutzer ein. Geben Sie für Schlüssel und Wert ein oder mehrere Tags als Schlüssel-Wert-Paare ein und wählen Sie Tag hinzufügen aus.
7. Wählen Sie Add (Hinzufügen), um den neuen Benutzer dem ausgewählten Server hinzuzufügen.

   Der neue Benutzer wird im Abschnitt Benutzer der Seite Serverdetails angezeigt.

Probleme, die beim ersten SFTP zu Ihrem Transfer Family-Server auftreten können:

- Wenn Sie den `sftp` Befehl ausführen und die Eingabeaufforderung nicht angezeigt wird, wird möglicherweise die folgende Meldung angezeigt:

  ```
  Couldn't canonicalize: Permission denied

  Need cwd
  ```

  In diesem Fall müssen Sie die Richtlinienberechtigungen für die Rolle Ihres Benutzers erhöhen. Sie können eine von AWS verwaltete Richtlinie hinzufügen, z. B. `AmazonElasticFileSystemClientFullAccess`.
- Wenn Sie pwd an der `sftp` Eingabeaufforderung eingeben, um das Stammverzeichnis des Benutzers anzuzeigen, wird möglicherweise die folgende Meldung angezeigt, wobei *`USER-HOME-DIRECTORY`* das Stammverzeichnis für den SFTP-Benutzer ist:

  ```
  remote readdir("/USER-HOME-DIRECTORY"): No such file or directory
  ```

  In diesem Fall sollten Sie zum übergeordneten Verzeichnis (`cd ..`) navigieren und das Stammverzeichnis (`mkdir username`) des Benutzers erstellen können.

Nächste Schritte – Fahren Sie für den nächsten Schritt mit fortÜbertragung von Dateien über einen Serverendpunkt mit einem Client.

## Verwalten von serviceverwalteten Benutzern

In diesem Abschnitt finden Sie Informationen zum Anzeigen einer Liste von Benutzern, zum Bearbeiten von Benutzerdetails und zum Hinzufügen eines öffentlichen SSH-Schlüssels.

- Anzeigen einer Liste von Benutzern
- Anzeigen oder Bearbeiten von Benutzerdetails
- Löschen eines Benutzers
- Öffentlichen SSH-Schlüssel hinzufügen
- Öffentlichen SSH-Schlüssel löschen

So finden Sie eine Liste Ihrer Benutzer

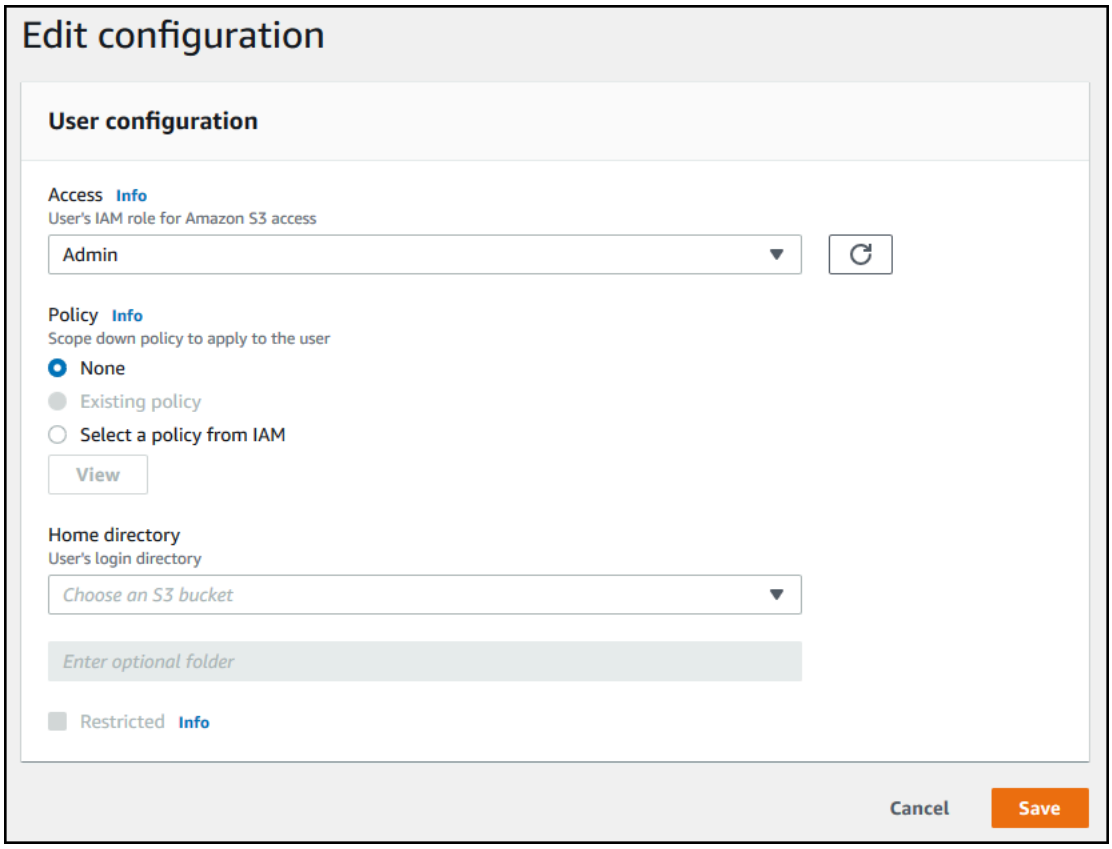
1. Öffnen Sie die -AWS Transfer FamilyKonsole unter https://console.aws.amazon.com/transfer/.
2. Wählen Sie im Navigationsbereich Server aus, um die Seite Server anzuzeigen.

3. Wählen Sie die Kennung in der Spalte Server-ID aus, um die Seite Serverdetails anzuzeigen.
4. Zeigen Sie unter Benutzer eine Liste von Benutzern an.

So zeigen Sie Benutzerdetails an oder bearbeiten sie

1. Öffnen Sie die -AWS Transfer FamilyKonsole unter https://console.aws.amazon.com/transfer/.
2. Wählen Sie im Navigationsbereich Server aus, um die Seite Server anzuzeigen.
3. Wählen Sie die Kennung in der Spalte Server-ID aus, um die Seite Serverdetails anzuzeigen.
4. Wählen Sie unter Benutzer einen Benutzernamen aus, um die Seite Benutzerdetails anzuzeigen.

   Sie können die Eigenschaften des Benutzers auf dieser Seite ändern, indem Sie Bearbeiten auswählen.
5. Wählen Sie auf der Seite Benutzerdetails neben Benutzerkonfiguration die Option Bearbeiten aus.




6. Wählen Sie auf der Seite Konfiguration bearbeiten für Zugriff die zuvor erstellte IAM-Rolle aus, die Zugriff auf Ihren Amazon S3-Bucket bietet.

Sie haben diese IAM-Rolle in der Prozedur Erstellen Sie eine IAM-Rolle und -Richtlinie erstellt. Diese IAM-Rolle enthält eine IAM-Richtlinie, die Zugriff auf Ihren Amazon S3-Bucket ermöglicht. Sie enthält zudem eine Vertrauensbeziehung zum AWS Transfer Family-Service, die in einer anderen IAM-Richtlinie definiert ist.

7. (Optional) Wählen Sie für Richtlinie eine der folgenden Optionen aus:

   - Keine
   - Bestehende Richtlinie
   - Wählen Sie eine Richtlinie aus IAM aus, um eine vorhandene Richtlinie auszuwählen. Wählen Sie Anzeigen, um ein JSON-Objekt anzuzeigen, das die Details der Richtlinie enthält.

   Weitere Informationen zu Sitzungsrichtlinien finden Sie unter Erstellen Sie eine IAM-Rolle und -Richtlinie. Weitere Informationen zum Erstellen einer Sitzungsrichtlinie finden Sie unter Sitzungsrichtlinie für einen Amazon S3 S3-Bucket erstellen.

8. Wählen Sie für Home directory den Amazon S3-Bucket aus, in dem die Daten gespeichert werden sollen, die mit übertragen werden sollenAWS Transfer Family. Geben Sie den Pfad zu dem home Verzeichnis ein, in dem Ihr Benutzer landet, wenn er sich mit seinem Client anmeldet.

   Wenn Sie diesen Parameter leer lassen, wird das `root` Verzeichnis Ihres Amazon S3-Buckets verwendet. Stellen Sie in diesem Fall sicher, dass die IAM-Rolle Zugriff auf dieses `root`-Verzeichnis gewährt.

   > **Note**
   >
   > Wir empfehlen Ihnen, einen Verzeichnispfad zu wählen, der den Benutzernamen des Benutzers enthält, sodass Sie eine Sitzungsrichtlinie effektiv verwenden können. Die Sitzungsrichtlinie beschränkt den Benutzerzugriff im Amazon S3-Bucket auf das home Verzeichnis dieses Benutzers.

9. (Optional) Aktivieren Sie für Eingeschränkt das Kontrollkästchen, damit Ihre Benutzer nicht auf etwas außerhalb dieses Ordners zugreifen können und den Amazon S3-Bucket oder Ordnernamen nicht sehen können.

> **Note**
>
> Wenn Sie dem Benutzer ein Stammverzeichnis zuweisen und den Benutzer auf dieses Stammverzeichnis beschränken, sollte dies ausreichend sein, um den Zugriff des Benutzers auf den angegebenen Ordner zu beschränken. Verwenden Sie eine Sitzungsrichtlinie, wenn Sie weitere Kontrollen anwenden müssen.

10. Wählen Sie Speichern, um Ihre Änderungen zu speichern.

Benutzer löschen

1. Öffnen Sie die -AWS Transfer FamilyKonsole unter https://console.aws.amazon.com/transfer/.
2. Wählen Sie im Navigationsbereich Server aus, um die Seite Server anzuzeigen.
3. Wählen Sie die Kennung in der Spalte Server-ID aus, um die Seite Serverdetails anzuzeigen.
4. Wählen Sie unter Benutzer einen Benutzernamen aus, um die Seite Benutzerdetails anzuzeigen.
5. Wählen Sie auf der Seite Benutzerdetails rechts neben dem Benutzernamen Löschen aus.
6. Geben Sie im daraufhin angezeigten Bestätigungsdialogfeld das Wort ein und wählen Sie dann Löschen**`delete`**, um zu bestätigen, dass Sie den Benutzer löschen möchten.

Der Benutzer wird aus der Benutzerliste gelöscht.

So fügen Sie einen öffentlichen SSH-Schlüssel für einen Benutzer hinzu

1. Öffnen Sie die -AWS Transfer FamilyKonsole unter https://console.aws.amazon.com/transfer/.
2. Klicken Sie im Navigationsbereich auf Servers (Server).
3. Wählen Sie die Kennung in der Spalte Server-ID aus, um die Seite Serverdetails anzuzeigen.
4. Wählen Sie unter Benutzer einen Benutzernamen aus, um die Seite Benutzerdetails anzuzeigen.
5. Wählen Sie Add SSH public key (Öffentlichen SSH-Schlüssel hinzufügen), um einem Benutzer einen neuen öffentlichen SSH-Schlüssel hinzuzufügen.

> **Note**
>
> SSH-Schlüssel werden nur von Servern verwendet, die für Secure Shell (SSH) File Transfer Protocol (SFTP) aktiviert sind. Informationen zum Generieren eines SSH-

Schlüsselpaars finden Sie unter Generieren Sie SSH-Schlüssel für vom Service verwaltete Benutzer.

6. Geben Sie für SSH public key (Öffentlicher SSH-Schlüssel) den öffentlichen SSH-Schlüssel des SSH-Schlüsselpaars ein.

   Der Schlüssel wird vom Service validiert, bevor Sie den neuen Benutzer hinzufügen können. Das Format des SSH-Schlüssels lautet `ssh-rsa string`. Informationen zum Generieren eines SSH-Schlüsselpaars finden Sie unter Generieren Sie SSH-Schlüssel für vom Service verwaltete Benutzer.
7. Wählen Sie Schlüssel hinzufügen.

So löschen Sie einen öffentlichen SSH-Schlüssel für einen Benutzer

1. Öffnen Sie die -AWS Transfer FamilyKonsole unter https://console.aws.amazon.com/transfer/.
2. Klicken Sie im Navigationsbereich auf Servers (Server).
3. Wählen Sie die Kennung in der Spalte Server-ID aus, um die Seite Serverdetails anzuzeigen.
4. Wählen Sie unter Benutzer einen Benutzernamen aus, um die Seite Benutzerdetails anzuzeigen.
5. Um einen öffentlichen Schlüssel zu löschen, aktivieren Sie das Kontrollkästchen SSH-Schlüssel und wählen Sie Löschen aus.

## Verwenden des AWS Directory Service-Identitätsanbieters

In diesem Thema wird beschrieben, wie Sie den AWS Directory-Service-Identitätsanbieter für verwenden AWS Transfer Family.

Themen

- Verwenden von AWS Directory Service for Microsoft Active Directory
- Verwenden von AWS Directory Service für Azure Active Directory Domain Services

### Verwenden von AWS Directory Service for Microsoft Active Directory

Sie können verwenden AWS Transfer Family , um Ihre Dateiübertragungs-Endbenutzer mit zu authentifizieren AWS Directory Service for Microsoft Active Directory. Es ermöglicht eine nahtlose Migration von Workflows zur Dateiübertragung, die auf der Active-Directory-Authentifizierung

basieren, ohne die Anmeldeinformationen der Endbenutzer zu ändern oder einen benutzerdefinierten Genehmiger zu benötigen.

Mit können AWS Managed Microsoft AD Sie AWS Directory Service Benutzern und Gruppen über SFTP, FTPS und FTP sicheren Zugriff auf Daten gewähren, die in Amazon Simple Storage Service (Amazon S3) oder Amazon Elastic File System (Amazon EFS) gespeichert sind. Wenn Sie Active Directory verwenden, um die Anmeldeinformationen Ihrer Benutzer zu speichern, haben Sie jetzt eine einfachere Möglichkeit, Dateiübertragungen für diese Benutzer zu aktivieren.

Sie können den Zugriff auf Active-Directory-Gruppen in AWS Managed Microsoft AD in Ihrer On-Premises-Umgebung oder in der AWS Cloud mithilfe von Active-Directory-Konnektoren bereitstellen. Sie können Benutzern, die bereits in Ihrer Microsoft Windows-Umgebung konfiguriert sind, entweder in der AWS Cloud oder in ihrem On-Premises-Netzwerk Zugriff auf einen - AWS Transfer Family Server gewähren, den AWS Managed Microsoft AD für die Identität verwendet.

> **Note**
>
> - AWS Transfer Family unterstützt Simple AD nicht.
> - Transfer Family unterstützt keine regionsübergreifenden Active-Directory-Konfigurationen: Wir unterstützen nur Active-Directory-Integrationen, die sich in derselben Region wie die des Transfer-Family-Servers befinden.
> - Transfer Family unterstützt nicht die Verwendung von AWS Managed Microsoft AD oder AD Connector, um die Multi-Faktor-Authentifizierung (MFA) für Ihre vorhandene RADIUS-basierte MFA-Infrastruktur zu aktivieren.
> - AWS Transfer Family unterstützt keine replizierten Regionen von Managed Active Directory.

Um zu verwenden AWS Managed Microsoft AD, müssen Sie die folgenden Schritte ausführen:

1. Erstellen Sie ein oder mehrere AWS Managed Microsoft AD Verzeichnisse mit der AWS Directory Service Konsole.
2. Verwenden Sie die Transfer Family-Konsole, um einen Server zu erstellen, der AWS Managed Microsoft AD als Identitätsanbieter verwendet.
3. Fügen Sie Zugriff von einer oder mehreren Ihrer AWS Directory Service Gruppen hinzu.
4. Obwohl dies nicht erforderlich ist, empfehlen wir Ihnen, den Benutzerzugriff zu testen und zu überprüfen.

Themen


- Bevor Sie mit der Verwendung von beginnen AWS Directory Service for Microsoft Active Directory
- Arbeiten mit Active Directory-Bereichen
- Auswählen von AWS Managed Microsoft AD als Identitätsanbieter
- Gewähren des Zugriffs auf Gruppen
- Testen von Benutzern
- Löschen des Serverzugriffs für eine Gruppe
- Herstellen einer Verbindung mit dem Server über SSH (Secure Shell)
- Herstellen einer Verbindung AWS Transfer Family zu einem selbstverwalteten Active Directory mithilfe von Gesamtstruktur und Vertrauensstellungen


## Bevor Sie mit der Verwendung von beginnen AWS Directory Service for Microsoft Active Directory

### Geben Sie eine eindeutige Kennung für Ihre AD-Gruppen an

Bevor Sie verwenden können AWS Managed Microsoft AD, müssen Sie für jede Gruppe in Ihrem Microsoft-AD-Verzeichnis eine eindeutige Kennung angeben. Dazu können Sie die Sicherheitskennung (SID) für jede Gruppe verwenden. Die Benutzer der Gruppe, die Sie zuordnen, haben über die aktivierten Protokolle mithilfe von AWS Transfer Family Zugriff auf Ihre Amazon S3- oder Amazon-EFS-Ressourcen.

Verwenden Sie den folgenden Windows- PowerShell Befehl, um die SID für eine Gruppe abzurufen, und ersetzen Sie durch *YourGroupName* den Namen der Gruppe.

```
Get-ADGroup -Filter {samAccountName -like "YourGroupName*"} -Properties * | Select
 SamAccountName,ObjectSid
```

> **Note**
>
> Wenn Sie AWS Directory Service als Identitätsanbieter verwenden und wenn `userPrincipalName` und unterschiedliche Werte `SamAccountName` haben, AWS Transfer Family akzeptiert den Wert in `SamAccountName`. Transfer Family akzeptiert nicht den in angegebenen Wert`userPrincipalName`.

Hinzufügen von AWS Directory Service Berechtigungen zu Ihrer Rolle

Sie benötigen auch AWS Directory Service API-Berechtigungen, um AWS Directory Service als Identitätsanbieter verwenden zu können. Die folgenden Berechtigungen sind erforderlich oder werden vorgeschlagen:

- `ds:DescribeDirectories` ist erforderlich, damit Transfer Family das Verzeichnis nachschlagen kann
- `ds:AuthorizeApplication` ist erforderlich, um die Autorisierung für Transfer Family hinzuzufügen
- `ds:UnauthorizeApplication` wird empfohlen, alle Ressourcen zu entfernen, die vorläufig erstellt wurden, falls während der Servererstellung etwas schief geht

Fügen Sie diese Berechtigungen der Rolle hinzu, die Sie zum Erstellen Ihrer Transfer Family-Server verwenden. Weitere Informationen zu diesen Berechtigungen finden Sie unter AWS Directory Service API-Berechtigungen: Referenz zu Aktionen, Ressourcen und Bedingungen.

Arbeiten mit Active Directory-Bereichen

Wenn Sie darüber nachdenken, wie Ihre Active-Directory-Benutzer auf AWS Transfer Family Server zugreifen können, denken Sie an den Bereich des Benutzers und den Bereich seiner Gruppe. Idealerweise sollten der Bereich des Benutzers und der Bereich seiner Gruppe übereinstimmen. Das heißt, sowohl der Benutzer als auch die Gruppe befinden sich im Standardbereich oder beide befinden sich im vertrauenswürdigen Bereich. Wenn dies nicht der Fall ist, kann der Benutzer nicht von Transfer Family authentifiziert werden.

Sie können den Benutzer testen, um sicherzustellen, dass die Konfiguration korrekt ist. Details hierzu finden Sie unter Testen von Benutzern. Wenn ein Problem mit dem Benutzer-/Gruppenbereich auftritt, erhalten Sie die Fehlermeldung Kein zugeordneter Zugriff für Benutzergruppen gefunden.

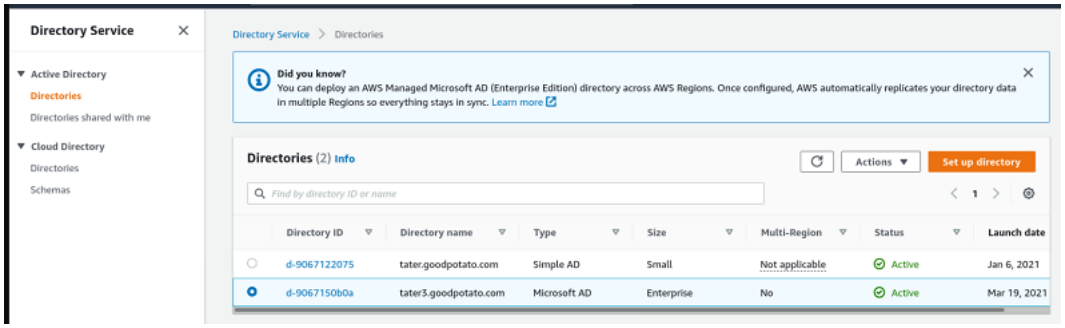
Auswählen von AWS Managed Microsoft AD als Identitätsanbieter

In diesem Abschnitt wird beschrieben, wie Sie AWS Directory Service for Microsoft Active Directory mit einem Server verwenden.

So verwenden Sie AWS Managed Microsoft AD mit Transfer Family

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die - AWS Directory Service Konsole unter https://console.aws.amazon.com/directoryservicev2/.

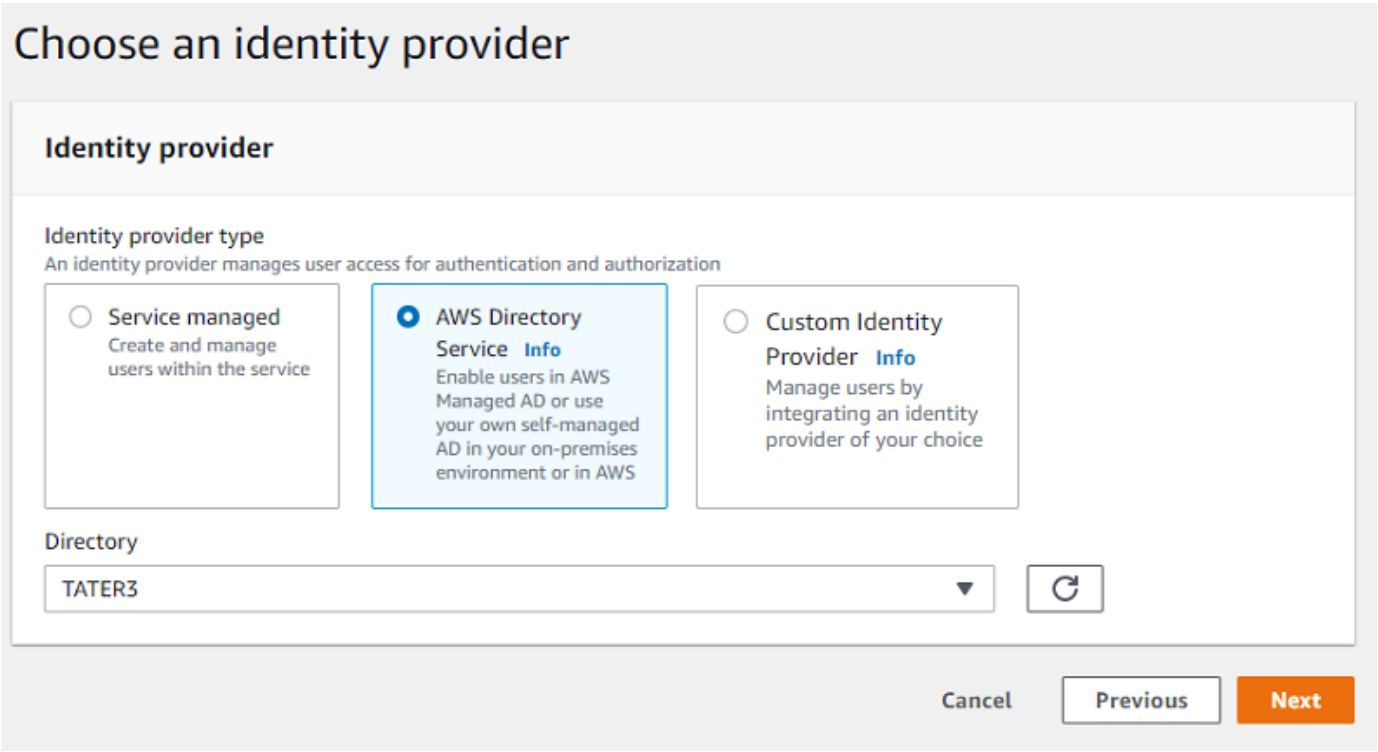
Verwenden Sie die - AWS Directory Service Konsole, um ein oder mehrere verwaltete Verzeichnisse zu konfigurieren. Weitere Informationen finden Sie unter AWS Managed Microsoft AD im Administratorhandbuch für AWS Directory Service .




2. Öffnen Sie die - AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/ und wählen Sie Server erstellen aus.
3. Wählen Sie auf der Seite Protokolle auswählen ein oder mehrere Protokolle aus der Liste aus.

> **Note**
>
> Wenn Sie FTPS auswählen, müssen Sie das AWS Certificate Manager Zertifikat angeben.

4. Wählen Sie für Identitätsanbieter auswählen die Option AWS Directory Service aus.



5. Die Liste Verzeichnis enthält alle von Ihnen konfigurierten verwalteten Verzeichnisse. Wählen Sie ein Verzeichnis aus der Liste aus und klicken Sie auf Weiter.

   > **Note**
   >
   > - Kontoübergreifende Verzeichnisse und freigegebene Verzeichnisse werden für nicht unterstützt AWS Managed Microsoft AD.
   > - Um einen Server mit Directory Service als Identitätsanbieter einzurichten, müssen Sie einige AWS Directory Service Berechtigungen hinzufügen. Details hierzu finden Sie unter Bevor Sie mit der Verwendung von beginnen AWS Directory Service for Microsoft Active Directory.

6. Verwenden Sie eines der folgenden Verfahren, um die Erstellung des Servers abzuschließen:

   - Erstellen Sie einen SFTP-fähigen Server
   - Erstellen Sie einen FTPS-fähigen Server
   - Erstellen Sie einen FTP-fähigen Server

   Fahren Sie bei diesen Verfahren mit dem Schritt fort, der der Auswahl eines Identitätsanbieters folgt.

> **Important**
>
> Sie können ein Microsoft AD-Verzeichnis in nicht löschen AWS Directory Service , wenn Sie es auf einem Transfer Family-Server verwendet haben. Sie müssen zuerst den Server löschen und können dann das Verzeichnis löschen.

Gewähren des Zugriffs auf Gruppen

Nachdem Sie den Server erstellt haben, müssen Sie auswählen, welche Gruppen im Verzeichnis Zugriff zum Hochladen und Herunterladen von Dateien über die aktivierten Protokolle mit haben sollen AWS Transfer Family. Dazu erstellen Sie einen -Zugriff.

> **Note**
>
> Benutzer müssen direkt zu der Gruppe gehören, der Sie Zugriff gewähren. Angenommen, Bob ist ein Benutzer und gehört zu groupA groupA selbst ist in groupB enthalten.
>
> - Wenn Sie Zugriff auf groupA gewähren, erhält Bob Zugriff.
> - Wenn Sie Zugriff auf groupB (und nicht auf groupA) gewähren, hat Bob keinen Zugriff.

So gewähren Sie Zugriff auf eine Gruppe

1. Öffnen Sie die - AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Navigieren Sie zur Seite mit den Serverdetails.
3. Wählen Sie im Abschnitt Zugriffe die Option Zugriff hinzufügen aus.
4. Geben Sie die SID für das AWS Managed Microsoft AD Verzeichnis ein, das Zugriff auf diesen Server haben soll.

   > **Note**
   >
   > Informationen zum Auffinden der SID für Ihre Gruppe finden Sie unter the section called “Bevor Sie mit der Verwendung von beginnen AWS Directory Service for Microsoft Active Directory”.

5. Wählen Sie für Zugriff eine AWS Identity and Access Management (IAM)-Rolle für die Gruppe aus.
6. Wählen Sie im Abschnitt Richtlinie eine Richtlinie aus. Die Standardeinstellung ist Keine .
7. Wählen Sie für Home directory einen S3-Bucket aus, der dem Home-Verzeichnis der Gruppe entspricht.

   > **Note**
   >
   > Sie können die Teile des Buckets einschränken, die Benutzer sehen, indem Sie eine Sitzungsrichtlinie erstellen. Um Benutzer beispielsweise auf ihren eigenen Ordner unter dem `/filetest` Verzeichnis zu beschränken, geben Sie den folgenden Text in das Feld ein.
   >
   > ```
   > /filetest/${transfer:UserName}
   > ```

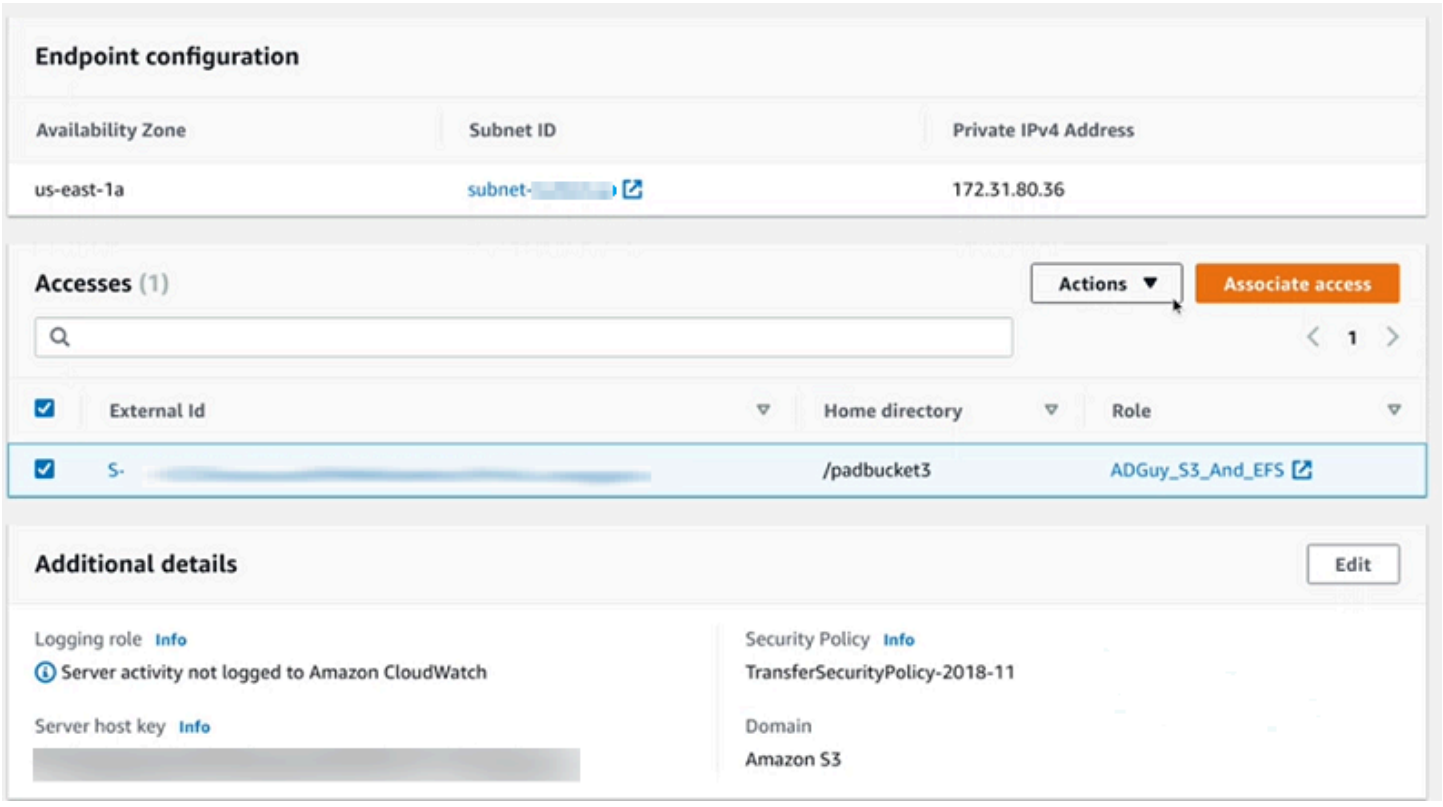
Weitere Informationen zum Erstellen einer Sitzungsrichtlinie finden Sie unter Sitzungsrichtlinie für einen Amazon S3 S3-Bucket erstellen.

8. Wählen Sie Hinzufügen, um die Zuordnung zu erstellen.
9. Wählen Sie Ihren Server aus.
10. Wählen Sie Zugriff hinzufügen aus.
    - Geben Sie die SID für die Gruppe ein.

    > **Note**
    >
    > Informationen zum Auffinden der SID finden Sie unter the section called “Bevor Sie mit der Verwendung von beginnen AWS Directory Service for Microsoft Active Directory”.

11. Wählen Sie Zugriff hinzufügen aus.

Im Abschnitt Zugriffe werden die Zugriffe für den Server aufgelistet.



## Testen von Benutzern

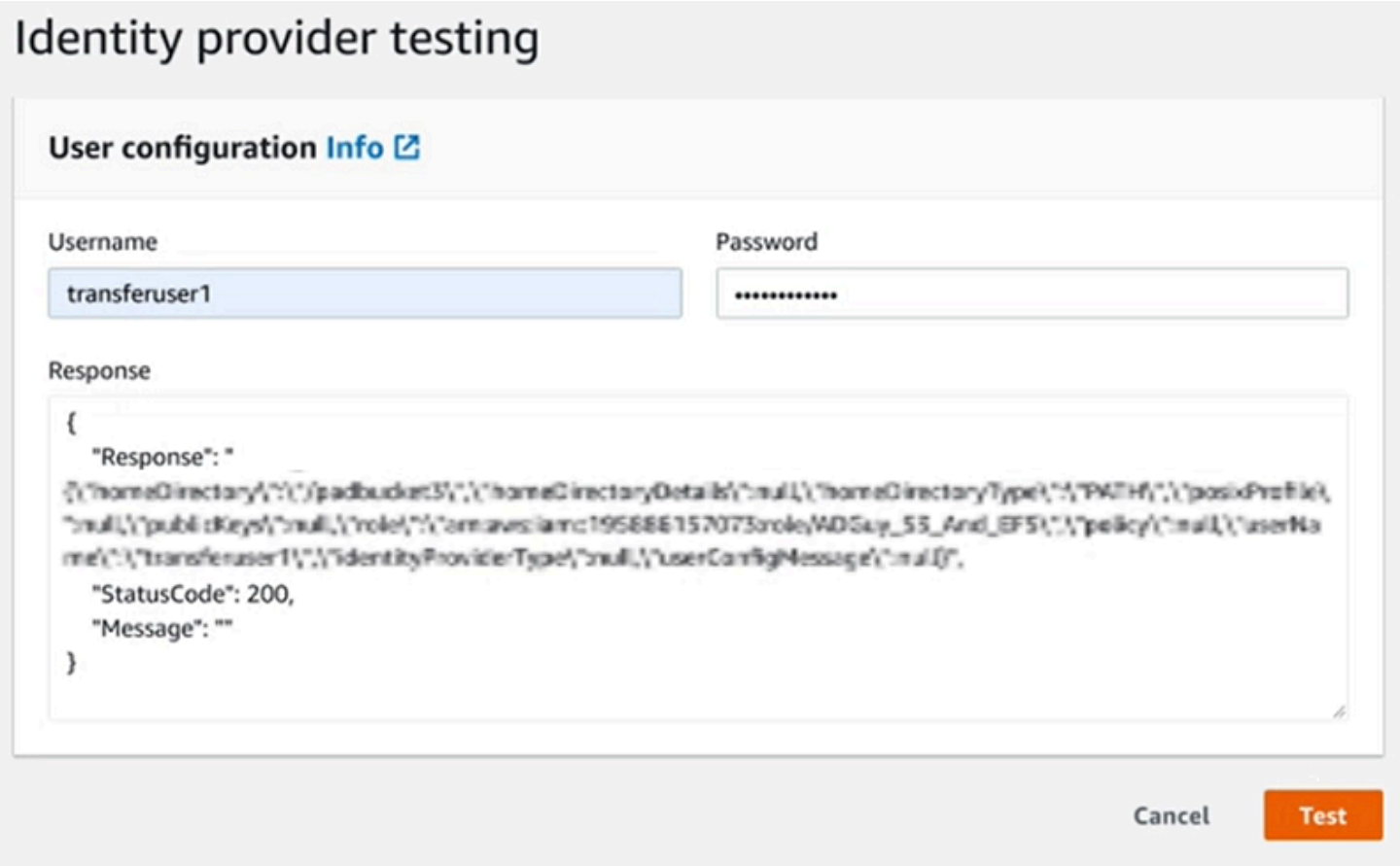
Sie können testen, ob ein Benutzer Zugriff auf das AWS Managed Microsoft AD Verzeichnis für Ihren Server hat.

> **Note**
>
> Ein Benutzer muss sich genau in einer Gruppe (einer externen ID) befinden, die im Abschnitt Zugriff der Seite Endpunktkonfiguration aufgeführt ist. Wenn sich der Benutzer in keiner Gruppe oder in mehr als einer einzigen Gruppe befindet, wird diesem Benutzer kein Zugriff gewährt.

So testen Sie, ob ein bestimmter Benutzer Zugriff hat

1. Wählen Sie auf der Seite mit den Serverdetails Aktionen und dann Testen aus.
2. Geben Sie unter Identitätsanbieter testen die Anmeldeinformationen für einen Benutzer ein, der sich in einer der Gruppen befindet, die Zugriff hat.
3. Wählen Sie Test aus.

Sie sehen einen erfolgreichen Identitätsanbietertest, der zeigt, dass dem ausgewählten Benutzer Zugriff auf den Server gewährt wurde.




Wenn der Benutzer mehreren Gruppen angehört, die Zugriff haben, erhalten Sie die folgende Antwort.

```
"Response":"",
"StatusCode":200,
"Message":"More than one associated access found for user's groups."
```

Löschen des Serverzugriffs für eine Gruppe

So löschen Sie den Serverzugriff für eine Gruppe

1. Wählen Sie auf der Seite mit den Serverdetails Aktionen und dann Zugriff löschen aus.
2. Bestätigen Sie im Dialogfeld, dass Sie den Zugriff für diese Gruppe entfernen möchten.

Wenn Sie zur Seite mit den Serverdetails zurückkehren, sehen Sie, dass der Zugriff für diese Gruppe nicht mehr aufgeführt ist.

Herstellen einer Verbindung mit dem Server über SSH (Secure Shell)

Nachdem Sie Ihren Server und Ihre Benutzer konfiguriert haben, können Sie über SSH eine Verbindung zum Server herstellen und den vollqualifizierten Benutzernamen für einen Benutzer verwenden, der Zugriff hat.

```
sftp user@active-directory-domain@vpc-endpoint
```

Zum Beispiel: `transferuserexample@mycompany.com@vpce-0123456abcdef-789xyz.vpc-svc-987654zyxabc.us-east-1.vpce.amazonaws.com`.

Dieses Format zielt auf die Suche nach dem Verbund ab und beschränkt die Suche nach einem potenziell großen Active Directory.

> **Note**
>
> Sie können den einfachen Benutzernamen angeben. In diesem Fall muss der Active-Directory-Code jedoch alle Verzeichnisse im Verbund durchsuchen. Dies kann die Suche einschränken und die Authentifizierung fehlschlagen, auch wenn der Benutzer Zugriff haben soll.

Nach der Authentifizierung befindet sich der Benutzer in dem Stammverzeichnis, das Sie bei der Konfiguration des Benutzers angegeben haben.

Herstellen einer Verbindung AWS Transfer Family zu einem selbstverwalteten Active Directory mithilfe von Gesamtstruktur und Vertrauensstellungen

Benutzer in Ihrem selbstverwalteten Active Directory (AD) können auch AWS IAM Identity Center für Single-Sign-On-Zugriff auf AWS-Konten und Transfer-Family-Server verwenden. Dazu AWS Directory Service stehen die folgenden Optionen zur Verfügung:

- Eine unidirektionale Gesamtstruktur-Vertrauensstellung (ausgehend von AWS Managed Microsoft AD und eingehend für On-Premises-Active-Directory) funktioniert nur für die Stammdomäne.
- Für untergeordnete Domänen können Sie eine der folgenden Optionen verwenden:
  - Bidirektionale Vertrauensstellung zwischen AWS Managed Microsoft AD und On-Premises-Active-Directory verwenden
  - Verwenden Sie eine unidirektionale externe Vertrauensstellung für jede untergeordnete Domain.

Wenn der Benutzer über eine vertrauenswürdige Domain eine Verbindung zum Server herstellt, muss er die vertrauenswürdige Domain angeben, z. B. `transferuserexample@mycompany.com`.

## Verwenden von AWS Directory Service für Azure Active Directory Domain Services

- Um Ihre bestehende Active-Directory-Gesamtstruktur für Ihre SFTP-Übertragungsanforderungen zu nutzen, können Sie Active Directory Connector verwenden.
- Wenn Sie die Vorteile von Active Directory und Hochverfügbarkeit in einem vollständig verwalteten Service nutzen möchten, können Sie verwenden AWS Directory Service for Microsoft Active Directory. Details hierzu finden Sie unter Verwenden des AWS Directory Service-Identitätsanbieters.

In diesem Thema wird beschrieben, wie Sie einen Active Directory Connector und Azure Active Directory Domain Services (Azure ADDS) verwenden, um SFTP-Transfer-Benutzer mit Azure Active Directory zu authentifizieren.

Themen

- Bevor Sie Directory AWS Service für Azure Active Directory Domain Services verwenden
- Schritt 1: Hinzufügen von Azure Active Directory-Domänenservices
- Schritt 2: Erstellen eines Servicekontos
- Schritt 3: Einrichten des AWS Verzeichnisses mit AD Connector
- Schritt 4: Einrichten des AWS Transfer Family Servers
- Schritt 5: Gewähren des Zugriffs auf Gruppen
- Schritt 6: Testen von Benutzern

### Bevor Sie Directory AWS Service für Azure Active Directory Domain Services verwenden

Für benötigen AWS Sie Folgendes:

- Eine Virtual Private Cloud (VPC) in einer - AWS Region, in der Sie Ihre Transfer Family-Server verwenden
- Mindestens zwei private Subnetze in Ihrer VPC
- Die VPC muss über Internetkonnektivität verfügen
- Ein Kunden-Gateway und ein Virtual Private Gateway für die site-to-site VPN-Verbindung mit Microsoft Azure

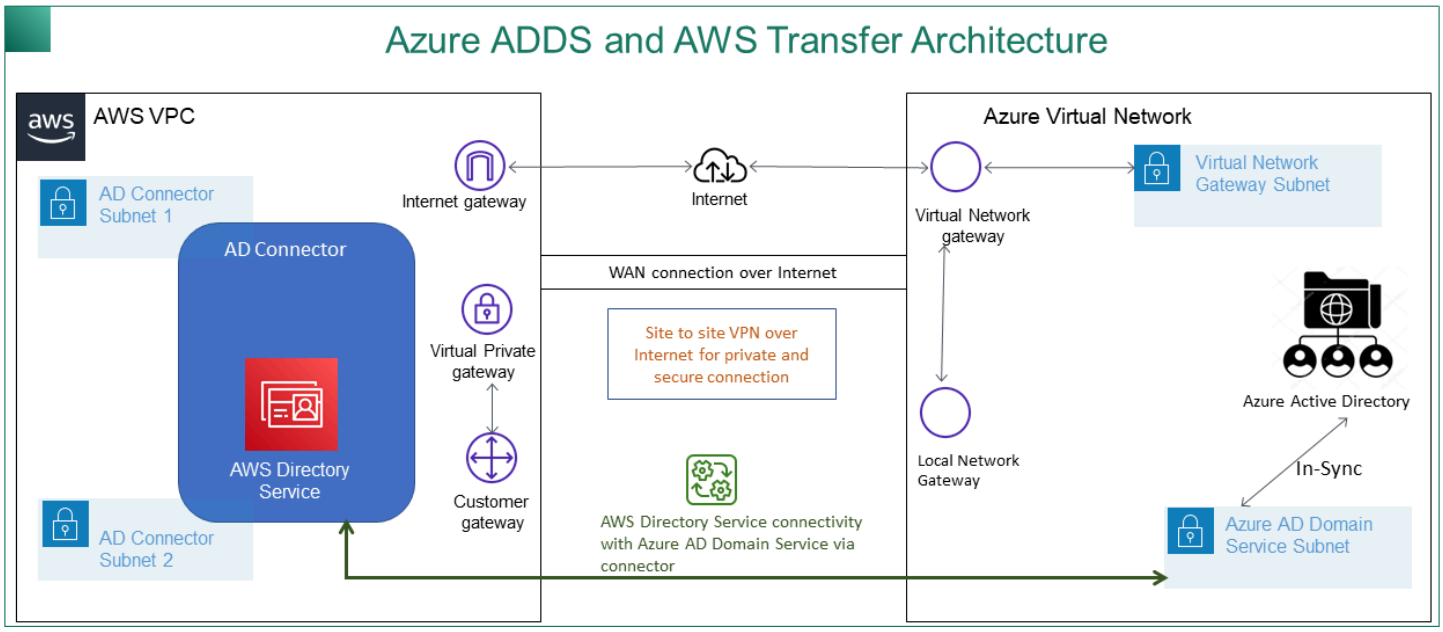
Für Microsoft Azure benötigen Sie Folgendes:

- Ein Azure Active Directory- und Active Directory-Domänenservice (Azure ADDS)
- Eine Azure-Ressourcengruppe
- Ein virtuelles Azure-Netzwerk
- VPN-Konnektivität zwischen Ihrer Amazon VPC und Ihrer Azure-Ressourcengruppe

> **Note**
>
> Dies kann über native IPSEC-Tunnel oder mithilfe von VPN-Appliances erfolgen. In diesem Thema verwenden wir IPSEC-Tunnel zwischen einem Azure Virtual Network Gateway und einem lokalen Netzwerk-Gateway. Die Tunnel müssen so konfiguriert sein, dass sie Datenverkehr zwischen Ihren Azure-ADDS-Endpunkten und den Subnetzen zulassen, in denen sich Ihre AWS VPC befindet.

- Ein Kunden-Gateway und ein Virtual Private Gateway für die site-to-site VPN-Verbindung mit Microsoft Azure

Das folgende Diagramm zeigt die Konfiguration, die erforderlich ist, bevor Sie beginnen.



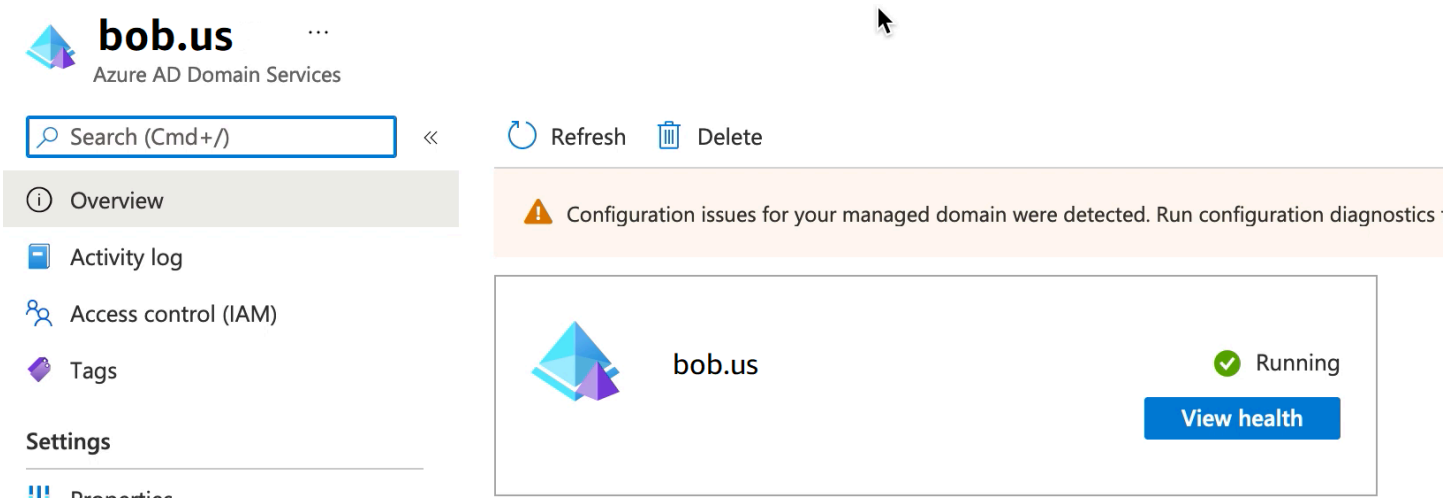

## Schritt 1: Hinzufügen von Azure Active Directory-Domänenservices

Azure AD unterstützt standardmäßig keine Domain-Joining-Instances. Um Aktionen wie Domain Join auszuführen und Tools wie Gruppenrichtlinien zu verwenden, müssen Administratoren Azure Active

Directory Domain Services aktivieren. Wenn Sie Azure AD DS noch nicht hinzugefügt haben oder Ihre vorhandene Implementierung nicht mit der Domain verknüpft ist, die Ihr SFTP-Transfer-Server verwenden soll, müssen Sie eine neue Instance hinzufügen.

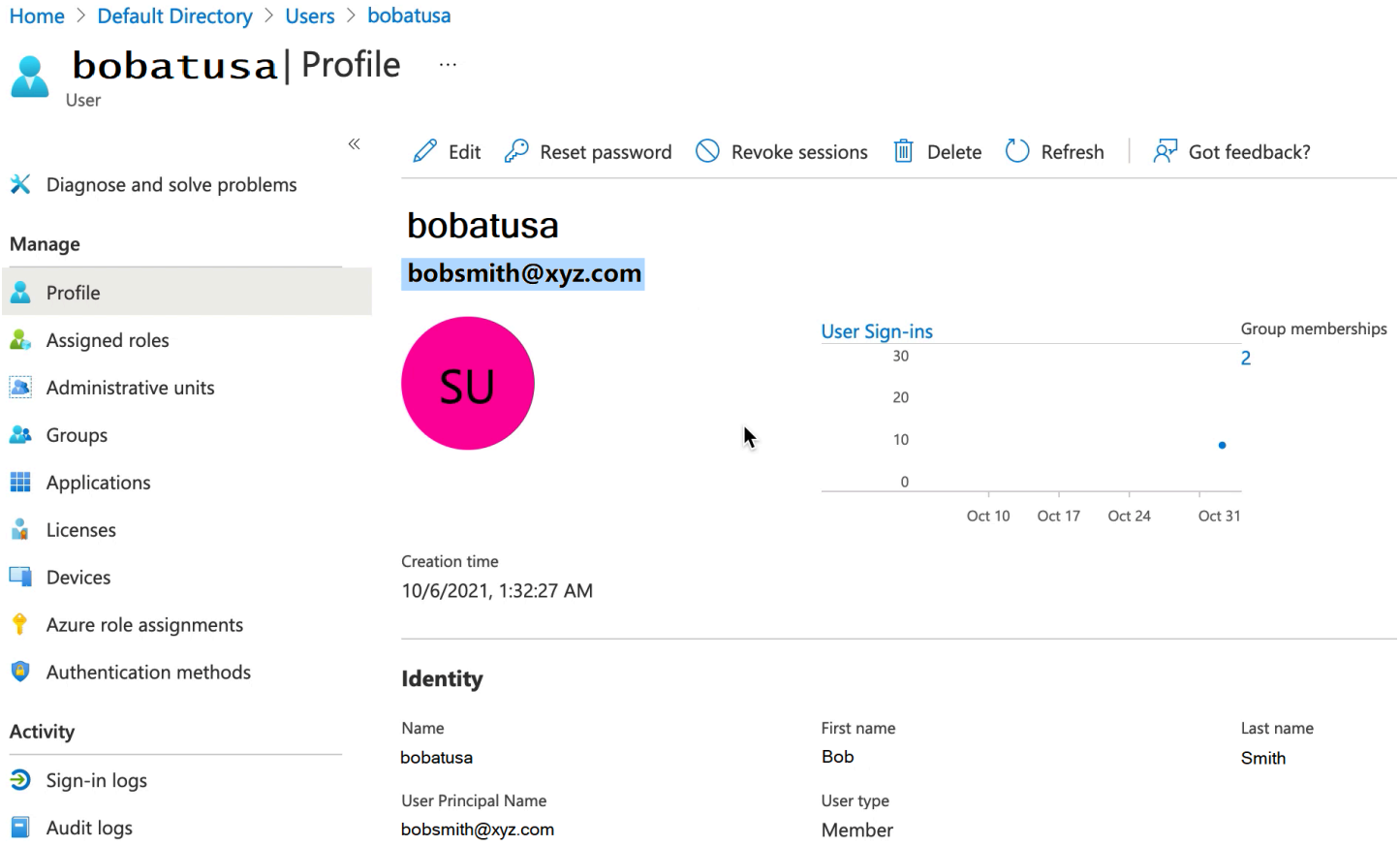
Informationen zum Aktivieren von Azure Active Directory Domain Services (Azure ADDS) finden Sie unter Tutorial: Erstellen und Konfigurieren einer verwalteten Domain von Azure Active Directory Domain Services.

> **Note**
>
> Wenn Sie Azure ADDS aktivieren, stellen Sie sicher, dass es für die Ressourcengruppe und die Azure-AD-Domäne konfiguriert ist, mit der Sie Ihren SFTP-Transfer-Server verbinden.

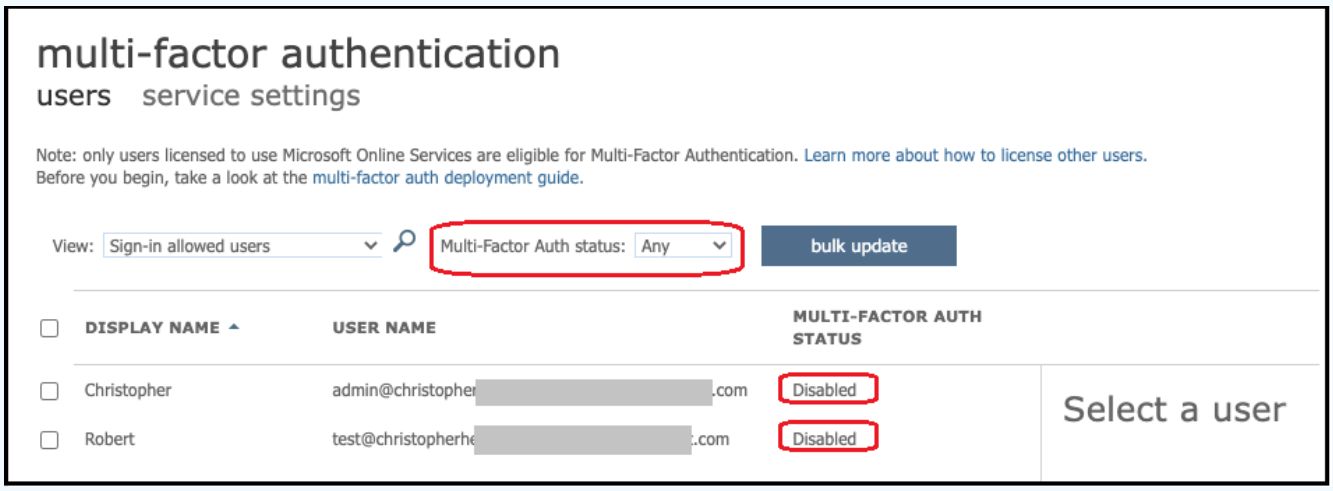



Schritt 2: Erstellen eines Servicekontos

Azure AD muss über ein Servicekonto verfügen, das Teil einer Admin-Gruppe in Azure ADDS ist. Dieses Konto wird mit dem AWS Active-Directory-Konnektor verwendet. Stellen Sie sicher, dass dieses Konto mit Azure ADDS synchronisiert ist.




> **Tip**
>
> Die Multi-Faktor-Authentifizierung für Azure Active Directory wird für Transfer Family-Server, die das SFTP-Protokoll verwenden, nicht unterstützt. Der Transfer Family-Server kann das MFA-Token nicht bereitstellen, nachdem sich ein Benutzer bei SFTP authentifiziert hat. Stellen Sie sicher, dass Sie MFA deaktivieren, bevor Sie versuchen, eine Verbindung herzustellen.




Schritt 3: Einrichten des AWS Verzeichnisses mit AD Connector

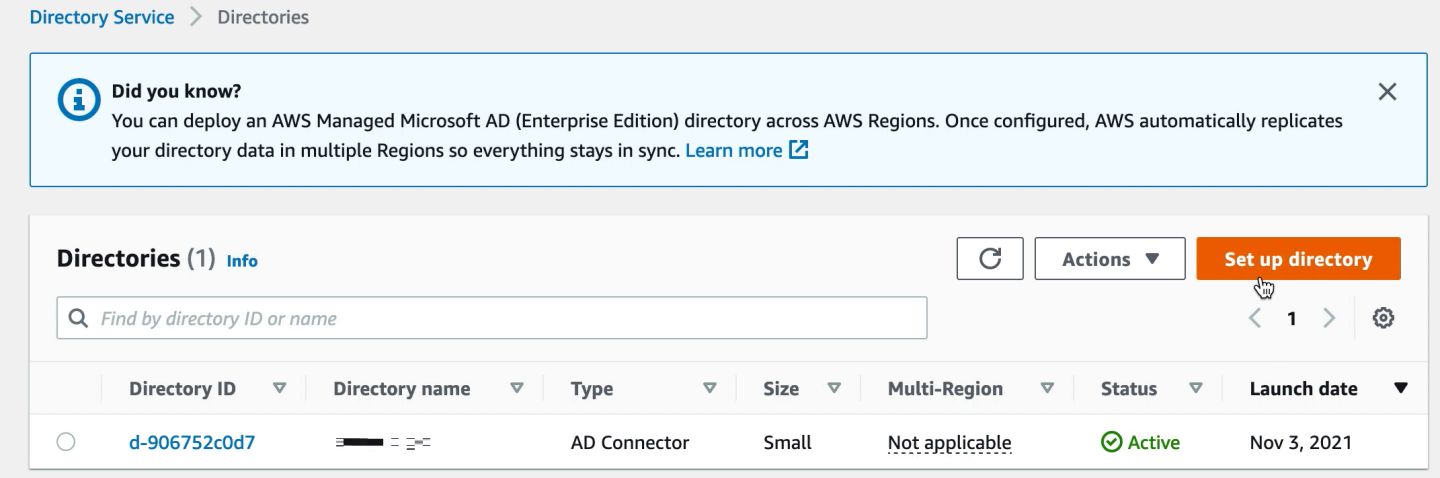
Nachdem Sie Azure ADDS konfiguriert und ein Servicekonto mit IPSEC-VPN-Tunneln zwischen Ihrer AWS VPC und dem Azure Virtual Network erstellt haben, können Sie die Konnektivität testen, indem Sie von jeder AWS EC2-Instance aus einen Ping an die Azure-ADDS-DNS-IP-Adresse senden.

Nachdem Sie überprüft haben, dass die Verbindung aktiv ist, können Sie weiter unten fortfahren.

So richten Sie Ihr AWS Verzeichnis mit AD Connector ein

1. Öffnen Sie die Directory-Service-Konsole und wählen Sie Verzeichnisse aus.
2. Wählen Sie Verzeichnis einrichten aus.
3. Wählen Sie als Verzeichnistyp AD Connector aus.
4. Wählen Sie eine Verzeichnisgröße, dann Weiter und dann Ihre VPC und Subnetze aus.
5. Wählen Sie Weiter und füllen Sie die Felder wie folgt aus:
   - Verzeichnis-DNS-Name: Geben Sie den Domänennamen ein, den Sie für Ihr Azure ADDS verwenden.
   - DNS-IP-Adressen: Geben Sie Ihre Azure-ADDS-IP-Adressen ein.
   - Benutzername und Passwort des Serverkontos: Geben Sie die Details für das Servicekonto ein, das Sie in Schritt 2: Erstellen eines Servicekontos erstellt haben.
6. Füllen Sie die Bildschirme aus, um den Verzeichnisdienst zu erstellen.

Jetzt sollte der Verzeichnisstatus Aktiv lauten und es kann mit einem SFTP-Transferserver verwendet werden.




## Schritt 4: Einrichten des AWS Transfer Family Servers

Erstellen Sie einen Transfer Family-Server mit dem SFTP-Protokoll und dem Identitätsanbietertyp von AWS Directory Service . Wählen Sie in der Dropdown-Liste Verzeichnis das Verzeichnis aus, das Sie in Schritt 3: AWS Verzeichnis mit AD Connector einrichten hinzugefügt haben.

> **Note**
>
> Sie können ein Microsoft AD-Verzeichnis in AWS Directory Service nicht löschen, wenn Sie es auf einem Transfer Family-Server verwendet haben. Sie müssen zuerst den Server löschen und können dann das Verzeichnis löschen.

## Schritt 5: Gewähren des Zugriffs auf Gruppen

Nachdem Sie den Server erstellt haben, müssen Sie auswählen, welche Gruppen im Verzeichnis Zugriff zum Hochladen und Herunterladen von Dateien über die aktivierten Protokolle mit haben sollen AWS Transfer Family. Dazu erstellen Sie einen -Zugriff.

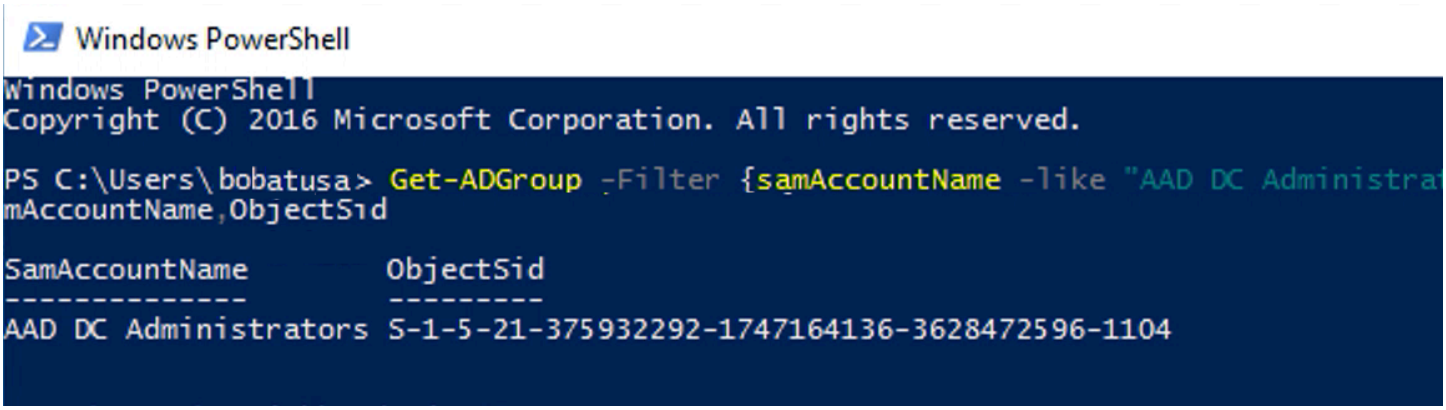
> **Note**
>
> Benutzer müssen direkt zu der Gruppe gehören, der Sie Zugriff gewähren. Angenommen, Bob ist ein Benutzer und gehört zu groupA groupA selbst ist in groupB enthalten.
>
> - Wenn Sie Zugriff auf groupA gewähren, erhält Bob Zugriff.

- Wenn Sie Zugriff auf groupB (und nicht auf groupA) gewähren, hat Bob keinen Zugriff.

Um Zugriff zu gewähren, müssen Sie die SID für die Gruppe abrufen.

Verwenden Sie den folgenden Windows- PowerShell Befehl, um die SID für eine Gruppe abzurufen und durch *YourGroupName* den Namen der Gruppe zu ersetzen.

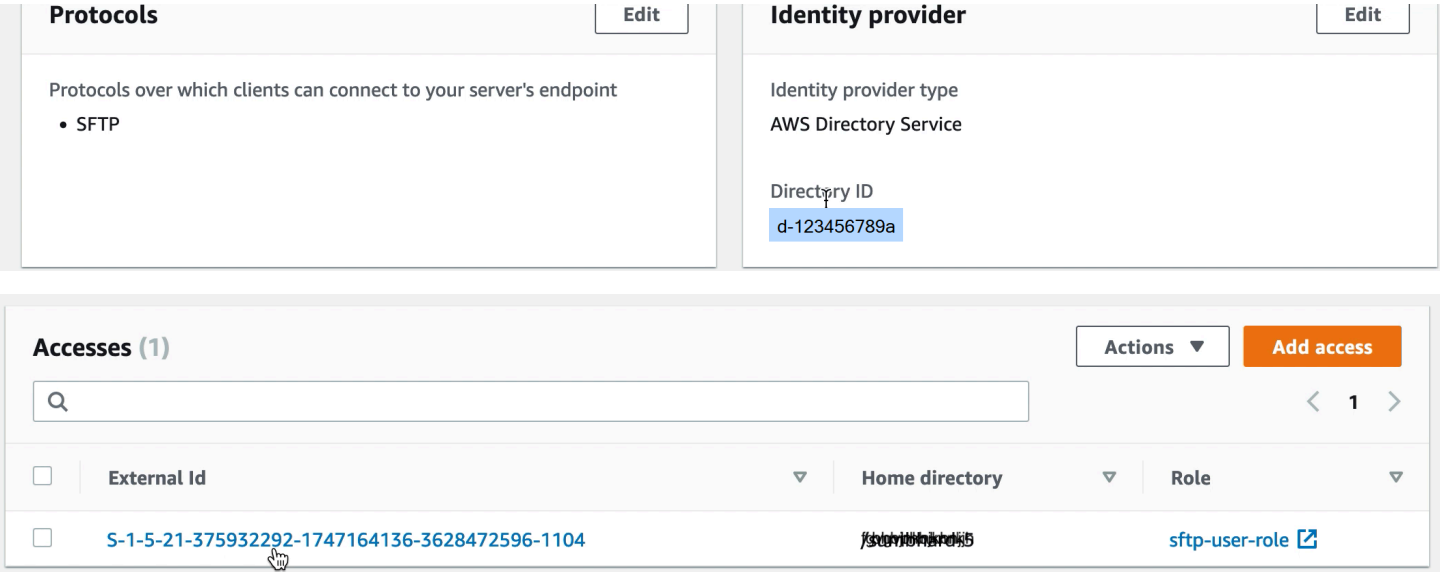
```
Get-ADGroup -Filter {samAccountName -like "YourGroupName*"} -Properties * | Select
 SamAccountName,ObjectSid
```




Gewähren von Zugriff auf Gruppen

1. Öffnen Sie https://console.aws.amazon.com/transfer/.
2. Navigieren Sie zur Seite mit den Serverdetails und wählen Sie im Abschnitt Zugriff hinzufügen aus.
3. Geben Sie die SID ein, die Sie aus der Ausgabe des vorherigen Verfahrens erhalten haben.
4. Wählen Sie für Zugriff eine - AWS Identity and Access Management Rolle für die Gruppe aus.
5. Wählen Sie im Abschnitt Richtlinie eine Richtlinie aus. Der Standardwert ist None (Kein).
6. Wählen Sie für Home directory einen S3-Bucket aus, der dem Home-Verzeichnis der Gruppe entspricht.
7. Wählen Sie Hinzufügen, um die Zuordnung zu erstellen.

Die Details von Ihrem Transfer-Server sollten in etwa wie folgt aussehen:




Schritt 6: Testen von Benutzern

Sie können (Testen von Benutzern) testen, ob ein Benutzer Zugriff auf das AWS Managed Microsoft AD Verzeichnis für Ihren Server hat. Ein Benutzer muss sich genau in einer Gruppe (einer externen ID) befinden, die im Abschnitt Zugriff der Seite Endpunktkonfiguration aufgeführt ist. Wenn sich der Benutzer in keiner Gruppe oder in mehr als einer einzigen Gruppe befindet, wird diesem Benutzer kein Zugriff gewährt.

## Mit Anbietern benutzerdefinierter Identitäten arbeiten

Um Ihre Benutzer zu authentifizieren, können Sie Ihren vorhandenen Identitätsanbieter mit AWS Transfer Family verwenden. Sie integrieren Ihren Identitätsanbieter mithilfe einer AWS Lambda Funktion, die Ihre Benutzer authentifiziert und für den Zugriff auf Amazon S3 oder Amazon Elastic File System (Amazon EFS) autorisiert. Details hierzu finden Sie unter Wird AWS Lambda zur Integration Ihres Identitätsanbieters verwendet. In der AWS Transfer Family Management Console können Sie auch auf CloudWatch Diagramme für Messwerte wie die Anzahl der übertragenen Dateien und Byte zugreifen, sodass Sie Dateiübertragungen über ein zentrales Dashboard überwachen können.

Alternativ können Sie eine RESTful-Schnittstelle mit einer einzigen Amazon API Gateway Gateway-Methode bereitstellen. Transfer Family ruft diese Methode auf, um eine Verbindung zu Ihrem Identitätsanbieter herzustellen, der Ihre Benutzer authentifiziert und für den Zugriff auf Amazon S3 oder Amazon EFS autorisiert. Verwenden Sie diese Option, wenn Sie eine RESTful-API zur Integration Ihres Identitätsanbieters benötigen oder wenn Sie dessen Funktionen für Geoblocking-

oder Ratenbegrenzungsanfragen nutzen AWS WAF möchten. Details hierzu finden Sie unter Verwenden Sie Amazon API Gateway zur Integration Ihres Identitätsanbieters.

In beiden Fällen können Sie mithilfe der AWS Transfer Family Konsole oder der API-Operation einen neuen Server erstellen. CreateServer

> **Note**
>
> Transfer Family bietet einen Blogbeitrag und einen Workshop, die Sie durch den Aufbau einer Dateiübertragungslösung führen. Diese Lösung nutzt AWS Transfer Family verwaltete SFTP/FTPS-Endpunkte sowie Amazon Cognito und DynamoDB für die Benutzerverwaltung. Der Blogbeitrag ist unter Amazon Cognito als Identitätsanbieter mit AWS Transfer Family Amazon S3 verwenden verfügbar. Die Details zum Workshop finden Sie hier.

AWS Transfer Family bietet die folgenden Optionen für die Arbeit mit benutzerdefinierten Identitätsanbietern.

- Verwenden Sie AWS Lambda , um Ihren Identitätsanbieter zu verbinden — Sie können einen vorhandenen Identitätsanbieter verwenden, der von einer Lambda-Funktion unterstützt wird. Sie geben den Namen der Lambda-Funktion an. Weitere Informationen finden Sie unter Wird AWS Lambda zur Integration Ihres Identitätsanbieters verwendet.
- Verwenden Sie Amazon API Gateway, um Ihren Identitätsanbieter zu verbinden — Sie können eine API Gateway Gateway-Methode erstellen, die von einer Lambda-Funktion unterstützt wird, um sie als Identitätsanbieter zu verwenden. Sie geben eine Amazon API Gateway Gateway-URL und eine Aufrufrolle an. Weitere Informationen finden Sie unter Verwenden Sie Amazon API Gateway zur Integration Ihres Identitätsanbieters.

Für beide Optionen können Sie auch angeben, wie Sie sich authentifizieren möchten.

- Passwort ODER Schlüssel — Benutzer können sich entweder mit ihrem Passwort oder ihrem Schlüssel authentifizieren. Dies ist der Standardwert.
- NUR Passwort — Benutzer müssen ihr Passwort angeben, um eine Verbindung herzustellen.
- NUR Schlüssel — Benutzer müssen ihren privaten Schlüssel angeben, um eine Verbindung herzustellen.
- Passwort UND Schlüssel — Benutzer müssen sowohl ihren privaten Schlüssel als auch ihr Passwort angeben, um eine Verbindung herzustellen. Der Server überprüft zuerst den Schlüssel.

Wenn der Schlüssel gültig ist, fordert das System dann zur Eingabe eines Kennworts auf. Wenn der angegebene private Schlüssel nicht mit dem gespeicherten öffentlichen Schlüssel übereinstimmt, schlägt die Authentifizierung fehl.

## Verwenden Sie mehrere Authentifizierungsmethoden, um sich bei Ihrem benutzerdefinierten Identitätsanbieter zu authentifizieren

Der Transfer Family Family-Server steuert die AND-Logik, wenn Sie mehrere Authentifizierungsmethoden verwenden. Transfer Family behandelt dies als zwei separate Anfragen an Ihren benutzerdefinierten Identitätsanbieter: Ihre Wirkung ist jedoch kombiniert.

Beide Anfragen müssen erfolgreich mit der richtigen Antwort zurückgegeben werden, damit die Authentifizierung abgeschlossen werden kann. Transfer Family setzt voraus, dass die beiden Antworten vollständig sind, d. h. sie enthalten alle erforderlichen Elemente (Rolle, Home-Verzeichnis, Richtlinie und das POSIX-Profil, wenn Sie Amazon EFS als Speicher verwenden). Transfer Family verlangt außerdem, dass die Passwortantwort keine öffentlichen Schlüssel enthalten darf.

Die Anfrage nach einem öffentlichen Schlüssel muss vom Identitätsanbieter separat beantwortet werden. Dieses Verhalten bleibt unverändert, wenn Password OR Key oder Password AND Key verwendet wird.

Das SSH/SFTP-Protokoll fordert den Software-Client zunächst mit einer Authentifizierung über einen öffentlichen Schlüssel auf und fordert dann eine Kennwortauthentifizierung an. Dieser Vorgang setzt voraus, dass beide erfolgreich sind, bevor der Benutzer die Authentifizierung abschließen kann.

Themen

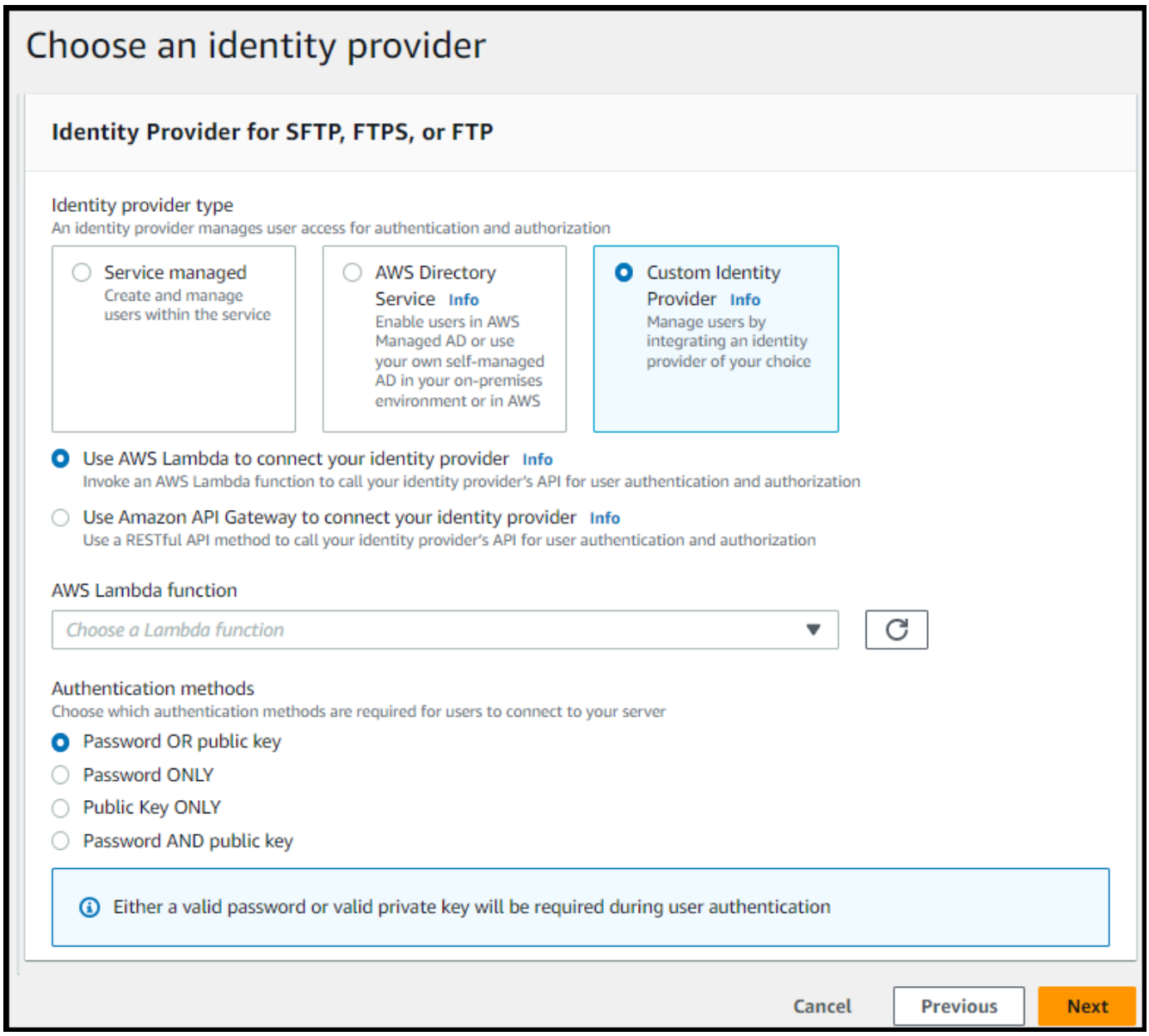
- Wird AWS Lambda zur Integration Ihres Identitätsanbieters verwendet
- Verwenden Sie Amazon API Gateway zur Integration Ihres Identitätsanbieters

## Wird AWS Lambda zur Integration Ihres Identitätsanbieters verwendet

Erstellen Sie eine AWS Lambda Funktion, die eine Verbindung zu Ihrem benutzerdefinierten Identitätsanbieter herstellt. Sie können einen beliebigen benutzerdefinierten Identitätsanbieter wie Okta, Secrets Manager oder einen benutzerdefinierten Datenspeicher verwenden OneLogin, der Autorisierungs- und Authentifizierungslogik enthält.

> **Note**
>
> Bevor Sie einen Transfer Family Family-Server erstellen, der Lambda als Identitätsanbieter verwendet, müssen Sie die Funktion erstellen. Eine Beispielfunktion für Lambda finden Sie unter Beispiele für Lambda-Funktionen. Oder Sie können einen CloudFormation Stack bereitstellen, der einen der Lambda-Funktionsvorlagen folgenden verwendet. Stellen Sie außerdem sicher, dass Ihre Lambda-Funktion eine ressourcenbasierte Richtlinie verwendet, die Transfer Family vertraut. Eine Beispielrichtlinie finden Sie unter Ressourcenbasierte Lambda-Richtlinie.

1. Öffnen Sie die AWS Transfer Family -Konsole.
2. Wählen Sie Server erstellen, um die Seite Server erstellen zu öffnen. Wählen Sie für Wählen Sie einen Identitätsanbieter die Option Benutzerdefinierter Identitätsanbieter aus, wie im folgenden Screenshot gezeigt.




> **Note**
>
> Die Wahl der Authentifizierungsmethoden ist nur verfügbar, wenn Sie SFTP als eines der Protokolle für Ihren Transfer Family Family-Server aktivieren.

3. Stellen Sie sicher, dass der Standardwert „ AWS Lambda Zur Verbindung Ihres Identitätsanbieters verwenden“ ausgewählt ist.
4. Wählen Sie unter AWS Lambda Funktion den Namen Ihrer Lambda-Funktion.

5. Füllen Sie die verbleibenden Felder aus und wählen Sie dann Server erstellen aus. Einzelheiten zu den verbleibenden Schritten zum Erstellen eines Servers finden Sie unterKonfiguration eines SFTP-, FTPS- oder FTP-Serverendpunkts.

### Ressourcenbasierte Lambda-Richtlinie

Sie benötigen eine Richtlinie, die auf den Transfer Family Family-Server und Lambda-ARNs verweist. Sie könnten beispielsweise die folgende Richtlinie mit Ihrer Lambda-Funktion verwenden, die eine Verbindung zu Ihrem Identitätsanbieter herstellt. Bei der Richtlinie wird JSON als Zeichenfolge maskiert.

```
"Policy":
"{
  "Version": "2012-10-17",
  "Id": "default",
  "Statement": [
    {
      "Sid": "AllowTransferInvocation",
      "Effect": "Allow",
      "Principal": {
        "Service": "transfer.amazonaws.com"
      },
      "Action": "lambda:InvokeFunction",
      "Resource": "arn:aws:transfer:region:account-id:function:my-lambda-auth-
function",
      "Condition": {
        "ArnLike": {
          "AWS:SourceArn": "arn:aws:transfer:region:account-id:server/server-id"
        }
      }
    }
  ]
}"
```

> **Note**
>
> Ersetzen Sie in der obigen Beispielrichtlinie jeden *Platzhalter für Benutzereingaben* durch Ihre eigenen Informationen.

Struktur von Ereignismeldungen

Die Struktur der Ereignisnachrichten, die vom SFTP-Server an die Lambda-Funktion des Autorisierers für einen benutzerdefinierten IDP gesendet werden, lautet wie folgt.

```
{
    "username": "value",
    "password": "value",
    "protocol": "SFTP",
    "serverId": "s-abcd123456",
    "sourceIp": "192.168.0.100"
}
```

Wo `username` und `password` sind die Werte für die Anmeldeinformationen, die an den Server gesendet werden.

Sie geben beispielsweise den folgenden Befehl ein, um eine Verbindung herzustellen:

```
sftp bobusa@server_hostname
```

Sie werden dann aufgefordert, Ihr Passwort einzugeben:

```
Enter password:
    mysecretpassword
```

Sie können dies von Ihrer Lambda-Funktion aus überprüfen, indem Sie das übergebene Ereignis aus der Lambda-Funktion heraus drucken. Es sollte dem folgenden Textblock ähneln.

```
{
    "username": "bobusa",
    "password": "mysecretpassword",
    "protocol": "SFTP",
    "serverId": "s-abcd123456",
    "sourceIp": "192.168.0.100"
}
```

Die Ereignisstruktur ist für FTP und FTPS ähnlich: Der einzige Unterschied besteht darin, dass diese Werte für den `protocol` Parameter und nicht für SFTP verwendet werden.

## Lambda-Funktionen für die Authentifizierung

Bearbeiten Sie die Lambda-Funktion, um verschiedene Authentifizierungsstrategien zu implementieren. Um die Anforderungen Ihrer Anwendung zu erfüllen, können Sie einen CloudFormation Stack bereitstellen. Weitere Informationen zu Lambda finden Sie im AWS Lambda Developer Guide oder im Building Lambda functions with Node.js.

Themen

- Lambda-Funktionsvorlagen
- Gültige Lambda-Werte
- Beispiele für Lambda-Funktionen
- Testen Sie Ihre Konfiguration

### Lambda-Funktionsvorlagen

Sie können einen AWS CloudFormation Stack bereitstellen, der eine Lambda-Funktion zur Authentifizierung verwendet. Wir stellen mehrere Vorlagen zur Verfügung, mit denen Sie Ihre Benutzer mithilfe von Anmeldeinformationen authentifizieren und autorisieren können. Sie können diese Vorlagen oder den AWS Lambda Code ändern, um den Benutzerzugriff weiter anzupassen.

> **Note**
>
> Sie können einen FIPS-fähigen AWS Transfer Family Server erstellen, AWS CloudFormation indem Sie in Ihrer Vorlage eine FIPS-fähige Sicherheitsrichtlinie angeben. Die verfügbaren Sicherheitsrichtlinien werden unter beschrieben Sicherheitsrichtlinien für AWS Transfer Family Server

Um einen AWS CloudFormation Stack für die Authentifizierung zu erstellen

1. Öffnen Sie die AWS CloudFormation Konsole unter https://console.aws.amazon.com/cloudformation.
2. Folgen Sie den Anweisungen zum Bereitstellen eines AWS CloudFormation Stacks anhand einer vorhandenen Vorlage im Abschnitt Auswahl einer Stack-Vorlage im AWS CloudFormation Benutzerhandbuch.
3. Verwenden Sie eine der folgenden Vorlagen, um eine Lambda-Funktion für die Authentifizierung in Transfer Family zu erstellen.

- Klassische (Amazon Cognito) Stack-Vorlage

  Eine grundlegende Vorlage zum Erstellen einer Vorlage AWS Lambda zur Verwendung als benutzerdefinierter Identitätsanbieter in AWS Transfer Family. Es authentifiziert sich bei Amazon Cognito für die kennwortbasierte Authentifizierung und öffentliche Schlüssel werden aus einem Amazon S3 S3-Bucket zurückgegeben, wenn eine Authentifizierung auf Basis eines öffentlichen Schlüssels verwendet wird. Nach der Bereitstellung können Sie den Lambda-Funktionscode ändern, um etwas anderes zu tun.

- AWS Secrets Manager Vorlage stapeln

  Eine grundlegende Vorlage, die AWS Lambda zusammen mit einem AWS Transfer Family Server verwendet wird, um Secrets Manager als Identitätsanbieter zu integrieren. Sie authentifiziert sich anhand eines Eintrags in AWS Secrets Manager diesem Format`aws/transfer/`*`server-id`*`/`*`username`*. Darüber hinaus muss das Geheimnis die Schlüssel-Wert-Paare für alle an Transfer Family zurückgegebenen Benutzereigenschaften enthalten. Nach der Bereitstellung können Sie den Lambda-Funktionscode ändern, um etwas anderes zu tun.

- Okta-Stack-Vorlage: Eine Basisvorlage, die AWS Lambda zusammen mit einem AWS Transfer Family Server verwendet wird, um Okta als benutzerdefinierten Identitätsanbieter zu integrieren.

- Okta-MFA-Stack-Vorlage: Eine Basisvorlage, die AWS Lambda zusammen mit einem AWS Transfer Family Server verwendet wird, um Okta mit MultiFactor Authentifizierung als benutzerdefinierten Identitätsanbieter zu integrieren.

- Azure Active Directory-Vorlage: Einzelheiten zu diesem Stack werden im Blogbeitrag Authentifizierung AWS Transfer Family mit Azure Active Directory und beschrieben. AWS Lambda

Nachdem der Stack bereitgestellt wurde, können Sie Details dazu auf der Registerkarte Ausgaben in der CloudFormation Konsole einsehen.

Die Bereitstellung eines dieser Stacks ist der einfachste Weg, einen benutzerdefinierten Identitätsanbieter in den Transfer Family Family-Workflow zu integrieren.

### Gültige Lambda-Werte

In der folgenden Tabelle werden Details zu den Werten beschrieben, die Transfer Family für Lambda-Funktionen akzeptiert, die für benutzerdefinierte Identitätsanbieter verwendet werden.

| Wert | Beschreibung | Erforderlich |
| --- | --- | --- |
| `Role` | Gibt den Amazon-Ressourcennamen (ARN) der IAM-Rolle an, die den Zugriff Ihrer Benutzer auf Ihren Amazon S3-Bucket oder Ihr Amazon EFS-Dateisystem steuert. Die mit dieser Rolle verknüpften Richtlinien bestimmen die Zugriffsebene, die Sie Ihren Benutzern beim Übertragen von Dateien in und aus Ihrem Amazon S3- oder Amazon EFS-Dateisystem gewähren möchten. Die IAM-Rolle sollte außerdem eine Vertrauensstellung enthalten, mit der der Server Zugriff auf Ihre Ressourcen erhält, wenn er die Übertragungsanfragen Ihres Benutzers bearbeitet.<br><br>Einzelheiten zum Aufbau einer Vertrauensbeziehung finden Sie unterSo stellen Sie eine Vertrauensbeziehung her. | Erforderlich |
| `PosixProfile` | Die vollständige POSIX-Identität, einschließlich Benutzer-ID (`Uid`), Gruppen-ID (`Gid`) und aller sekundären Gruppen-IDs (`Secondary` | Erforderlich für Amazon EFS-Backing-Speicher |

| Wert | Beschreibung | Erforderlich |
| --- | --- | --- |
| | `Gids` ), die den Zugriff Ihrer Benutzer auf Ihre Amazon EFS-Dateisysteme steuert. Die POSIX-Berechtigungen, die für Dateien und Verzeichnisse in Ihrem Dateisystem festgelegt sind, bestimmen die Zugriffsebene, die Ihre Benutzer beim Übertragen von Dateien in und aus Ihren Amazon EFS-Dateisystemen erhalten. | |
| `PublicKeys` | Eine Liste der Werte für öffentliche SSH-Schlüssel, die für diesen Benutzer gültig sind. Eine leere Liste bedeutet, dass dies kein gültiges Login ist. Darf bei der Passwortauthentifizierung nicht zurückgegeben werden. | Optional |
| `Policy` | Eine Sitzungsrichtlinie für Ihren Benutzer, sodass Sie dieselbe IAM-Rolle für mehrere Benutzer verwenden können. Diese Richtlinie grenzt den Benutzerzugriff auf Teile ihres Amazon S3-Buckets ein. | Optional |

| Wert | Beschreibung | Erforderlich |
| --- | --- | --- |
| `HomeDirectoryType` | Die Art des Zielverzeichnisses (Ordners), das das Home-Verzeichnis Ihrer Benutzer sein soll, wenn sie sich beim Server anmelden.<br>• Wenn Sie es auf einstelle n`PATH`, sieht der Benutzer die absoluten Amazon S3 S3-Bucket- oder Amazon EFS-Pfade wie in seinen File Transfer Protocol-Clients.<br>• Wenn Sie es auf einstelle n`LOGICAL`, müssen Sie im `HomeDirectoryDetails` Parameter Zuordnungen angeben, um Amazon S3- oder Amazon EFS-Pfade für Ihre Benutzer sichtbar zu machen. | Optional |

| Wert | Beschreibung | Erforderlich |
| --- | --- | --- |
| `HomeDirectoryDetails` | Logische Verzeichniszuordnungen, die angeben, welche Amazon S3- oder Amazon EFS-Pfade und -Schlüssel für Ihren Benutzer sichtbar sein sollen und wie Sie sie sichtbar machen möchten. Sie müssen das `Entry` `Target` Und-Paar angeben, das `Entry` zeigt, wie der Pfad sichtbar gemacht `Target` wird und der tatsächliche Amazon S3- oder Amazon EFS-Pfad ist. | Erforderlich, wenn `HomeDirectoryType` der Wert `LOGICAL` |
| `HomeDirectory` | Das Zielverzeichnis für einen Benutzer, wenn er sich über den Client am Server anmeldet. | Optional |

> **Note**
>
> `HomeDirectoryDetails`ist eine Zeichenkettendarstellung einer JSON-Map. Dies steht im Gegensatz zu`PosixProfile`, was ein eigentliches JSON-Map-Objekt ist und `PublicKeys` welches ein JSON-Array von Zeichenketten ist. Die sprachspezifischen Details finden Sie in den Codebeispielen.

Beispiele für Lambda-Funktionen

In diesem Abschnitt werden einige Lambda-Beispielfunktionen sowohl in NodeJS als auch in Python vorgestellt.

> **Note**
>
> In diesen Beispielen sind der Benutzer, die Rolle, das POSIX-Profil, das Passwort und die Home-Verzeichnisdetails allesamt Beispiele und müssen durch Ihre tatsächlichen Werte ersetzt werden.

### Logical home directory, NodeJS

Die folgende NodeJS-Beispielfunktion stellt die Details für einen Benutzer bereit, der über ein logisches Home-Verzeichnis verfügt.

```
// GetUserConfig Lambda

exports.handler = (event, context, callback) => {
  console.log("Username:", event.username, "ServerId: ", event.serverId);

  var response;
  // Check if the username presented for authentication is correct. This doesn't
 check the value of the server ID, only that it is provided.
  if (event.serverId !== "" && event.username == 'example-user') {
    var homeDirectoryDetails = [
      {
        Entry: "/",
        Target: "/fs-faa1a123"
      }
    ];
    response = {
      Role: 'arn:aws:iam::123456789012:role/transfer-access-role', // The user is
 authenticated if and only if the Role field is not blank
      PosixProfile: {"Gid": 65534, "Uid": 65534}, // Required for EFS access, but
 not needed for S3
      HomeDirectoryDetails: JSON.stringify(homeDirectoryDetails),
      HomeDirectoryType: "LOGICAL",
    };

    // Check if password is provided
    if (!event.password) {
      // If no password provided, return the user's SSH public key
      response['PublicKeys'] = [ "ssh-
rsa abcdef0123456789abcdef0123456789abcdef0123456789abcdef0123456789" ];
    // Check if password is correct
```

```
    } else if (event.password !== 'Password1234') {
      // Return HTTP status 200 but with no role in the response to indicate
 authentication failure
      response = {};
    }
  } else {
    // Return HTTP status 200 but with no role in the response to indicate
 authentication failure
    response = {};
  }
  callback(null, response);
};
```

Path-based home directory, NodeJS

Die folgende NodeJS-Beispielfunktion stellt die Details für einen Benutzer bereit, der über ein pfadbasiertes Home-Verzeichnis verfügt.

```
// GetUserConfig Lambda

exports.handler = (event, context, callback) => {
  console.log("Username:", event.username, "ServerId: ", event.serverId);

  var response;
  // Check if the username presented for authentication is correct. This doesn't
 check the value of the server ID, only that it is provided.
  // There is also event.protocol (one of "FTP", "FTPS", "SFTP") and event.sourceIp
 (e.g., "127.0.0.1") to further restrict logins.
  if (event.serverId !== "" && event.username == 'example-user') {
    response = {
      Role: 'arn:aws:iam::123456789012:role/transfer-access-role', // The user is
 authenticated if and only if the Role field is not blank
      Policy: '', // Optional, JSON stringified blob to further restrict this user's
 permissions
      HomeDirectory: '/fs-faa1a123' // Not required, defaults to '/'
    };

    // Check if password is provided
    if (!event.password) {
      // If no password provided, return the user's SSH public key
     response['PublicKeys'] = [ "ssh-
rsa abcdef0123456789abcdef0123456789abcdef0123456789abcdef0123456789" ];
    // Check if password is correct
```

```
    } else if (event.password !== 'Password1234') {
      // Return HTTP status 200 but with no role in the response to indicate
 authentication failure
      response = {};
    }
  } else {
    // Return HTTP status 200 but with no role in the response to indicate
 authentication failure
    response = {};
  }
  callback(null, response);
};
```

Logical home directory, Python

Die folgende Python-Beispielfunktion stellt die Details für einen Benutzer bereit, der über ein logisches Home-Verzeichnis verfügt.

```
# GetUserConfig Python Lambda with LOGICAL HomeDirectoryDetails
import json

def lambda_handler(event, context):
  print("Username: {}, ServerId: {}".format(event['username'], event['serverId']))

  response = {}

  # Check if the username presented for authentication is correct. This doesn't
 check the value of the server ID, only that it is provided.
  if event['serverId'] != '' and event['username'] == 'example-user':
    homeDirectoryDetails = [
      {
        'Entry': '/',
        'Target': '/fs-faa1a123'
      }
    ]
    response = {
      'Role': 'arn:aws:iam::123456789012:role/transfer-access-role', # The user will
 be authenticated if and only if the Role field is not blank
      'PosixProfile': {"Gid": 65534, "Uid": 65534}, # Required for EFS access, but
 not needed for S3
      'HomeDirectoryDetails': json.dumps(homeDirectoryDetails),
      'HomeDirectoryType': "LOGICAL"
    }
```

```
    # Check if password is provided
    if event.get('password', '') == '':
      # If no password provided, return the user's SSH public key
     response['PublicKeys'] = [ "ssh-
rsa abcdef0123456789abcdef0123456789abcdef0123456789abcdef0123456789" ]
    # Check if password is correct
    elif event['password'] != 'Password1234':
      # Return HTTP status 200 but with no role in the response to indicate
 authentication failure
      response = {}
  else:
    # Return HTTP status 200 but with no role in the response to indicate
 authentication failure
    response = {}

  return response
```

Path-based home directory, Python

Die folgende Python-Beispielfunktion stellt die Details für einen Benutzer bereit, der über ein pfadbasiertes Home-Verzeichnis verfügt.

```
# GetUserConfig Python Lambda with PATH HomeDirectory

def lambda_handler(event, context):
  print("Username: {}, ServerId: {}".format(event['username'], event['serverId']))

  response = {}

  # Check if the username presented for authentication is correct. This doesn't
 check the value of the server ID, only that it is provided.
  # There is also event.protocol (one of "FTP", "FTPS", "SFTP") and event.sourceIp
 (e.g., "127.0.0.1") to further restrict logins.
  if event['serverId'] != '' and event['username'] == 'example-user':
    response = {
      'Role': 'arn:aws:iam::123456789012:role/transfer-access-role', # The user will
 be authenticated if and only if the Role field is not blank
      'Policy': '', #  Optional, JSON stringified blob to further restrict this
 user's permissions
      'HomeDirectory': '/fs-fs-faa1a123',
      'HomeDirectoryType': "PATH" # Not strictly required, defaults to PATH
    }
```

```
    # Check if password is provided
    if event.get('password', '') == '':
      # If no password provided, return the user's SSH public key
     response['PublicKeys'] = [ "ssh-
rsa abcdef0123456789abcdef0123456789abcdef0123456789abcdef0123456789" ]
    # Check if password is correct
    elif event['password'] != 'Password1234':
      # Return HTTP status 200 but with no role in the response to indicate
 authentication failure
      response = {}
  else:
    # Return HTTP status 200 but with no role in the response to indicate
 authentication failure
    response = {}

  return response
```

### Testen Sie Ihre Konfiguration

Nachdem Sie Ihren benutzerdefinierten Identitätsanbieter erstellt haben, sollten Sie Ihre Konfiguration testen.

#### Console

Um Ihre Konfiguration mit der AWS Transfer Family Konsole zu testen

1. Öffnen Sie die AWS Transfer Family -Konsole.
2. Wählen Sie auf der Seite Server Ihren neuen Server aus, klicken Sie auf Aktionen und dann auf Test.
3. Geben Sie den Text für Benutzername und Passwort ein, den Sie bei der Bereitstellung des AWS CloudFormation Stacks festgelegt haben. Wenn Sie die Standardoptionen beibehalten haben, lautet der Benutzername `myuser` und das Passwort`MySuperSecretPassword`.
4. Wählen Sie das Serverprotokoll und geben Sie die IP-Adresse für Quell-IP ein, falls Sie diese bei der Bereitstellung des AWS CloudFormation Stacks festgelegt haben.

CLI

So testen Sie Ihre Konfiguration mit der AWS CLI

1. Führen Sie den Befehl test-identity-provider aus. Ersetzen Sie jeden *user input placeholder* durch Ihre eigenen Informationen, wie in den nachfolgenden Schritten beschrieben.

   ```
   aws transfer test-identity-provider --server-id s-1234abcd5678efgh --user-
   name myuser --user-password MySuperSecretPassword --server-protocol FTP --
   source-ip 127.0.0.1
   ```

2. Geben Sie die Server-ID ein.
3. Geben Sie den Benutzernamen und das Passwort ein, die Sie bei der Bereitstellung des AWS CloudFormation Stacks festgelegt haben. Wenn Sie die Standardoptionen beibehalten haben, lautet der Benutzername `myuser` und das Passwort`MySuperSecretPassword`.
4. Geben Sie das Serverprotokoll und die Quell-IP-Adresse ein, falls Sie sie bei der Bereitstellung des AWS CloudFormation Stacks festgelegt haben.

Wenn die Benutzerauthentifizierung erfolgreich ist, gibt der Test eine `StatusCode: 200` HTTP-Antwort, eine leere Zeichenfolge `Message: ""` (die andernfalls einen Grund für den Fehler enthalten würde) und ein `Response` Feld zurück.

> **Note**
>
> Im Antwortbeispiel unten ist das `Response` Feld ein JSON-Objekt, das „stringifiziert“ wurde (in eine einfache JSON-Zeichenfolge umgewandelt, die innerhalb eines Programms verwendet werden kann) und die Details der Rollen und Berechtigungen des Benutzers enthält.

```
{
    "Response":"{\"Policy\":\"{\\\"Version\\\":\\\"2012-10-17\\\",\\\"Statement\\\":
[{\\\"Sid\\\":\\\"ReadAndListAllBuckets\\\",\\\"Effect\\\":\\\"Allow\\\",\\\"Action\\
\":[\\\"s3:ListAllMybuckets\\\",\\\"s3:GetBucketLocation\\\",\\\"s3:ListBucket\\\",\\
\"s3:GetObjectVersion\\\",\\\"s3:GetObjectVersion\\\"],\\\"Resource\\\":\\\"*\\\"}]}\",
\"Role\":\"arn:aws:iam::000000000000:role/MyUserS3AccessRole\",\"HomeDirectory\":\"/
\"}",
    "StatusCode": 200,
```

```
    "Message": ""
}
```

## Verwenden Sie Amazon API Gateway zur Integration Ihres Identitätsanbieters

In diesem Thema wird beschrieben, wie Sie eine AWS Lambda Funktion verwenden, um eine API-Gateway-Methode zu unterstützen. Verwenden Sie diese Option, wenn Sie eine RESTful-API zur Integration Ihres Identitätsanbieters benötigen oder wenn Sie deren Funktionen für Geoblocking- oder Ratenbegrenzungsanfragen nutzen möchten. AWS WAF

Einschränkungen bei der Verwendung eines API Gateway zur Integration Ihres Identitätsanbieters

- Diese Konfiguration unterstützt keine benutzerdefinierten Domänen.
- Diese Konfiguration unterstützt keine private API-Gateway-URL.

Wenn Sie eines davon benötigen, können Sie Lambda als Identitätsanbieter ohne API Gateway verwenden. Details hierzu finden Sie unter Wird AWS Lambda zur Integration Ihres Identitätsanbieters verwendet.

Authentifizierung mit einer API-Gateway-Methode

Sie können eine API-Gateway-Methode zur Verwendung als Identitätsanbieter für Transfer Family erstellen. Dieser Ansatz bietet Ihnen eine äußerst sichere Möglichkeit, APIs zu erstellen und bereitzustellen. Mit API Gateway können Sie einen HTTPS-Endpunkt erstellen, sodass alle eingehenden API-Aufrufe mit größerer Sicherheit übertragen werden. Weitere Informationen zum API Gateway-Dienst finden Sie im API Gateway Developer Guide.

API Gateway bietet eine Autorisierungsmethode mit dem Namen`AWS_IAM`, die Ihnen dieselbe Authentifizierung auf Basis von AWS Identity and Access Management (IAM) bietet, die auch intern AWS verwendet wird. Wenn Sie die Authentifizierung mit aktivieren`AWS_IAM`, können nur Aufrufer mit ausdrücklichen Berechtigungen zum Aufrufen einer API die API-Gateway-Methode dieser API erreichen.

Um Ihre API Gateway Gateway-Methode als benutzerdefinierten Identitätsanbieter für Transfer Family zu verwenden, aktivieren Sie IAM für Ihre API Gateway Gateway-Methode. Im Rahmen dieses Prozesses stellen Sie eine IAM-Rolle mit Berechtigungen für Transfer Family bereit, Ihr Gateway zu verwenden.

> **Note**
>
> Um die Sicherheit zu verbessern, können Sie eine Firewall für Webanwendungen konfigurieren. AWS WAF ist eine Firewall für Webanwendungen, mit der Sie die HTTP- und HTTPS-Anfragen überwachen können, die an ein Amazon API Gateway weitergeleitet werden. Details hierzu finden Sie unter Fügen Sie eine Firewall für Webanwendungen hinzu.

So verwenden Sie Ihre API Gateway Gateway-Methode für die benutzerdefinierte Authentifizierung mit Transfer Family

1. Erstellen Sie einen AWS CloudFormation Stapel. So gehen Sie vor:

   > **Note**
   >
   > Die Stack-Vorlagen wurden aktualisiert und verwenden nun Base64-kodierte Passwörter: Einzelheiten finden Sie unter. Verbesserungen an den Vorlagen AWS CloudFormation

   a. Öffnen Sie die AWS CloudFormation Konsole unter https://console.aws.amazon.com/cloudformation.

   b. Folgen Sie den Anweisungen zum Bereitstellen eines AWS CloudFormation Stacks anhand einer vorhandenen Vorlage im Abschnitt Auswahl einer Stack-Vorlage im AWS CloudFormation Benutzerhandbuch.

   c. Verwenden Sie eine der folgenden Basisvorlagen, um eine AWS Lambda-gestützte API-Gateway-Methode zur Verwendung als benutzerdefinierter Identitätsanbieter in Transfer Family zu erstellen.

      - Grundlegende Stack-Vorlage

        Standardmäßig wird Ihre API Gateway Gateway-Methode als benutzerdefinierter Identitätsanbieter verwendet, um einen einzelnen Benutzer auf einem einzelnen Server mithilfe eines hartcodierten SSH-Schlüssels (Secure Shell) oder eines Passworts zu authentifizieren. Nach der Bereitstellung können Sie den Lambda-Funktionscode ändern, um etwas anderes zu tun.

      - AWS Secrets Manager Vorlage stapeln

Standardmäßig authentifiziert sich Ihre API Gateway Gateway-Methode anhand eines Eintrags im Secrets Manager des Formats`aws/transfer/`*`server-id`*`/`*`username`*. Darüber hinaus muss das Geheimnis die Schlüssel-Wert-Paare für alle an Transfer Family zurückgegebenen Benutzereigenschaften enthalten. Nach der Bereitstellung können Sie den Lambda-Funktionscode ändern, um etwas anderes zu tun. Weitere Informationen finden Sie im Blogbeitrag Aktivieren der Kennwortauthentifizierung für die AWS Transfer Family Verwendung AWS Secrets Manager.

- Okta-Stack-Vorlage

  Ihre API-Gateway-Methode lässt sich in Okta als benutzerdefinierter Identitätsanbieter in Transfer Family integrieren. Weitere Informationen finden Sie im Blogbeitrag Okta als Identitätsanbieter verwenden mit. AWS Transfer Family

Die Bereitstellung eines dieser Stacks ist der einfachste Weg, einen benutzerdefinierten Identitätsanbieter in den Transfer Family Family-Workflow zu integrieren. Jeder Stack verwendet die Lambda-Funktion, um Ihre API-Methode auf Basis von API Gateway zu unterstützen. Anschließend können Sie Ihre API-Methode als benutzerdefinierten Identitätsanbieter in Transfer Family verwenden. Standardmäßig authentifiziert die Lambda-Funktion einen einzelnen Benutzer, der `myuser` mit dem Passwort aufgerufen wird. `MySuperSecretPassword` Nach der Bereitstellung können Sie diese Anmeldeinformationen bearbeiten oder den Lambda-Funktionscode aktualisieren, um etwas anderes zu tun.

> **Important**
>
> Wir empfehlen, dass Sie die standardmäßigen Benutzer- und Kennwortanmeldedaten bearbeiten.

Nachdem der Stack bereitgestellt wurde, können Sie Details dazu auf der Registerkarte Ausgaben in der CloudFormation Konsole einsehen. Zu diesen Details gehören der Amazon-Ressourcenname (ARN) des Stacks, der ARN der IAM-Rolle, die der Stack erstellt hat, und die URL für Ihr neues Gateway.

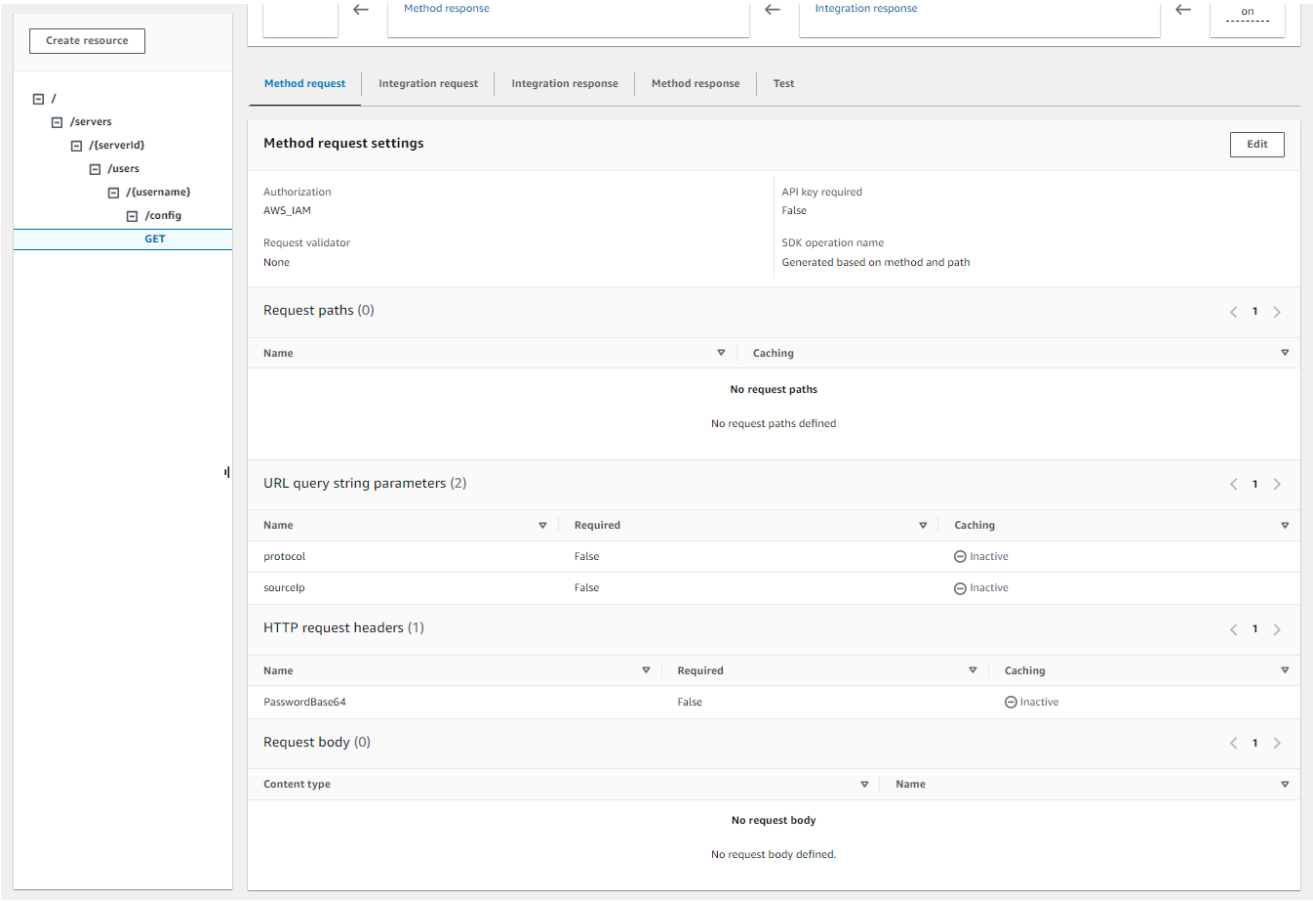
> **Note**
>
> Wenn Sie die Option des benutzerdefinierten Identitätsanbieters verwenden, um die passwortbasierte Authentifizierung für Ihre Benutzer zu aktivieren, und Sie die von API Gateway bereitgestellte Anfrage- und Antwortprotokollierung aktivieren, protokolliert API Gateway die Passwörter Ihrer Benutzer in Ihren Amazon Logs. CloudWatch Wir empfehlen nicht, dieses Protokoll in Ihrer Produktionsumgebung zu verwenden. Weitere Informationen finden Sie unter CloudWatch API-Protokollierung in API Gateway einrichten im API Gateway Developer Guide.

2. Überprüfen Sie die Konfiguration der API Gateway Gateway-Methode für Ihren Server. So gehen Sie vor:

   a. Öffnen Sie die API Gateway-Konsole unter https://console.aws.amazon.com/apigateway/.

   b. Wählen Sie die Basisvorlagen-API für Transfer Custom Identity Provider aus, die von der AWS CloudFormation Vorlage generiert wurde. Möglicherweise müssen Sie Ihre Region auswählen, um Ihre Gateways zu sehen.

   c. Wählen Sie im Bereich Ressourcen die Option GET aus. Der folgende Screenshot zeigt die korrekte Methodenkonfiguration.

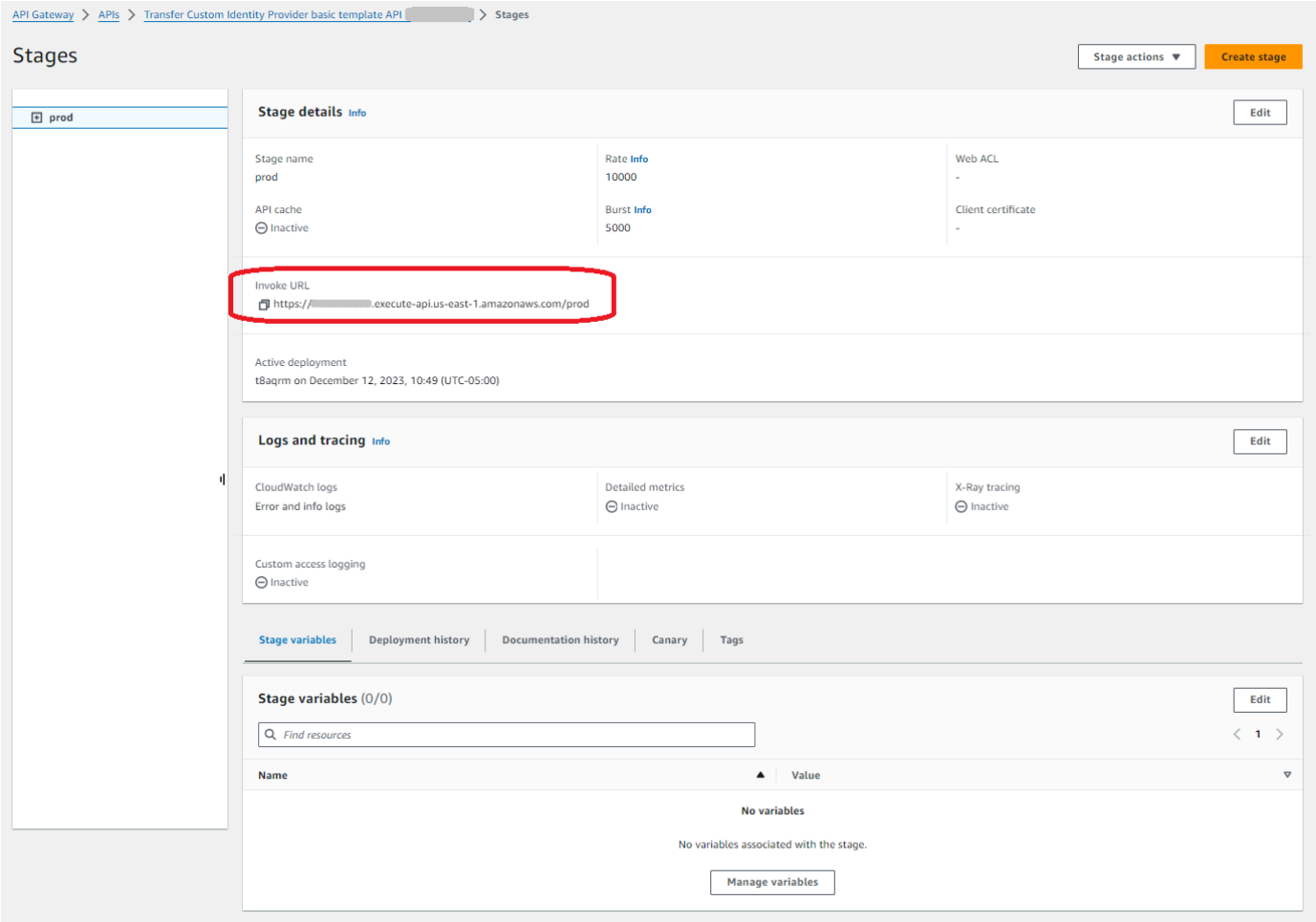



Zu diesem Zeitpunkt ist Ihr API-Gateway bereit für die Bereitstellung.

3. Wählen Sie für Aktionen die Option Deploy API aus. Wählen Sie für die Bereitstellungsphase die Option prod und dann Deploy aus.

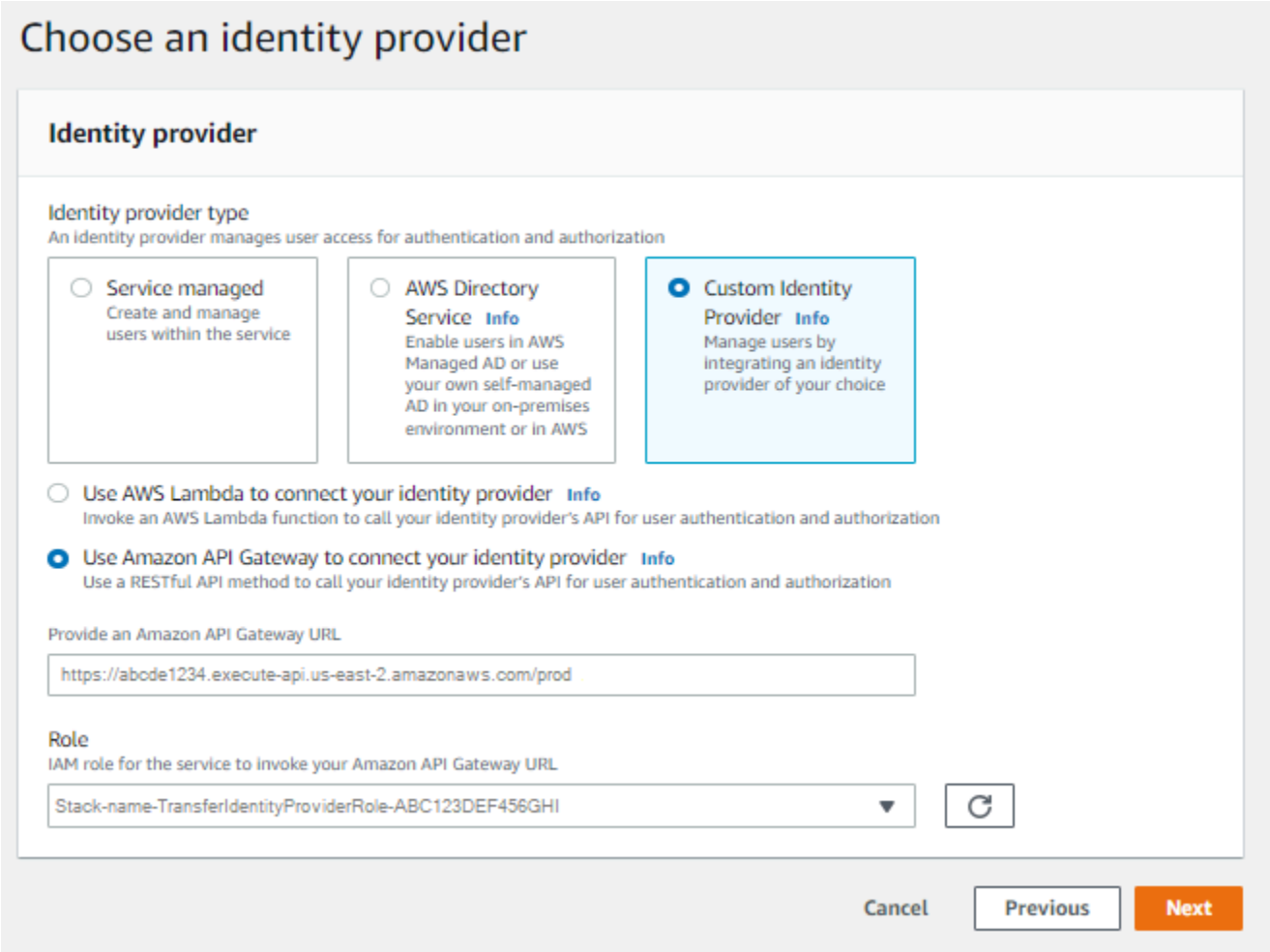
   Nachdem die API Gateway Gateway-Methode erfolgreich bereitgestellt wurde, können Sie sich ihre Leistung unter Stufen > Phasendetails ansehen, wie im folgenden Screenshot gezeigt.

> **Note**
>
> Kopieren Sie die Aufruf-URL-Adresse, die oben auf dem Bildschirm angezeigt wird. Möglicherweise benötigen Sie sie für den nächsten Schritt.




4. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
5. Eine Transfer Family sollte für Sie erstellt worden sein, als Sie den Stack erstellt haben. Wenn nicht, konfigurieren Sie Ihren Server mithilfe dieser Schritte.

   a. Wählen Sie Server erstellen, um die Seite Server erstellen zu öffnen. Wählen Sie für Wählen Sie einen Identitätsanbieter die Option Benutzerdefiniert und dann Amazon API Gateway verwenden, um eine Verbindung zu Ihrem Identitätsanbieter herzustellen, wie im folgenden Screenshot gezeigt.




b. Fügen Sie in das Textfeld Geben Sie eine Amazon API Gateway Gateway-URL ein die Aufruf-URL-Adresse des API Gateway-Endpunkts ein, den Sie in Schritt 3 dieses Verfahrens erstellt haben.

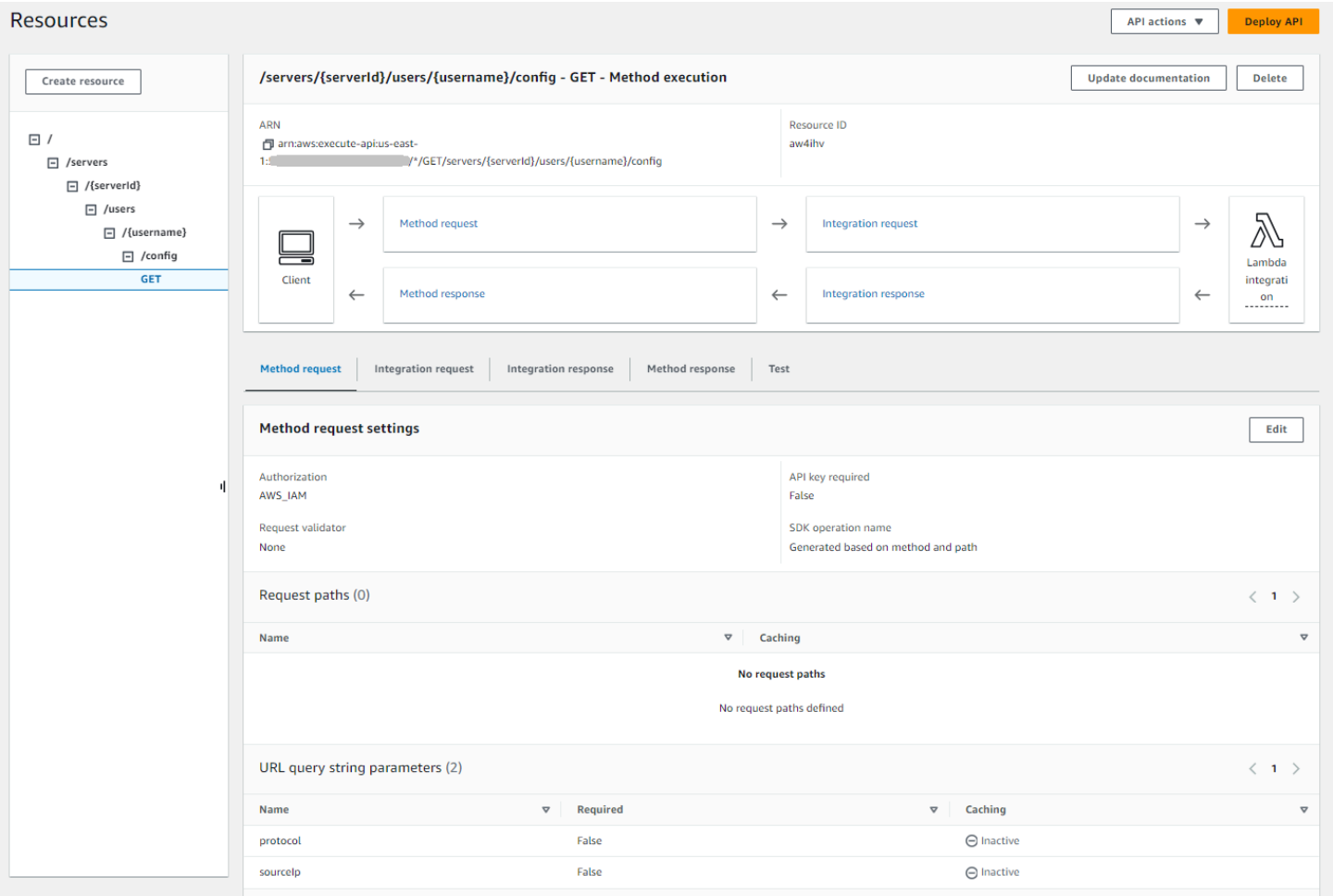
c. Wählen Sie unter Rolle die IAM-Rolle aus, die mit der AWS CloudFormation Vorlage erstellt wurde. Diese Rolle ermöglicht es Transfer Family, Ihre API-Gateway-Methode aufzurufen.

   Die Aufrufrolle enthält den AWS CloudFormation Stack-Namen, den Sie für den Stack ausgewählt haben, den Sie in Schritt 1 erstellt haben. Es hat das folgende Format:*`CloudFormation-stack-name`*`-TransferIdentityProviderRole-`*`ABC123DEF456GHI`*.

d. Füllen Sie die verbleibenden Felder aus und wählen Sie dann Server erstellen. Einzelheiten zu den verbleibenden Schritten zum Erstellen eines Servers finden Sie unterKonfiguration eines SFTP-, FTPS- oder FTP-Serverendpunkts.

## Implementierung Ihrer API-Gateway-Methode

Um einen benutzerdefinierten Identitätsanbieter für Transfer Family zu erstellen, muss Ihre API-Gateway-Methode eine einzelne Methode implementieren, die einen Ressourcenpfad von hat/`servers/`*`serverId`*`/users/`*`username`*`/config`. Die *`username`* Werte *`serverId`* und stammen aus dem RESTful-Ressourcenpfad. Fügen Sie außerdem `sourceIp` und `protocol` als URL-Abfragezeichenfolgenparameter zur Methodenanforderung hinzu, wie in der folgenden Abbildung gezeigt.




> **Note**
>
> Der Benutzername muss mindestens 3 und maximal 100 Zeichen lang sein. Sie können die folgenden Zeichen im Benutzernamen verwenden: a—z, A-Z, 0—9, Unterstrich (_), Bindestrich (-), Punkt (.) und At-Zeichen (@). Der Benutzername darf jedoch nicht mit einem Bindestrich (-), einem Punkt (.) oder einem Zeichen (@) beginnen.

Wenn Transfer Family versucht, Ihren Benutzer mit einem Passwort zu authentifizieren, stellt der Dienst ein `Password:` Header-Feld bereit. In Ermangelung eines `Password:` Headers versucht Transfer Family, Ihren Benutzer mit einer Authentifizierung mit öffentlichem Schlüssel zu authentifizieren.

Wenn Sie einen Identitätsanbieter zur Authentifizierung und Autorisierung von Endbenutzern verwenden, können Sie neben der Überprüfung ihrer Anmeldeinformationen auch Zugriffsanfragen auf der Grundlage der IP-Adressen der von Ihren Endbenutzern verwendeten Clients zulassen oder ablehnen. Mit dieser Funktion können Sie sicherstellen, dass auf Daten, die in Ihren S3-Buckets oder Ihrem Amazon EFS-Dateisystem gespeichert sind, über die unterstützten Protokolle nur von IP-Adressen aus zugegriffen werden kann, die Sie als vertrauenswürdig angegeben haben. Um diese Funktion zu aktivieren, müssen Sie sie `sourceIp` in der Abfragezeichenfolge angeben.

Wenn Sie mehrere Protokolle für Ihren Server aktiviert haben und den Zugriff mit demselben Benutzernamen über mehrere Protokolle ermöglichen möchten, können Sie dies tun, sofern die für jedes Protokoll spezifischen Anmeldeinformationen in Ihrem Identitätsanbieter eingerichtet wurden. Um diese Funktion zu aktivieren, müssen Sie den *`protocol`* Wert in den RESTful-Ressourcenpfad aufnehmen.

Ihre API Gateway Gateway-Methode sollte immer den HTTP-Statuscode zurückgeben`200`. Jeder andere HTTP-Statuscode bedeutet, dass beim Zugriff auf die API ein Fehler aufgetreten ist.

Amazon S3 S3-Beispielantwort

Der Beispielantworttext ist ein JSON-Dokument der folgenden Form für Amazon S3.

```
{
 "Role": "IAM role with configured S3 permissions",
 "PublicKeys": [
     "ssh-rsa public-key1",
     "ssh-rsa public-key2"
  ],
 "Policy": "STS Assume role session policy",
 "HomeDirectory": "/bucketName/path/to/home/directory"
}
```

> **Note**
>
> Der Richtlinie wird JSON als Zeichenfolge maskiert. Beispielsweise:
>
> ```
> "Policy":
> ```

```
"{
  \"Version\": \"2012-10-17\",
  \"Statement\":
     [
     {\"Condition\":
        {\"StringLike\":
            {\"s3:prefix\":
               [\"user/*\", \"user/\"]}},
     \"Resource\": \"arn:aws:s3:::bucket\",
     \"Action\": \"s3:ListBucket\",
     \"Effect\": \"Allow\",
     \"Sid\": \"ListHomeDir\"},
     {\"Resource\": \"arn:aws:s3:::*\",
        \"Action\": [\"s3:PutObject\",
        \"s3:GetObject\",
        \"s3:DeleteObjectVersion\",
        \"s3:DeleteObject\",
        \"s3:GetObjectVersion\",
        \"s3:GetObjectACL\",
        \"s3:PutObjectACL\"],
     \"Effect\": \"Allow\",
     \"Sid\": \"HomeDirObjectAccess\"}]
}"
```

Die folgende Beispielantwort zeigt, dass ein Benutzer einen logischen Home-Verzeichnistyp hat.

```
{
   "Role": "arn:aws:iam::123456789012:role/transfer-access-role-s3",
   "HomeDirectoryType":"LOGICAL",
   "HomeDirectoryDetails":"[{\"Entry\":\"/\",\"Target\":\"/MY-HOME-BUCKET\"}]",
   "PublicKeys":[""]
}
```

Amazon EFS-Beispielantwort

Der Beispielantworttext ist ein JSON-Dokument der folgenden Form für Amazon EFS.

```
{
 "Role": "IAM role with configured EFS permissions",
 "PublicKeys": [
     "ssh-rsa public-key1",
```

```
      "ssh-rsa public-key2"
   ],
  "PosixProfile": {
    "Uid": "POSIX user ID",
    "Gid": "POSIX group ID",
    "SecondaryGids": [Optional list of secondary Group IDs],
  },
  "HomeDirectory": "/fs-id/path/to/home/directory"
}
```

Das `Role` Feld zeigt, dass eine erfolgreiche Authentifizierung stattgefunden hat. Bei der Passwortauthentifizierung (wenn Sie einen `Password:` Header angeben), müssen Sie keine öffentlichen SSH-Schlüssel angeben. Wenn ein Benutzer nicht authentifiziert werden kann, z. B. wenn das Passwort falsch ist, sollte Ihre Methode eine Antwort zurückgeben, die nicht gesetzt ist. `Role` Ein Beispiel für eine solche Antwort ist ein leeres JSON-Objekt.

Die folgende Beispielantwort zeigt einen Benutzer mit einem logischen Home-Verzeichnistyp.

```
{
    "Role": "arn:aws:iam::123456789012:role/transfer-access-role-efs",
    "HomeDirectoryType": "LOGICAL",
    "HomeDirectoryDetails":"[{\"Entry\":\"/\",\"Target\":\"/faa1a123\"}]",
    "PublicKeys":[""],
    "PosixProfile":{"Uid":65534,"Gid":65534}
}
```

Sie können Benutzerrichtlinien in die Lambda-Funktion im JSON-Format aufnehmen. Weitere Informationen zur Konfiguration von Benutzerrichtlinien in Transfer Family finden Sie unterVerwaltung der Zugriffskontrollen.

Standard-Lambda-Funktion

Um verschiedene Authentifizierungsstrategien zu implementieren, bearbeiten Sie die Lambda-Funktion, die Ihr Gateway verwendet. Um die Anforderungen Ihrer Anwendung zu erfüllen, können Sie die folgenden Lambda-Beispielfunktionen in Node.js verwenden. Weitere Informationen zu Lambda finden Sie im AWS Lambda Developer Guide oder im Building Lambda functions with Node.js.

Die folgende Lambda-Beispielfunktion verwendet Ihren Benutzernamen, Ihr Passwort (wenn Sie eine Passwortauthentifizierung durchführen), Ihre Server-ID, Ihr Protokoll und Ihre Client-IP-Adresse. Sie

können eine Kombination dieser Eingaben verwenden, um nach Ihrem Identitätsanbieter zu suchen und festzustellen, ob die Anmeldung akzeptiert werden soll.

> **Note**
>
> Wenn Sie mehrere Protokolle für Ihren Server aktiviert haben und den Zugriff mit demselben Benutzernamen über mehrere Protokolle ermöglichen möchten, können Sie dies tun, sofern die für das Protokoll spezifischen Anmeldeinformationen in Ihrem Identitätsanbieter eingerichtet wurden.
> Für das File Transfer Protocol (FTP) empfehlen wir, separate Anmeldeinformationen für Secure Shell (SSH) File Transfer Protocol (SFTP) und File Transfer Protocol over SSL (FTPS) zu verwenden. Wir empfehlen, separate Anmeldeinformationen für FTP zu verwenden, da FTP im Gegensatz zu SFTP und FTPS Anmeldeinformationen im Klartext überträgt. Wenn FTP-Anmeldeinformationen von SFTP oder FTPS isoliert werden, bleiben Ihre Workloads, die SFTP oder FTPS verwenden, sicher, wenn FTP-Anmeldeinformationen geteilt oder offengelegt werden.

Diese Beispielfunktion gibt die Rollen- und logischen Basisverzeichnisdetails zusammen mit den öffentlichen Schlüsseln zurück (sofern sie die Authentifizierung mit öffentlichen Schlüsseln durchführt).

Wenn Sie vom Service verwaltete Benutzer erstellen, legen Sie deren Basisverzeichnis fest, entweder logisch oder physisch. In ähnlicher Weise benötigen wir die Ergebnisse der Lambda-Funktion, um die gewünschte physische oder logische Verzeichnisstruktur des Benutzers zu vermitteln. Die von Ihnen festgelegten Parameter hängen vom Wert für das HomeDirectoryTypeFeld ab.

- `HomeDirectoryType`gesetzt auf `PATH` — das `HomeDirectory` Feld muss dann ein absolutes Amazon S3 S3-Bucket-Präfix oder ein absoluter Amazon EFS-Pfad sein, der für Ihre Benutzer sichtbar ist.
- `HomeDirectoryType`gesetzt auf `LOGICAL` — Legen Sie kein `HomeDirectory` Feld fest. Stattdessen legen wir ein `HomeDirectoryDetails` Feld fest, das die gewünschten Eingabe-/Zielzuordnungen bereitstellt, ähnlich den im HomeDirectoryDetailsParameter beschriebenen Werten für vom Service verwaltete Benutzer.

Die Beispielfunktionen sind unter aufgeführt. Beispiele für Lambda-Funktionen

Lambda-Funktion zur Verwendung mit AWS Secrets Manager

Um sie AWS Secrets Manager als Identitätsanbieter zu verwenden, können Sie mit der Lambda-Funktion in der AWS CloudFormation Beispielvorlage arbeiten. Die Lambda-Funktion fragt den Secrets Manager Manager-Dienst mit Ihren Anmeldeinformationen ab und gibt bei Erfolg ein bestimmtes Geheimnis zurück. Weitere Informationen zu Secrets Manager finden Sie im Benutzerhandbuch für AWS Secrets Manager.

Um eine AWS CloudFormation Beispielvorlage herunterzuladen, die diese Lambda-Funktion verwendet, rufen Sie den Amazon S3 S3-Bucket auf, der von bereitgestellt wird AWS Transfer Family.

Verbesserungen an den Vorlagen AWS CloudFormation

An den veröffentlichten CloudFormation Vorlagen wurden Verbesserungen an der API Gateway Gateway-Schnittstelle vorgenommen. Die Vorlagen verwenden jetzt Base64-kodierte Passwörter mit dem API Gateway. Ihre bestehenden Bereitstellungen funktionieren auch ohne diese Erweiterung, lassen jedoch keine Passwörter zu, deren Zeichen außerhalb des grundlegenden US-ASCII-Zeichensatzes liegen.

Die Änderungen in der Vorlage, die diese Funktion ermöglichen, lauten wie folgt:

- Die `GetUserConfigRequest AWS::ApiGateway::Method` Ressource muss diesen `RequestTemplates` Code haben (die kursiv gedruckte Zeile ist die aktualisierte Zeile)

```
RequestTemplates:
   application/json: |
   {
      "username": "$util.urlDecode($input.params('username'))",
      "password":
 "$util.escapeJavaScript($util.base64Decode($input.params('PasswordBase64'))).replaceAll("\
\'","'")",
      "protocol": "$input.params('protocol')",
      "serverId": "$input.params('serverId')",
      "sourceIp": "$input.params('sourceIp')"
}
```

- Der Wert `RequestParameters` für die `GetUserConfig` Ressource muss geändert werden, damit der `PasswordBase64` Header verwendet werden kann (die kursiv gedruckte Zeile ist die aktualisierte Zeile):

```
RequestParameters:
   method.request.header.PasswordBase64: false
   method.request.querystring.protocol: false
   method.request.querystring.sourceIp: false
```

Um zu überprüfen, ob die Vorlage für Ihren Stack die neueste ist

1. Öffnen Sie die AWS CloudFormation Konsole unter https://console.aws.amazon.com/cloudformation.
2. Wählen Sie aus der Liste der Stacks Ihren Stack aus.
3. Wählen Sie im Detailbereich die Registerkarte Vorlage aus.
4. Achten Sie auf Folgendes:

   - Suchen Sie nach dieser Zeile und stellen Sie sicher`RequestTemplates`, dass Sie sie haben:

     ```
     "password":
      "$util.escapeJavaScript($util.base64Decode($input.params('PasswordBase64'))).replaceAll(
     \'","'")",
     ```

   - Suchen Sie nach dieser Zeile und stellen Sie sicher`RequestParameters`, dass Sie sie haben:

     ```
     method.request.header.PasswordBase64: false
     ```

Wenn Sie die aktualisierten Zeilen nicht sehen, bearbeiten Sie Ihren Stapel. Einzelheiten zum Aktualisieren Ihres AWS CloudFormation Stacks finden Sie unter Ändern einer Stack-Vorlage im AWS CloudFormation; Benutzerhandbuch.

## Verwendung logischer Verzeichnisse zur Vereinfachung Ihrer Transfer Family Family-Verzeichnisstrukturen

Um Ihre AWS Transfer Family Serververzeichnisstruktur zu vereinfachen, können Sie logische Verzeichnisse verwenden. Mit logischen Verzeichnissen können Sie eine virtuelle Verzeichnisstruktur erstellen, die benutzerfreundliche Namen verwendet, unter denen Ihre Benutzer navigieren, wenn sie sich mit Ihrem Amazon S3 S3-Bucket oder Amazon EFS-Dateisystem verbinden. Wenn Sie

logische Verzeichnisse verwenden, können Sie vermeiden, absolute Verzeichnispfade, Amazon S3 S3-Bucket-Namen und EFS-Dateisystemnamen an Ihre Endbenutzer weiterzugeben.

> **Note**
>
> Sie sollten Sitzungsrichtlinien verwenden, damit Ihre Endbenutzer nur Vorgänge ausführen können, deren Ausführung Sie ihnen gestatten.
> Sie sollten logische Verzeichnisse verwenden, um ein benutzerfreundliches, virtuelles Verzeichnis für Ihre Endbenutzer zu erstellen und Bucket-Namen zu abstrahieren. Logische Verzeichniszuordnungen ermöglichen Benutzern nur den Zugriff auf die ihnen zugewiesenen logischen Pfade und Unterverzeichnisse und verbieten relative Pfade, die die logischen Wurzeln durchqueren.
> Transfer Family validiert jeden Pfad, der relative Elemente enthalten könnte, und blockiert aktiv die Auflösung dieser Pfade, bevor wir diese Pfade an Amazon S3 übergeben. Dadurch wird verhindert, dass Ihre Benutzer ihre logischen Zuordnungen verlassen.
> Transfer Family verhindert zwar, dass Ihre Endbenutzer auf Verzeichnisse außerhalb ihres logischen Verzeichnisses zugreifen, wir empfehlen Ihnen jedoch, auch eindeutige Rollen oder Sitzungsrichtlinien zu verwenden, um die geringsten Rechte auf Speicherebene durchzusetzen.

Sie können logische Verzeichnisse verwenden, um das Stammverzeichnis des Benutzers an einem gewünschten Ort innerhalb Ihrer Speicherhierarchie festzulegen, indem Sie eine sogenannte chroot Operation ausführen. In diesem Modus können Benutzer nicht zu einem Verzeichnis außerhalb des Home- oder Stammverzeichnisses navigieren, das Sie für sie konfiguriert haben.

Obwohl beispielsweise ein Amazon S3-Benutzer nur auf den Zugriff beschränkt wurde, ermöglichen es einige Clients Benutzern`/mybucket/home/${transfer:UserName}`, einen Ordner nach oben zu durchsuchen. `/mybucket/home` In dieser Situation landet der Benutzer erst wieder in seinem vorgesehenen Home-Verzeichnis, nachdem er sich vom Transfer Family Family-Server ab- und wieder angemeldet hat. Durch das Ausführen eines chroot Vorgangs kann verhindert werden, dass diese Situation eintritt.

Sie können Ihre eigene Verzeichnisstruktur mit Buckets und Präfixen erstellen. Diese Funktion ist nützlich, wenn Sie einen Workflow haben, der eine bestimmte Verzeichnisstruktur erwartet, die Sie nicht über Bucket-Präfixe replizieren können. Sie können auch Links zu mehreren nicht zusammenhängenden Speicherorten innerhalb von Amazon S3 erstellen, ähnlich wie beim Erstellen

eines symbolischen Links in einem Linux-Dateisystem, in dem Ihr Verzeichnispfad auf einen anderen Speicherort im Dateisystem verweist.

> **Logische Verzeichnis-FILE-Zuordnungen**
>
> Der `HomeDirectoryMapEntry` Datentyp enthält jetzt einen `Type` Parameter. Bevor dieser Parameter existierte, hätten Sie eine logische Verzeichniszuordnung erstellen können, bei der das Ziel eine Datei war. Wenn Sie zuvor eine dieser Arten von logischen Verzeichniszuordnungen erstellt haben, müssen Sie den Wert explizit `Type` auf festlegen`FILE`, da diese Zuordnungen sonst in Zukunft nicht mehr richtig funktionieren. Eine Möglichkeit, dies zu tun, besteht darin, die `UpdateUser` API aufzurufen und den Wert `Type` auf `FILE` für die vorhandene Zuordnung festzulegen.

## Regeln für die Verwendung logischer Verzeichnisse

Bevor Sie Ihre logischen Verzeichniszuordnungen erstellen, sollten Sie sich mit den folgenden Regeln vertraut machen:

- Wenn dies `Entry` der Fall ist`"/"`, können Sie nur eine Zuordnung verwenden, da überlappende Pfade nicht zulässig sind.
- Logische Verzeichnisse unterstützen Zuordnungen von bis zu 2,1 MB (für Benutzer, die vom Service verwaltet werden, liegt dieser Grenzwert bei 2.000 Einträgen). Das heißt, die Datenstruktur, die die Zuordnungen enthält, hat eine maximale Größe von 2,1 MB. Wenn Sie über viele Mappings verfügen, können Sie die Größe Ihrer Mappings wie folgt berechnen:
  1. Schreiben Sie ein typisches Mapping in dem Format aus`{"Entry":"/`*`entry-path`*`","Target":"/`*`target-path`*`"}`, wo *`entry-path`* und *`target-path`* sind die tatsächlichen Werte, die Sie verwenden werden.
  2. Zählen Sie die Zeichen in dieser Zeichenfolge und fügen Sie dann eins hinzu (1).
  3. Multiplizieren Sie diese Zahl mit der ungefähren Anzahl von Zuordnungen, die Sie für Ihren Server haben.

  Wenn die Zahl, die Sie in Schritt 3 geschätzt haben, weniger als 2,1 MB beträgt, liegen Ihre Zuordnungen innerhalb des akzeptablen Grenzwerts.
- Ziele können die `${transfer:UserName}` Variable verwenden, wenn der Bucket- oder Dateisystempfad anhand des Benutzernamens parametrisiert wurde.

- Ziele können Pfade in verschiedenen Buckets oder Dateisystemen sein. Sie müssen jedoch sicherstellen, dass die zugeordnete Rolle AWS Identity and Access Management (IAM) (der `Role` Parameter in der Antwort) Zugriff auf diese Buckets oder Dateisysteme bietet.
- Geben Sie den `HomeDirectory` Parameter nicht an, da dieser Wert durch die `Entry` `Target` Paare impliziert wird, wenn Sie den Wert für den `LOGICAL` Parameter verwenden. `HomeDirectoryType`
- Ziele müssen mit einem Schrägstrich (`/`) beginnen, verwenden aber keine abschließenden Schrägstriche (`/`), wenn Sie den angeben. `Target` Ist zum Beispiel akzeptabel, `/DOC-EXAMPLE-BUCKET/images` ist es aber auch nicht`DOC-EXAMPLE-BUCKET/images`. `/DOC-EXAMPLE-BUCKET/images/`
- Amazon S3 ist ein Objektspeicher, was bedeutet, dass Ordner ein virtuelles Konzept sind und es keine tatsächliche Verzeichnishierarchie gibt. Wenn Ihre Anwendung einen `stat` Vorgang von einem Client ausgibt, wird alles als Datei klassifiziert, wenn Sie Amazon S3 als Speicher verwenden. Dieses Verhalten wird unter Organisieren von Objekten in der Amazon S3 S3-Konsole mithilfe von Ordnern im Amazon Simple Storage Service-Benutzerhandbuch beschrieben. Wenn in Ihrer Anwendung `stat` genau angegeben werden muss, ob es sich um eine Datei oder einen Ordner handelt, können Sie Amazon Elastic File System (Amazon EFS) als Speicheroption für Ihre Transfer Family Family-Server verwenden.
- Wenn Sie logische Verzeichniswerte für Ihren Benutzer angeben, hängt der verwendete Parameter vom Benutzertyp ab:
  - Geben Sie für vom Dienst verwaltete Benutzer logische Verzeichniswerte in `HomeDirectoryMappings` ein.
  - Geben Sie für Benutzer eines benutzerdefinierten Identitätsanbieters logische Verzeichniswerte in an`HomeDirectoryDetails`.

> **Important**
>
> Sofern Sie sich nicht dafür entscheiden, die Leistung Ihrer Amazon S3 S3-Verzeichnisse zu optimieren (wenn Sie einen Server erstellen oder aktualisieren), muss das Stammverzeichnis beim Start vorhanden sein. Für Amazon S3 bedeutet dies, dass Sie bereits ein Null-Byte-Objekt erstellt haben müssen, das mit einem Schrägstrich (`/`) endet, um den Stammordner zu erstellen. Die Vermeidung dieses Problems ist ein Grund, eine Optimierung der Amazon S3 S3-Leistung in Betracht zu ziehen.

# Implementierung logischer Verzeichnisse und **chroot**

Um logische Verzeichnisse und chroot Funktionen zu verwenden, müssen Sie wie folgt vorgehen:

Aktivieren Sie logische Verzeichnisse für jeden Benutzer. Setzen Sie dazu den `HomeDirectoryType` Parameter auf, `LOGICAL` wenn Sie Ihren Benutzer erstellen oder aktualisieren.

```
"HomeDirectoryType": "LOGICAL"
```

## **chroot**

Erstellen Sie für eine Verzeichnisstrukturchroot, die aus einer einzelnen Verzeichnisstruktur `Entry` und einer `Target` Paarung für jeden Benutzer besteht. Der Stammordner ist der `Entry` Punkt und der `Target` ist der Speicherort in Ihrem Bucket oder Dateisystem, dem die Zuordnung zugewiesen werden soll.

Example for Amazon S3

```
[{"Entry": "/", "Target": "/mybucket/jane"}]
```

Example for Amazon EFS

```
[{"Entry": "/", "Target": "/fs-faa1a123/jane"}]
```

Sie können einen absoluten Pfad wie im vorherigen Beispiel verwenden, oder Sie können eine dynamische Ersetzung für den Benutzernamen durch verwenden`${transfer:UserName}`, wie im folgenden Beispiel.

```
[{"Entry": "/", "Target":
"/mybucket/${transfer:UserName}"}]
```

Im vorherigen Beispiel ist der Benutzer an sein Stammverzeichnis gebunden und kann sich in der Hierarchie nicht weiter oben bewegen.

### Struktur des virtuellen Verzeichnisses

Für eine virtuelle Verzeichnisstruktur können Sie mehrere `Entry` `Target` Paarungen mit Zielen an einer beliebigen Stelle in Ihren S3-Buckets oder EFS-Dateisystemen erstellen, auch über

mehrere Buckets oder Dateisysteme hinweg, sofern die IAM-Rollenzuordnung des Benutzers über Zugriffsberechtigungen verfügt.

Im folgenden Beispiel für eine virtuelle Struktur befindet sich der Benutzer, wenn er sich bei AWS SFTP anmeldet, im Stammverzeichnis mit den Unterverzeichnissen,, und. `/pics` `/doc` `/reporting` `/anotherpath/subpath/financials`

> **Note**
>
> Sofern Sie sich nicht dafür entscheiden, die Leistung Ihrer Amazon S3 S3-Verzeichnisse zu optimieren (wenn Sie einen Server erstellen oder aktualisieren), muss entweder der Benutzer oder ein Administrator die Verzeichnisse erstellen, sofern sie noch nicht existieren. Die Vermeidung dieses Problems ist ein Grund, eine Optimierung der Amazon S3 S3-Leistung in Betracht zu ziehen.
> Für Amazon EFS benötigen Sie weiterhin den Administrator, um die logischen Zuordnungen oder das / Verzeichnis zu erstellen.

```
[
{"Entry": "/pics", "Target": "/bucket1/pics"},
{"Entry": "/doc", "Target": "/bucket1/anotherpath/docs"},
{"Entry": "/reporting", "Target": "/reportingbucket/Q1"},
{"Entry": "/anotherpath/subpath/financials", "Target": "/reportingbucket/financials"}]
```

> **Note**
>
> Sie können Dateien nur in die spezifischen Ordner hochladen, die Sie zuordnen. Das bedeutet, dass Sie im vorherigen Beispiel nicht in `/anotherpath` oder `anotherpath/subpath` Verzeichnisse hochladen können, sondern nur`anotherpath/subpath/financials`. Sie können diesen Pfaden auch nicht direkt zuordnen, da überlappende Pfade nicht zulässig sind.
> Gehen Sie beispielsweise davon aus, dass Sie die folgenden Zuordnungen erstellen:
>
> ```
> {
>    "Entry": "/pics",
>    "Target": "/mybucket/pics"
> },
> {
> ```

```
   "Entry": "/doc",
   "Target": "/mybucket/mydocs"
},
{
   "Entry": "/temp",
   "Target": "/mybucket"
}
```

Sie können nur Dateien in diese Buckets hochladen. Wenn Sie zum ersten Mal eine Verbindung herstellen`sftp`, werden Sie im Stammverzeichnis abgelegt. `/` Wenn Sie versuchen, eine Datei in dieses Verzeichnis hochzuladen, schlägt der Upload fehl. Die folgenden Befehle zeigen eine Beispielsequenz:

```
sftp> pwd
Remote working directory: /
sftp> put file
Uploading file to /file
remote open("/file"): No such file or directory
```

Um in eine beliebige Datei hochzuladen`directory/sub-directory`, müssen Sie den Pfad explizit dem zuordnen`sub-directory`.

Weitere Informationen zur Konfiguration logischer Verzeichnisse und chroot für Ihre Benutzer, einschließlich einer AWS CloudFormation Vorlage, die Sie herunterladen und verwenden können, finden Sie unter Vereinfachen Sie Ihre AWS SFTP-Struktur mit Chroot und logischen Verzeichnissen im AWS Storage-Blog.

## Beispiel für die Konfiguration logischer Verzeichnisse

In diesem Beispiel erstellen wir einen Benutzer und weisen ihm zwei logische Verzeichnisse zu. Der folgende Befehl erstellt einen neuen Benutzer (für einen vorhandenen Transfer Family Family-Server) mit logischen Verzeichnissen `pics` und`doc`.

```
aws transfer create-user --user-name marymajor-logical --server-id s-11112222333344445
 --role arn:aws:iam::1234abcd5678:role/marymajor-role --home-directory-type LOGICAL \
 --home-directory-mappings "[{\"Entry\":\"/pics\", \"Target\":\"/DOC-EXAMPLE-BUCKET1/
pics\"}, {\"Entry\":\"/doc\", \"Target\":\"/DOC-EXAMPLE-BUCKET2/test/mydocs\"}]" \
 --ssh-public-key-body file://~/.ssh/id_rsa.pub
```

Wenn **marymajor** es sich um einen bestehenden Benutzer handelt und sein Home-Verzeichnistyp istPATH, können Sie ihn LOGICAL mit einem ähnlichen Befehl wie dem vorherigen ändern.

```
aws transfer update-user --user-name marymajor-logical \
 --server-id s-11112222333344445 --role arn:aws:iam::1234abcd5678:role/marymajor-role \
 --home-directory-type LOGICAL --home-directory-mappings "[{\"Entry\":\"/pics\",
 \"Target\":\"/DOC-EXAMPLE-BUCKET1/pics\"}, \
 {\"Entry\":\"/doc\", \"Target\":\"/DOC-EXAMPLE-BUCKET2/test/mydocs\"}]"
```

Beachten Sie Folgendes:

- Wenn die Verzeichnisse /DOC-EXAMPLE-BUCKET1/pics und noch /DOC-EXAMPLE-BUCKET2/test/mydocs nicht existieren, muss der Benutzer (oder ein Administrator) sie erstellen.
- Wenn sie **marymajor** eine Verbindung zum Server herstellt und den ls -l Befehl ausführt, sieht sie Folgendes:

  ```
  drwxr--r--   1        -        -        0 Mar 17 15:42 doc
  drwxr--r--   1        -        -        0 Mar 17 16:04 pics
  ```

- **marymajor**kann auf dieser Ebene keine Dateien oder Verzeichnisse erstellen. Innerhalb von pics und kann sie doc jedoch Unterverzeichnisse hinzufügen.
- Dateien, die sie zu pics Amazon S3 S3-Pfaden hinzufügt /DOC-EXAMPLE-BUCKET1/pics / DOC-EXAMPLE-BUCKET2/test/mydocs bzw. zu diesen hinzugefügt doc werden.
- In diesem Beispiel geben wir zwei verschiedene Buckets an, um diese Möglichkeit zu veranschaulichen. Sie können jedoch denselben Bucket für mehrere oder alle logischen Verzeichnisse verwenden, die Sie für den Benutzer angeben.

## Logische Verzeichnisse für Amazon EFS konfigurieren

Wenn Ihr Transfer Family Family-Server Amazon EFS verwendet, muss das Home-Verzeichnis für den Benutzer mit Lese- und Schreibzugriff erstellt werden, bevor der Benutzer in seinem logischen Home-Verzeichnis arbeiten kann. Der Benutzer kann dieses Verzeichnis nicht selbst erstellen, da ihm die entsprechenden Berechtigungen für mkdir sein logisches Home-Verzeichnis fehlen würden.

Wenn das Basisverzeichnis des Benutzers nicht existiert und er einen ls Befehl ausführt, reagiert das System wie folgt:

```
sftp> ls
```

```
remote readdir ("/"): No such file or directory
```

Ein Benutzer mit Administratorzugriff auf das übergeordnete Verzeichnis muss das logische Home-Verzeichnis des Benutzers erstellen.

## Benutzerdefinierte AWS Lambda Antwort

Sie können logische Verzeichnisse mit einer Lambda-Funktion verwenden, die eine Verbindung zu Ihrem benutzerdefinierten Identitätsanbieter herstellt. Dazu geben Sie in Ihrer Lambda-Funktion die `Target` Werte `HomeDirectoryType` as **`LOGICAL`** und add `Entry` und für den `HomeDirectoryDetails` Parameter an. Beispielsweise:

```
HomeDirectoryType: "LOGICAL"
HomeDirectoryDetails: "[{\"Entry\": \"/\", \"Target\": \"/DOC-EXAMPLE-BUCKET/
theRealFolder"}]"
```

Der folgende Code ist ein Beispiel für eine erfolgreiche Antwort auf einen benutzerdefinierten Lambda-Authentifizierungsaufruf.

```
aws transfer test-identity-provider --server-id s-1234567890abcdef0 --user-name myuser
{
    "Url": "https://a1b2c3d4e5.execute-api.us-east-2.amazonaws.com/prod/servers/
s-1234567890abcdef0/users/myuser/config",
    "Message": "",
    "Response": "{\"Role\": \"arn:aws:iam::123456789012:role/bob-usa-role\",
\"HomeDirectoryType\": \"LOGICAL\",\"HomeDirectoryDetails\": \"[{\\\"Entry\\\":\\\"/
myhome\\\",\\\"Target\\\":\\\"/DOC-EXAMPLE-BUCKET/theRealFolder\\\"}]\",\"PublicKeys\":
 \"[ssh-rsa myrsapubkey]\"}",
    "StatusCode": 200
}
```

> **Note**
>
> Die `"Url":` Zeile wird nur zurückgegeben, wenn Sie eine API Gateway Gateway-Methode als Ihren benutzerdefinierten Identitätsanbieter verwenden.

# AWS Transfer Family SFTP-Anschlüsse

AWS Transfer Family SFTP-Konnektoren stellen mithilfe des SFTP-Protokolls eine Beziehung für das Senden von Dateien und Nachrichten zwischen Amazon-Speicher und einem externen Partner her. Sie können Dateien von Amazon S3 an ein externes, partnereigenes Ziel senden. Sie können auch einen SFTP-Connector verwenden, um Dateien vom SFTP-Server eines Partners abzurufen.

> **Note**
>
> Derzeit können SFTP-Konnektoren nur verwendet werden, um eine Verbindung zu Remote-SFTP-Servern herzustellen, die einen über das Internet zugänglichen Endpunkt bieten.

Die folgenden Blogbeiträge bieten eine Referenzarchitektur für die Erstellung eines MFT-Workflows mithilfe von SFTP-Konnektoren, einschließlich der Verschlüsselung von Dateien mit PGP, bevor sie mithilfe von SFTP-Konnektoren an einen Remote-SFTP-Server gesendet werden: Architektur sicherer und richtlinienkonformer verwalteter Dateiübertragungen mit SFTP-Konnektoren und PGP-Verschlüsselung. AWS Transfer Family

Unter AWS Transfer Family SFTP-Konnektoren finden Sie eine kurze Einführung in die SFTP-Konnektoren der Transfer Family.

Themen


- Konfigurieren Sie SFTP-Anschlüsse
- Senden und Abrufen von Dateien mithilfe eines SFTP-Connectors
- Inhalt eines Remote-Verzeichnisses auflisten
- SFTP-Konnektoren verwalten


## Konfigurieren Sie SFTP-Anschlüsse

In diesem Thema wird beschrieben, wie Sie SFTP-Konnektoren erstellen, welche SicherheitsalGORITHmen damit verknüpft sind, wie Sie ein Geheimnis für Anmeldeinformationen speichern, Einzelheiten zur Formatierung des privaten Schlüssels und Anweisungen zum Testen Ihrer Konnektoren.

Themen

- Erstellen Sie einen SFTP-Connector
- Speichern Sie ein Geheimnis zur Verwendung mit einem SFTP-Connector
- Generieren und formatieren Sie den privaten Schlüssel des SFTP-Connectors
- Testen Sie einen SFTP-Connector

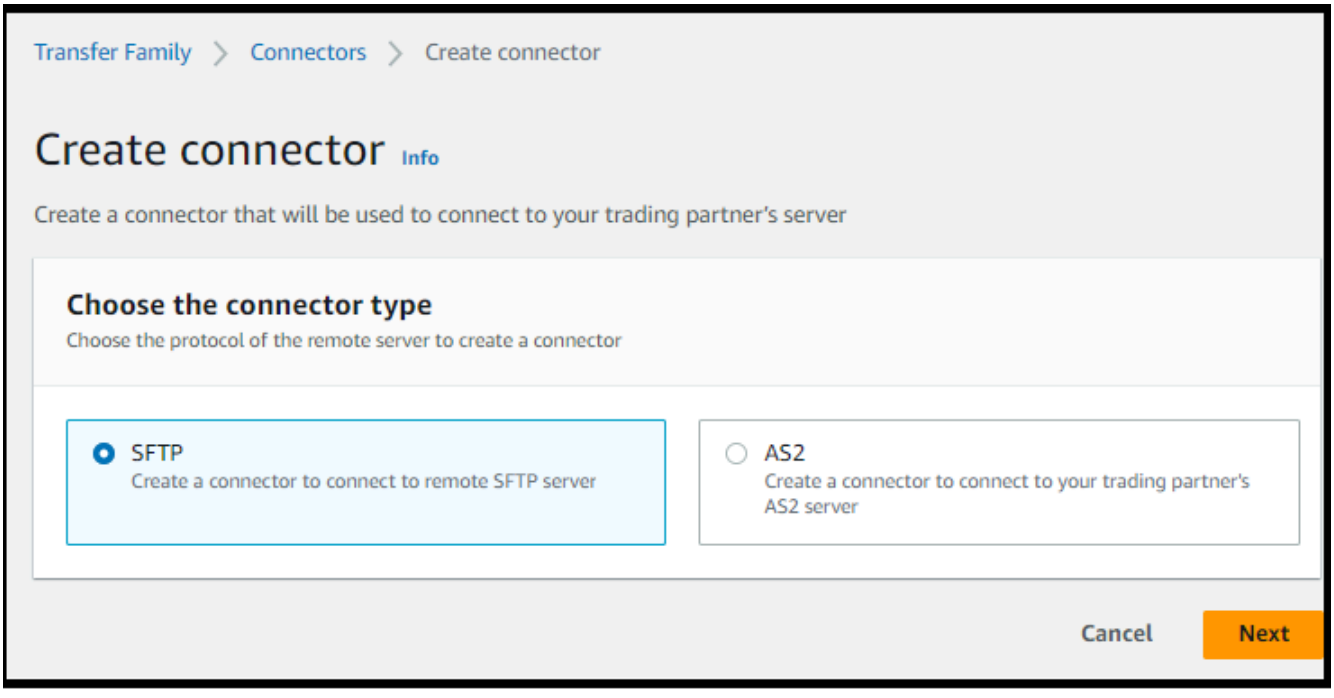
## Erstellen Sie einen SFTP-Connector

In diesem Verfahren wird erklärt, wie Sie SFTP-Konnektoren mithilfe der AWS Transfer Family Konsole oder erstellen. AWS CLI

Console

Um einen SFTP-Connector zu erstellen

1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Wählen Sie im linken Navigationsbereich Connectors und anschließend Create Connector aus.
3. Wählen Sie SFTP als Konnektortyp aus, um einen SFTP-Connector zu erstellen, und wählen Sie dann Weiter aus.




4. Geben Sie im Abschnitt Connector-Konfiguration die folgenden Informationen ein:

- Geben Sie für die URL die URL für einen Remote-SFTP-Server ein. Diese URL muss beispielsweise als `sftp://`*`partner-SFTP-server-url`* formatiert sein. `sftp://AnyCompany.com`

  > **Note**
  >
  > Optional können Sie in Ihrer URL eine Portnummer angeben. Das Format ist `sftp://`*`partner-SFTP-server-url`*`:`*`port-number`*. Die Standard-Portnummer (wenn kein Port angegeben ist) ist Port 22.

- Wählen Sie für die Access-Rolle den Amazon-Ressourcennamen (ARN) der zu verwendenden AWS Identity and Access Management (IAM) -Rolle aus.
  - Stellen Sie sicher, dass diese Rolle Lese- und Schreibzugriff auf das übergeordnete Verzeichnis des Dateispeicherorts bietet, der in der `StartFileTransfer` Anfrage verwendet wird.
  - Stellen Sie sicher, dass diese Rolle die Berechtigung **`secretsmanager:GetSecretValue`** zum Zugriff auf den geheimen Schlüssel gewährt.

    > **Note**
    >
    > In der Richtlinie müssen Sie den ARN für das Geheimnis angeben. Der ARN enthält den geheimen Namen, fügt dem Namen jedoch sechs zufällige alphanumerische Zeichen hinzu. Ein ARN für ein Geheimnis hat das folgende Format.
    >
    > ```
    > arn:aws:secretsmanager:region:account-id:secret:aws/
    > transfer/SecretName-6RandomCharacters
    > ```

  - Stellen Sie sicher, dass diese Rolle eine Vertrauensstellung enthält, die es dem Connector ermöglicht, auf Ihre Ressourcen zuzugreifen, wenn er Übertragungsanfragen Ihrer Benutzer bearbeitet. Einzelheiten zum Aufbau einer Vertrauensbeziehung finden Sie unterSo stellen Sie eine Vertrauensbeziehung her.

Das folgende Beispiel gewährt die erforderlichen Berechtigungen für den Zugriff auf den *DOC-EXAMPLE-BUCKET* in Amazon S3 und den angegebenen Secret, der in Secrets Manager gespeichert ist.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
        "Sid": "AllowListingOfUserFolder",
        "Action": [
            "s3:ListBucket",
            "s3:GetBucketLocation"
        ],
        "Effect": "Allow",
        "Resource": [
            "arn:aws:s3:::DOC-EXAMPLE-BUCKET"
        ]
    },
    {
        "Sid": "HomeDirObjectAccess",
        "Effect": "Allow",
        "Action": [
            "s3:PutObject",
            "s3:GetObject",
            "s3:DeleteObject",
            "s3:DeleteObjectVersion",
            "s3:GetObjectVersion",
            "s3:GetObjectACL",
            "s3:PutObjectACL"
        ],
        "Resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
    },
    {
        "Sid": "GetConnectorSecretValue",
        "Effect": "Allow",
        "Action": [
            "secretsmanager:GetSecretValue"
        ],
        "Resource": "arn:aws:secretsmanager:region:account-id:secret:aws/
transfer/SecretName-6RandomCharacters"
    }
  ]
```

```
}
```

> **Note**
>
> Für die Zugriffsrolle gewährt das Beispiel Zugriff auf ein einzelnes Geheimnis. Sie können jedoch ein Platzhalterzeichen verwenden, was Ihnen Arbeit ersparen kann, wenn Sie dieselbe IAM-Rolle für mehrere Benutzer und Geheimnisse wiederverwenden möchten. Die folgende Ressourcenanweisung gewährt beispielsweise Berechtigungen für alle Geheimnisse, deren Namen mit beginnen. `aws/transfer`
>
> ```
> "Resource": "arn:aws:secretsmanager:region:account-id:secret:aws/
> transfer/*"
> ```
>
> Sie können Geheimnisse, die Ihre SFTP-Anmeldeinformationen enthalten, auch in einem anderen AWS-Konto speichern. Einzelheiten zur Aktivierung des kontoübergreifenden geheimen Zugriffs finden Sie unter Berechtigungen für AWS Secrets Manager geheime Daten für Benutzer in einem anderen Konto.

- (Optional) Wählen Sie für die Logging-Rolle die IAM-Rolle aus, die der Connector verwenden soll, um Ereignisse in Ihre CloudWatch Logs zu übertragen. In der folgenden Beispielrichtlinie sind die erforderlichen Berechtigungen für die Protokollierung von Ereignissen für SFTP-Konnektoren aufgeführt.

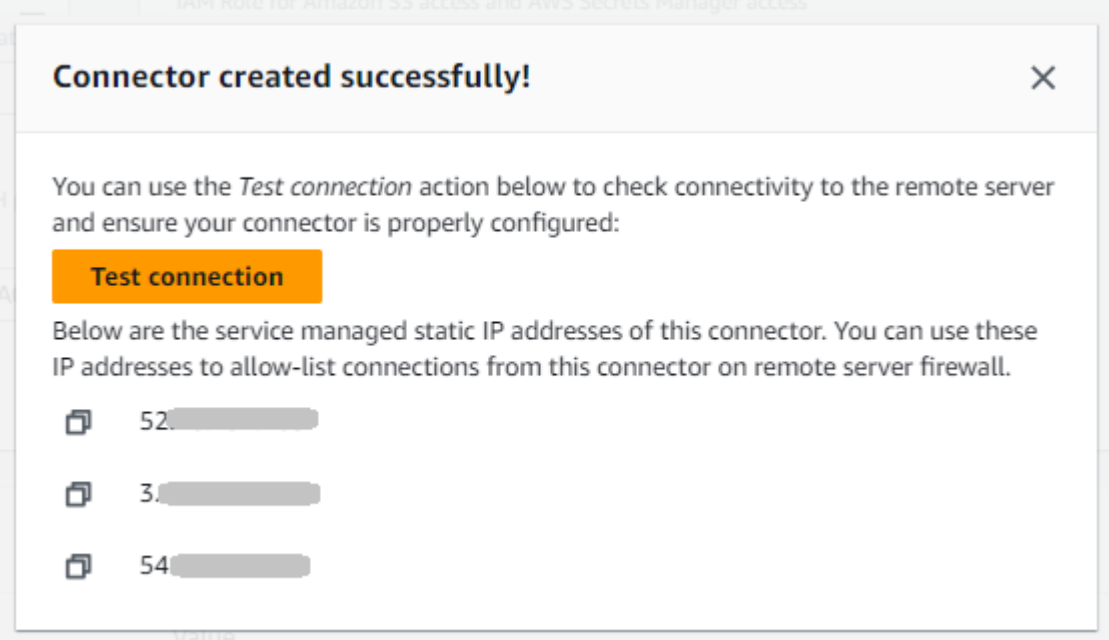
```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Sid": "SFTPConnectorPermissions",
        "Effect": "Allow",
        "Action": [
            "logs:CreateLogStream",
            "logs:DescribeLogStreams",
            "logs:CreateLogGroup",
            "logs:PutLogEvents"
        ],
        "Resource": [
            "arn:aws:logs:*:*:log-group:/aws/transfer/*"
        ]
    }]
```

```
}
```

5. Geben Sie im Abschnitt SFTP-Konfiguration die folgenden Informationen ein:

   - Wählen Sie für Connector-Anmeldeinformationen aus der Dropdownliste den Namen eines Geheimnisses aus, das den privaten Schlüssel oder AWS Secrets Manager das Passwort des SFTP-Benutzers enthält. Sie müssen ein Geheimnis erstellen und es auf eine bestimmte Weise speichern. Details hierzu finden Sie unter Speichern Sie ein Geheimnis zur Verwendung mit einem SFTP-Connector.
   - Fügen Sie für vertrauenswürdige Hostschlüssel den öffentlichen Teil des Hostschlüssels ein, der zur Identifizierung des externen Servers verwendet wird. Sie können mehr als einen Schlüssel hinzufügen, indem Sie Vertrauenswürdigen Hostschlüssel hinzufügen wählen, um einen zusätzlichen Schlüssel hinzuzufügen. Sie können den `ssh-keyscan` Befehl für den SFTP-Server verwenden, um den erforderlichen Schlüssel abzurufen. Einzelheiten zum Format und Typ der vertrauenswürdigen Hostschlüssel, die Transfer Family unterstützt, finden Sie unter SFTPConnectorConfig.

6. Wählen Sie im Abschnitt Optionen für kryptografische Algorithmen eine Sicherheitsrichtlinie aus der Dropdownliste im Feld Sicherheitsrichtlinie aus. Mit der Sicherheitsrichtlinie können Sie die kryptografischen Algorithmen auswählen, die Ihr Connector unterstützt. Einzelheiten zu den verfügbaren Sicherheitsrichtlinien und Algorithmen finden Sie unterAWS Transfer Family Sicherheitsrichtlinien für SFTP-Konnektoren.
7. (Optional) Geben Sie im Abschnitt Tags für Schlüssel und Wert ein oder mehrere Tags als Schlüssel-Wert-Paare ein.
8. Nachdem Sie alle Ihre Einstellungen bestätigt haben, wählen Sie Create Connector aus, um den SFTP-Connector zu erstellen. Wenn der Connector erfolgreich erstellt wurde, erscheint ein Bildschirm mit einer Liste der zugewiesenen statischen IP-Adressen und der Schaltfläche Verbindung testen. Verwenden Sie die Schaltfläche, um die Konfiguration für Ihren neuen Connector zu testen.




Die Seite Connectors wird angezeigt, auf der die ID Ihres neuen SFTP-Connectors zur Liste hinzugefügt wurde. Einzelheiten zu Ihren Konnektoren finden Sie unterDetails zum SFTP-Connector anzeigen.

CLI

Sie verwenden den create-connectorBefehl, um einen Konnektor zu erstellen. Um mit diesem Befehl einen SFTP-Connector zu erstellen, müssen Sie die folgenden Informationen angeben.

- Die URL für einen Remote-SFTP-Server. Diese URL muss beispielsweise als `sftp://partner-SFTP-server-url` formatiert sein. `sftp://AnyCompany.com`
- Die Zugriffsrolle. Wählen Sie den Amazon-Ressourcennamen (ARN) der zu AWS Identity and Access Management verwendenden (IAM) -Rolle aus.
  - Stellen Sie sicher, dass diese Rolle Lese- und Schreibzugriff auf das übergeordnete Verzeichnis des Dateispeicherorts bietet, der in der `StartFileTransfer` Anfrage verwendet wird.
  - Stellen Sie sicher, dass diese Rolle die Berechtigung **`secretsmanager:GetSecretValue`** zum Zugriff auf den geheimen Schlüssel gewährt.

> **Note**
>
> In der Richtlinie müssen Sie den ARN für das Geheimnis angeben. Der ARN enthält den geheimen Namen, fügt dem Namen jedoch sechs zufällige alphanumerische Zeichen hinzu. Ein ARN für ein Geheimnis hat das folgende Format.
>
> ```
> arn:aws:secretsmanager:region:account-id:secret:aws/
> transfer/SecretName-6RandomCharacters
> ```

- Stellen Sie sicher, dass diese Rolle eine Vertrauensstellung enthält, die es dem Connector ermöglicht, auf Ihre Ressourcen zuzugreifen, wenn er Übertragungsanfragen Ihrer Benutzer bearbeitet. Einzelheiten zum Aufbau einer Vertrauensbeziehung finden Sie unterSo stellen Sie eine Vertrauensbeziehung her.

Das folgende Beispiel gewährt die erforderlichen Berechtigungen für den Zugriff auf den *DOC-EXAMPLE-BUCKET* in Amazon S3 und den angegebenen Secret, der in Secrets Manager gespeichert ist.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
        "Sid": "AllowListingOfUserFolder",
        "Action": [
            "s3:ListBucket",
            "s3:GetBucketLocation"
        ],
        "Effect": "Allow",
        "Resource": [
            "arn:aws:s3:::DOC-EXAMPLE-BUCKET"
        ]
    },
    {
        "Sid": "HomeDirObjectAccess",
        "Effect": "Allow",
        "Action": [
            "s3:PutObject",
            "s3:GetObject",
            "s3:DeleteObject",
            "s3:DeleteObjectVersion",
```

```
            "s3:GetObjectVersion",
            "s3:GetObjectACL",
            "s3:PutObjectACL"
        ],
        "Resource": "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
    },
    {
        "Sid": "GetConnectorSecretValue",
        "Effect": "Allow",
        "Action": [
            "secretsmanager:GetSecretValue"
        ],
        "Resource": "arn:aws:secretsmanager:region:account-id:secret:aws/
transfer/SecretName-6RandomCharacters"
    }
  ]
}
```

> **Note**
>
> Für die Zugriffsrolle gewährt das Beispiel Zugriff auf ein einzelnes Geheimnis. Sie können jedoch ein Platzhalterzeichen verwenden, was Ihnen Arbeit ersparen kann, wenn Sie dieselbe IAM-Rolle für mehrere Benutzer und Geheimnisse wiederverwenden möchten. Die folgende Ressourcenanweisung gewährt beispielsweise Berechtigungen für alle Geheimnisse, deren Namen mit beginnen. `aws/transfer`
>
> ```
> "Resource": "arn:aws:secretsmanager:region:account-id:secret:aws/transfer/
> *"
> ```
>
> Sie können Geheimnisse, die Ihre SFTP-Anmeldeinformationen enthalten, auch in einem anderen AWS-Konto speichern. Einzelheiten zur Aktivierung des kontoübergreifenden geheimen Zugriffs finden Sie unter Berechtigungen für AWS Secrets Manager geheime Daten für Benutzer in einem anderen Konto.

- (Optional) Wählen Sie die IAM-Rolle aus, die der Connector verwenden soll, um Ereignisse in Ihre CloudWatch Logs zu übertragen. In der folgenden Beispielrichtlinie sind die erforderlichen Berechtigungen für die Protokollierung von Ereignissen für SFTP-Konnektoren aufgeführt.

```
{
    "Version": "2012-10-17",
```

```
    "Statement": [{
        "Sid": "SFTPConnectorPermissions",
        "Effect": "Allow",
        "Action": [
            "logs:CreateLogStream",
            "logs:DescribeLogStreams",
            "logs:CreateLogGroup",
            "logs:PutLogEvents"
        ],
        "Resource": [
            "arn:aws:logs:*:*:log-group:/aws/transfer/*"
        ]
    }]
}
```

- Geben Sie die folgenden SFTP-Konfigurationsinformationen an.
  - Der ARN eines Geheimnisses AWS Secrets Manager , das den privaten Schlüssel oder das Passwort des SFTP-Benutzers enthält.
  - Der öffentliche Teil des Hostschlüssels, der zur Identifizierung des externen Servers verwendet wird. Sie können mehrere vertrauenswürdige Hostschlüssel angeben, wenn Sie möchten.

  Der einfachste Weg, die SFTP-Informationen bereitzustellen, besteht darin, sie in einer Datei zu speichern. Kopieren Sie beispielsweise den folgenden Beispieltext in eine Datei mit dem Name`testSFTPConfig.json`.

```
// Listing for testSFTPConfig.json
{
   "UserSecretId": "arn:aws::secretsmanager:us-east-2:123456789012:secret:aws/
transfer/example-username-key",
   "TrustedHostKeys": [
      "sftp.example.com ssh-rsa AAAAbbbb...EEEE="
   ]
}
```

- Geben Sie eine Sicherheitsrichtlinie für Ihren Connector an und geben Sie den Namen der Sicherheitsrichtlinie ein.

> **Note**
>
> Das `SecretId` kann entweder der gesamte ARN oder der Name des Geheimnisses sein (*`example-username-key`* in der vorherigen Liste).

Führen Sie dann den folgenden Befehl aus, um den Connector zu erstellen.

```
aws transfer create-connector --url "sftp://partner-SFTP-server-url" \
--access-role your-IAM-role-for-bucket-access \
--logging-role arn:aws:iam::your-account-id:role/service-role/
AWSTransferLoggingAccess \
--sftp-config file:///path/to/testSFTPConfig.json
--security-policy-name security-policy-name
```

## Speichern Sie ein Geheimnis zur Verwendung mit einem SFTP-Connector

Sie können Secrets Manager verwenden, um Benutzeranmeldeinformationen für Ihre SFTP-Konnektoren zu speichern. Wenn Sie Ihr Geheimnis erstellen, müssen Sie einen Benutzernamen angeben. Darüber hinaus können Sie entweder ein Passwort, einen privaten Schlüssel oder beides angeben. Details hierzu finden Sie unter Kontingente für SFTP-Konnektoren.

> **Note**
>
> Wenn Sie Geheimnisse im Secrets Manager speichern, AWS-Konto fallen Gebühren an. Informationen zu Preisen erhalten Sie unter AWS Secrets Manager -Preise.

Um Benutzeranmeldeinformationen in Secrets Manager für einen SFTP-Connector zu speichern

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die AWS Secrets Manager Konsole unter https://console.aws.amazon.com/secretsmanager/.
2. Wählen Sie im linken Navigationsbereich Secrets aus.
3. Wählen Sie auf der Seite Secrets die Option Neues Geheimnis speichern aus.
4. Wählen Sie auf der Seite Geheimtyp auswählen für Geheimtyp die Option Anderer Geheimtyp aus.
5. Wählen Sie im Abschnitt Schlüssel/Wert-Paare die Registerkarte Schlüssel/Wert aus.

- Schlüssel — Geben Sie ein. **`Username`**
- Wert — Geben Sie den Namen des Benutzers ein, der berechtigt ist, eine Verbindung zum Server des Partners herzustellen.

6. Wenn Sie ein Passwort angeben möchten, wählen Sie Zeile hinzufügen und wählen Sie im Abschnitt Schlüssel/Wert-Paare die Registerkarte Schlüssel/Wert.

   Wählen Sie Zeile hinzufügen und wählen Sie im Abschnitt Schlüssel/Wert-Paare die Registerkarte Schlüssel/Wert-Paare aus.

   - Schlüssel — Geben Sie ein. **`Password`**
   - Wert — Geben Sie das Passwort für den Benutzer ein.

7. Wenn Sie einen privaten Schlüssel angeben möchten, finden Sie unterGenerieren und formatieren Sie den privaten Schlüssel des SFTP-Connectors, wo beschrieben wird, wie Sie private Schlüsseldaten eingeben.

   > **Note**
   >
   > Die von Ihnen eingegebenen privaten Schlüsseldaten müssen dem öffentlichen Schlüssel entsprechen, der für diesen Benutzer auf dem Remote-SFTP-Server gespeichert ist.

8. Wählen Sie Weiter aus.
9. Geben Sie auf der Seite Geheimen Schlüssel konfigurieren einen Namen und eine Beschreibung für Ihr Geheimnis ein. Wir empfehlen, dass Sie **`aws/transfer/`** für den Namen das Präfix von verwenden. Sie könnten beispielsweise Ihr Geheimnis benennen**`aws/transfer/connector-1`**.
10. Wählen Sie Weiter und akzeptieren Sie dann die Standardeinstellungen auf der Seite „Rotation konfigurieren“. Wählen Sie anschließend Weiter.
11. Wählen Sie auf der Seite „Überprüfen“ die Option Speichern aus, um das Geheimnis zu erstellen und zu speichern.

# Generieren und formatieren Sie den privaten Schlüssel des SFTP-Connectors

Vollständige Informationen zur Generierung eines öffentlichen/privaten key pair finden Sie unter. SSH-Schlüssel auf macOS, Linux oder Unix erstellen

Um beispielsweise einen privaten Schlüssel für die Verwendung mit SFTP-Konnektoren zu generieren, erzeugt der folgende Beispielbefehl den richtigen Schlüsseltyp (ersetzen Sie *key_name* durch den tatsächlichen Dateinamen für Ihr key pair):

```
ssh-keygen -t rsa -b 4096 -m PEM -f key_name -N ""
```

> **Note**
>
> Verwenden Sie keine Passphrase, wenn Sie Ihr key pair für die Verwendung mit SFTP-Anschlüssen erstellen. Eine leere Passphrase ist erforderlich, damit die SFTP-Konfiguration korrekt funktioniert.

Dieser Befehl erstellt ein RSA-Schlüsselpaar mit einer Schlüsselgröße von 4096 Bit. Der Schlüssel wird im alten PEM-Format generiert, das von Transfer Family für die Verwendung mit dem geheimen SFTP-Connector benötigt wird. Die Schlüssel werden in *key_name* (privater Schlüssel) und *key_name*`.pub` (öffentlicher Schlüssel) im aktuellen Verzeichnis gespeichert, d. h. in dem Verzeichnis, in dem Sie den Befehl ausführen. `ssh-keygen`

> **Note**
>
> Transfer Family unterstützt das OpenSSH-Format (`-----BEGIN OPENSSH PRIVATE KEY-----`) für die Schlüssel, die für Ihren SFTP-Anschluss verwendet werden, nicht. Der Schlüssel muss im alten PEM-Format (oder) vorliegen. `-----BEGIN RSA PRIVATE KEY----- -----BEGIN EC PRIVATE KEY-----` Sie können das ssh-keygen Tool verwenden, um Ihren Schlüssel zu konvertieren, indem Sie die `-m PEM` Option angeben, wenn Sie den Befehl ausführen.

Nachdem Sie den Schlüssel generiert haben, müssen Sie sicherstellen, dass der private Schlüssel mit eingebetteten Zeilenumbruchzeichen (“\n„) im JSON-Format formatiert ist.

Verwenden Sie einen Befehl, um Ihren vorhandenen privaten Schlüssel in das richtige Format zu konvertieren — das JSON-Format mit eingebetteten Zeilenumbruchzeichen. Hier finden Sie Beispiele für `jq` Powershell und Powershell. Sie können jedes Tool oder jeden Befehl verwenden, mit dem Sie den privaten Schlüssel in das JSON-Format mit eingebetteten Zeilenumbruchzeichen konvertieren möchten.

jq command

In diesem Beispiel wird der `jq` Befehl verwendet, der von Download jq heruntergeladen werden kann.

```
jq -sR . path-to-private-key-file
```

Wenn sich Ihre private Schlüsseldatei beispielsweise in befindet`~/.ssh/my_private_key`, lautet der Befehl wie folgt.

```
jq -sR . ~/.ssh/my_private_key
```

Dadurch wird der Schlüssel im richtigen Format (mit eingebetteten Zeilenumbruchzeichen) in die Standardausgabe ausgegeben.

PowerShell

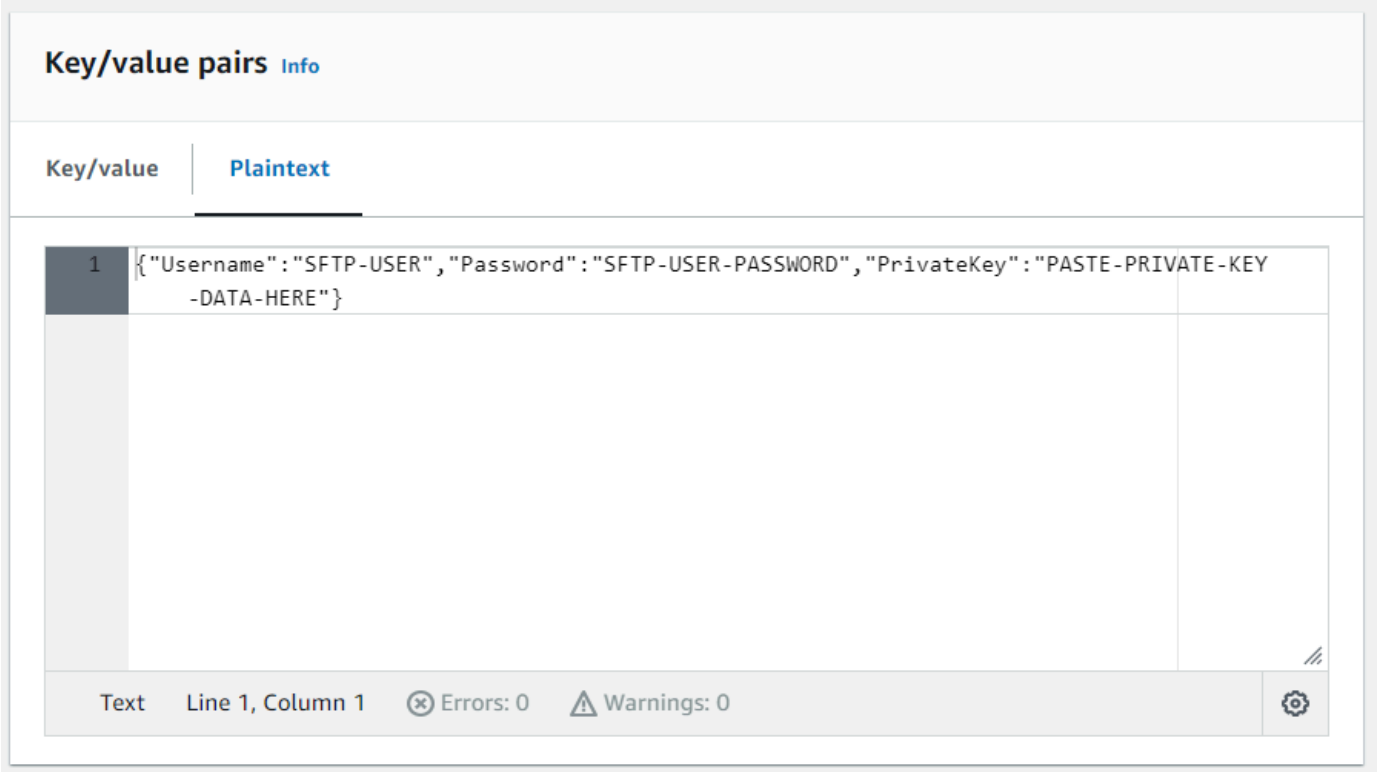
Wenn Sie Windows verwenden, können Sie PowerShell damit den Schlüssel in das richtige Format konvertieren. Der folgende Powershell-Befehl konvertiert den privaten Schlüssel in das richtige Format.

```
Get-Content -Raw path-to-private-key-file | ConvertTo-Json
```

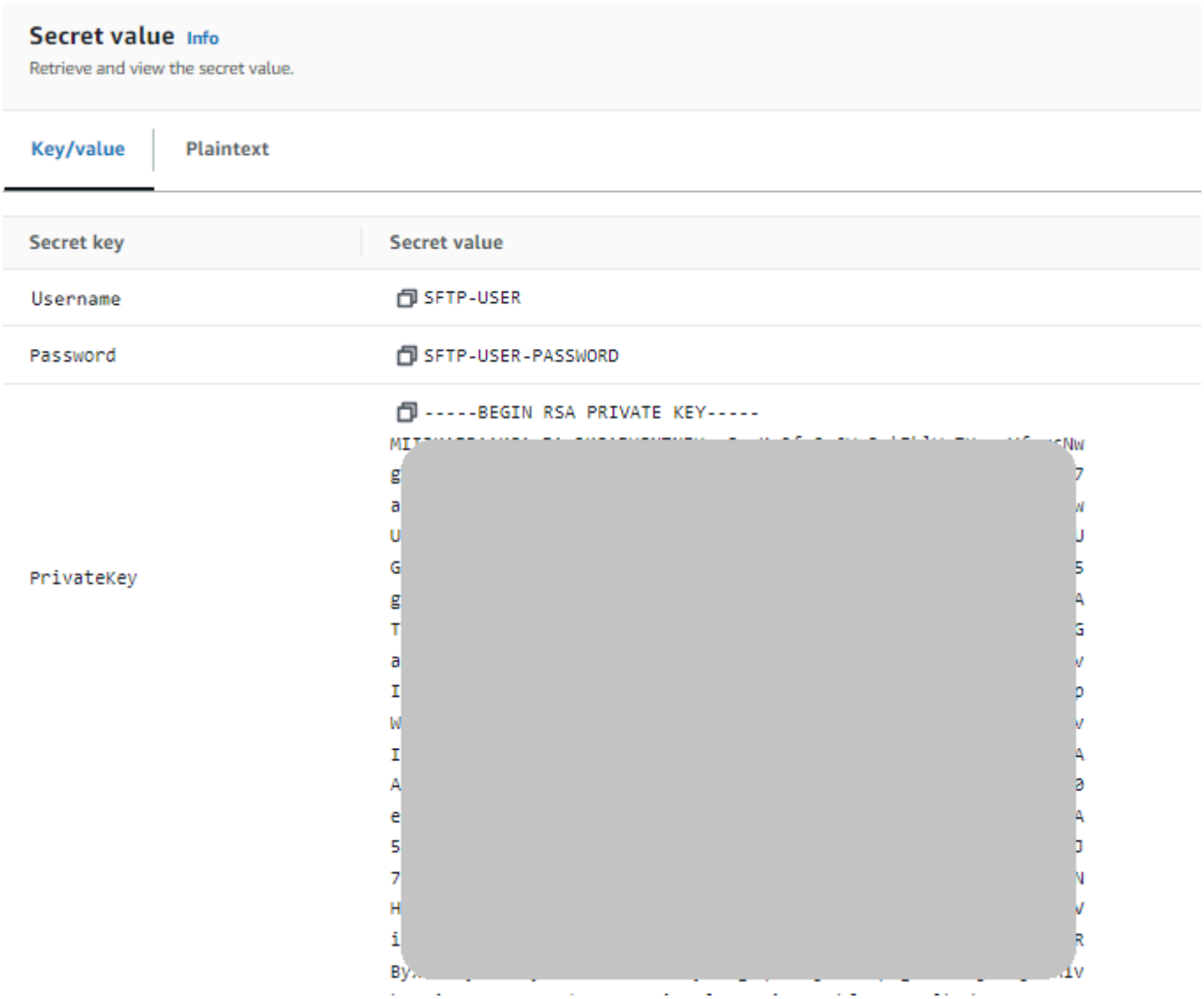
Um dem Secret private Schlüsseldaten zur Verwendung mit SFTP-Konnektoren hinzuzufügen

1. Wählen Sie in der Secrets Manager Manager-Konsole beim Speichern von Andere Arten von Geheimnissen die Registerkarte Klartext aus. Der Text sollte leer sein und nur eine öffnende und schließende Klammer, {}, enthalten.
2. Fügen Sie Ihren Benutzernamen, Ihre privaten Schlüsseldaten und/oder Ihr Passwort im folgenden Format ein. Fügen Sie für Ihre privaten Schlüsseldaten die Ausgabe des Befehls ein, den Sie in Schritt 1 ausgeführt haben.

```
{"Username":"SFTP-USER","Password":"SFTP-USER-PASSWORD","PrivateKey":"PASTE-
PRIVATE-KEY-DATA-HERE"}
```




Wenn Sie die privaten Schlüsseldaten korrekt einfügen, sollten Sie bei Auswahl der Registerkarte Schlüssel/Wert Folgendes sehen. Beachten Sie, dass die Daten des privaten Schlüssels angezeigt werden line-by-line und nicht als fortlaufende Textfolge.




3. Fahren Sie mit dem Verfahren Speichern Sie ein Geheimnis zur Verwendung mit einem SFTP-Connector in Schritt 8 fort und folgen Sie diesem Verfahren bis zum Ende.

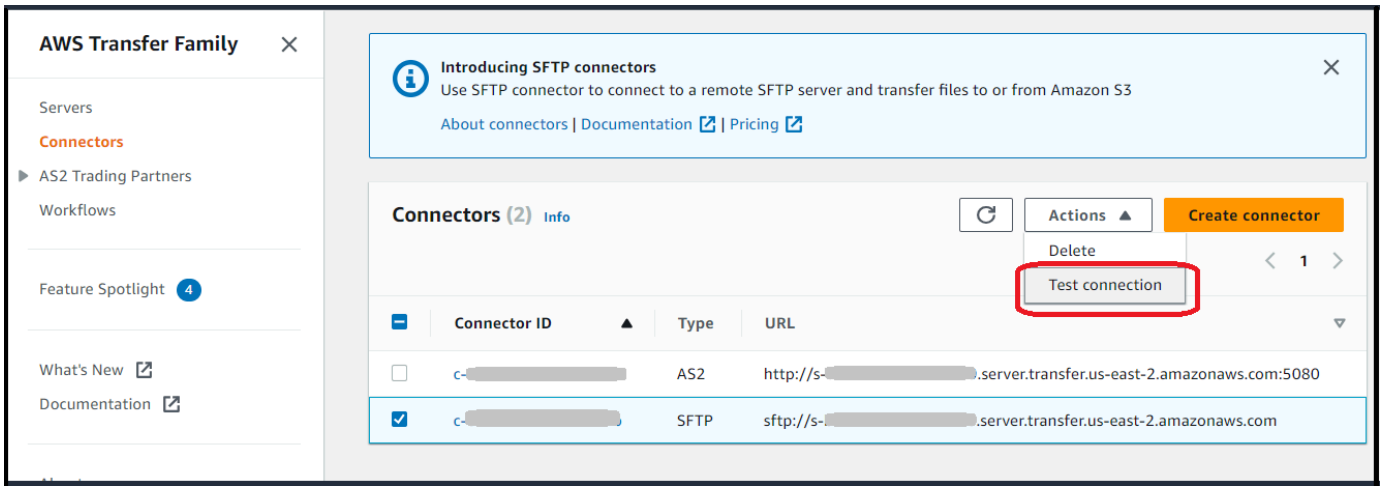
## Testen Sie einen SFTP-Connector

Nachdem Sie einen SFTP-Connector erstellt haben, empfehlen wir, ihn zu testen, bevor Sie versuchen, Dateien mit Ihrem neuen Connector zu übertragen.

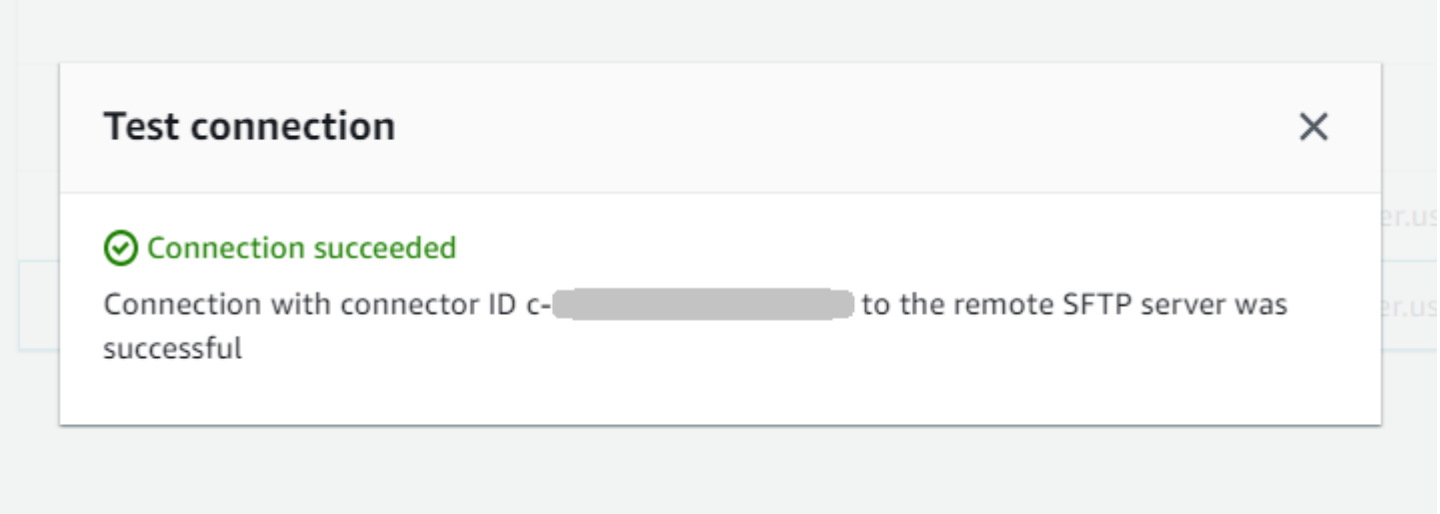
Um einen SFTP-Connector zu testen

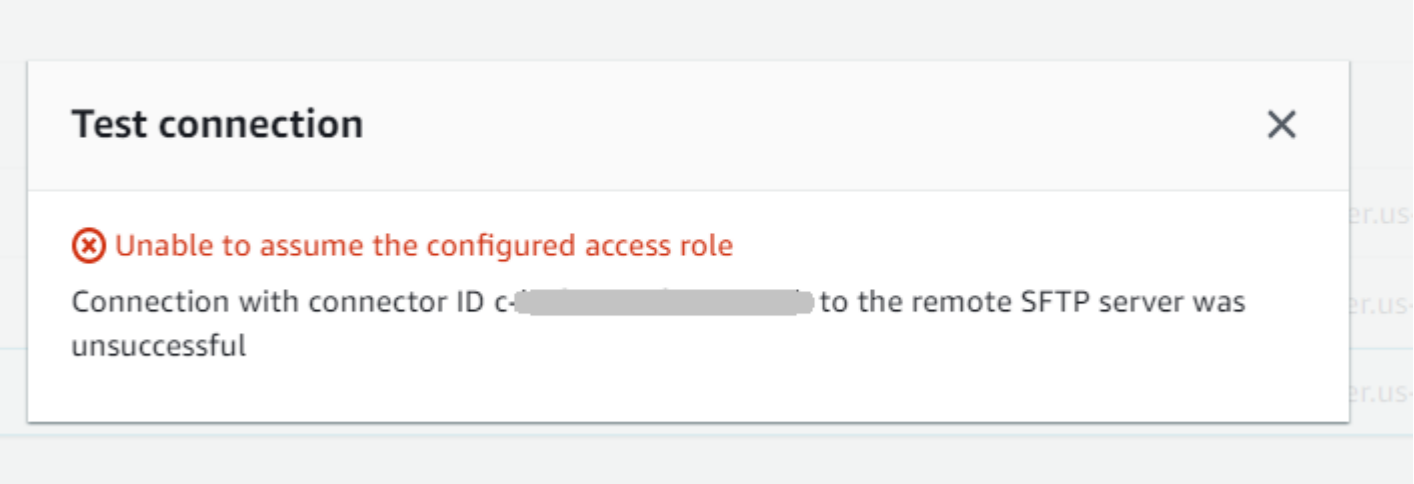
1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Wählen Sie im linken Navigationsbereich Connectors und wählen Sie einen Connector aus.

3. Wählen Sie im Menü Aktionen die Option Verbindung testen aus.




Das System gibt eine Meldung zurück, in der angegeben wird, ob der Test erfolgreich war oder nicht. Wenn der Test fehlschlägt, gibt das System eine Fehlermeldung aus, die auf dem Grund basiert, warum der Test fehlgeschlagen ist.






> **Note**
>
> Informationen zum Testen Ihres Connectors mithilfe der API finden Sie in der TestConnectionAPI-Dokumentation.

## Senden und Abrufen von Dateien mithilfe eines SFTP-Connectors

SFTP-Konnektoren erweitern die Möglichkeiten der AWS Transfer Family Kommunikation mit Remoteservern sowohl in der Cloud als auch vor Ort. Sie können Daten, die in Remote-Quellen generiert und gespeichert werden, in Ihre AWS gehosteten Data Warehouses für Analysen, Geschäftsanwendungen, Berichte und Prüfungen integrieren.

Um eine Dateiübertragung zu einem Remote-SFTP-Server zu initiieren, verwenden Sie den StartFileTransferAPI-Vorgang, bei dem SFTP-Konnektoren für die Übertragung verwendet werden. Jede `StartFileTransfer` Anfrage kann 10 verschiedene Pfade enthalten.

Sie können Ihre Dateiübertragungen überwachen, indem Sie Ihre Serverprotokolle überprüfen. Die Connector-Aktivität wird protokolliert, um Streams zu protokollierenaws/transfer/*connector-id*, die beispielsweise das Format habenaws/transfer/c-1234567890abcdef0. Wenn Sie keine Protokolle für Ihren Connector sehen, stellen Sie sicher, dass Sie eine Protokollierungsrolle mit den richtigen Berechtigungen für Ihren Connector angegeben haben.

Einzelheiten zum Erstellen von Konnektoren finden Sie unterKonfigurieren Sie SFTP-Anschlüsse.

Um Dateien mithilfe eines SFTP-Connectors zu senden und abzurufen, verwenden Sie den Befehl `start-file-transfer` AWS Command Line Interface (AWS CLI). Sie geben die folgenden

Parameter an, je nachdem, ob Sie Dateien senden (ausgehende Übertragungen) oder Dateien empfangen (eingehende Übertragungen).

- Ausgehende Übertragungen
  - `send-file-paths`enthält einen bis zehn Quelldateipfade für Dateien, die auf den SFTP-Server des Partners übertragen werden sollen.
  - `remote-directory-path`ist der Remote-Pfad, an den eine Datei auf dem SFTP-Server des Kunden gesendet werden soll.
- Eingehende Übertragungen
  - `retrieve-file-paths`enthält einen bis zehn Remote-Pfade. Jeder Pfad gibt einen Speicherort für die Übertragung von Dateien vom SFTP-Server des Partners auf Ihren Transfer Family Family-Server an.
  - `local-directory-path`ist der Amazon S3 S3-Speicherort (Bucket und optionales Präfix), an dem Ihre Dateien gespeichert sind.

Um Dateien zu senden, geben Sie die `remote-directory-path` Parameter `send-file-paths` und an. Sie können bis zu 10 Dateien für den `send-file-paths` Parameter angeben. Der folgende Beispielbefehl sendet die Dateien`/DOC-EXAMPLE-SOURCE-BUCKET/file2.txt`, die benannt sind `/DOC-EXAMPLE-SOURCE-BUCKET/file1.txt` und sich im Amazon S3 S3-Speicher befinden, an das `/tmp` Verzeichnis auf dem SFTP-Server Ihres Partners. Um diesen Beispielbefehl zu verwenden, ersetzen Sie den *`DOC-EXAMPLE-SOURCE-BUCKET`* durch Ihren eigenen Bucket.

```
aws transfer start-file-transfer --send-file-paths /DOC-EXAMPLE-SOURCE-BUCKET/
file1.txt /DOC-EXAMPLE-SOURCE-BUCKET/file2.txt \
    --remote-directory-path /tmp --connector-id c-1111AAAA2222BBBB3 --region us-east-2
```

Um Dateien zu empfangen, geben Sie die `local-directory-path` Parameter `retrieve-file-paths` und an. *`Das folgende Beispiel ruft die Dateien /my/remote/file2.txt auf dem SFTP-Server des Partners ab /my/remote/file1.txt und platziert sie im Amazon S3 S3-Speicherort /DOC-EXAMPLE-BUCKET/.`* Wenn Sie diesen Beispielbefehl verwenden möchten, ersetzen Sie *`user input placeholders`* durch Ihre Informationen.

```
aws transfer start-file-transfer --retrieve-file-paths /my/remote/file1.txt  /my/
remote/file2.txt \
   --local-directory-path /DOC-EXAMPLE-BUCKET/prefix --connector-id c-2222BBBB3333CCCC4
 --region us-east-2
```

Die vorherigen Beispiele spezifizieren absolute Pfade auf dem SFTP-Server. Sie können auch relative Pfade verwenden, d. h. Pfade, die relativ zum Home-Verzeichnis des SFTP-Benutzers sind. Wenn der SFTP-Benutzer beispielsweise `marymajor` und sein Home-Verzeichnis auf dem SFTP-Server sind`/users/marymajor/`, sendet der folgende Befehl an `/DOC-EXAMPLE-SOURCE-BUCKET/file1.txt` `/users/marymajor/test-connectors/file1.txt`

```
aws transfer start-file-transfer --send-file-paths /DOC-EXAMPLE-SOURCE-BUCKET/file1.txt
 \
   --remote-directory-path test-connectors --connector-id c-2222BBBB3333CCCC4 --
region us-east-2
```

## Inhalt eines Remote-Verzeichnisses auflisten

Bevor Sie Dateien von einem Remote-SFTP-Server abrufen, können Sie den Inhalt eines Verzeichnisses auf dem Remote-SFTP-Server abrufen. Dazu verwenden Sie den StartDirectoryListingAPI-Aufruf.

Das folgende Beispiel listet den Inhalt des `home` Ordners auf dem Remote-SFTP-Server auf, der in der Konfiguration des Connectors angegeben ist. Die Ergebnisse werden am Amazon S3 S3-Speicherort `/DOC-EXAMPLE-BUCKET/connector-files` und in einer Datei mit dem Namen abgelegt`c-AAAA1111BBBB2222C-6666abcd-11aa-22bb-cc33-0000aaaa3333.json`.

```
aws transfer start-directory-listing  \
   --connector-id c-AAAA1111BBBB2222C  \
   --output-directory-path /DOC-EXAMPLE-BUCKET/example/connector-files  \
   --remote-directory-path /home
```

Dieser AWS CLI Befehl gibt eine Listing-ID und den Namen der Datei zurück, die die Ergebnisse enthält.

```
{
    "ListingId": "6666abcd-11aa-22bb-cc33-0000aaaa3333",
    "OutputFileName": "c-AAAA1111BBBB2222C-6666abcd-11aa-22bb-cc33-0000aaaa3333.json"
}
```

> **Note**
>
> Die Benennungskonvention für die Ausgabedatei lautet*connector-ID*-*listing-ID*`.json`.

Die JSON-Datei enthält die folgenden Informationen:

- `filePath`: Der vollständige Pfad einer Remote-Datei, relativ zum Verzeichnis der Listing-Anfrage für Ihren SFTP-Connector auf dem Remoteserver.
- `modifiedTimestamp`: das letzte Mal, als die Datei geändert wurde, in Sekunden, UTC-Format (Coordinated Universal Time). Dies ist ein optionales Feld. Wenn die Attribute der Remote-Datei keinen Zeitstempel enthalten, wird dieser in der Dateiliste weggelassen.
- `size`: Die Größe der Datei in Byte. Dies ist ein optionales Feld. Wenn die Remote-Dateiattribute keine Dateigröße enthalten, wird sie in der Dateiliste weggelassen.
- `path`: der vollständige Pfad eines Remote-Verzeichnisses, relativ zum Verzeichnis der Listing-Anfrage für Ihren SFTP-Connector auf dem Remoteserver.
- `truncated`: ein Flag, das angibt, ob die Listenausgabe alle im Remote-Verzeichnis enthaltenen Elemente enthält oder nicht. Wenn Ihr `truncated` Ausgabewert wahr ist, können Sie den im optionalen `max-items` Eingabeattribut angegebenen Wert erhöhen, um mehr Elemente auflisten zu können (bis zur maximal zulässigen Listengröße von 10.000 Elementen).

Das Folgende ist ein Beispiel für den Inhalt der Ausgabedatei (`c-AAAA1111BBBB2222C-6666abcd-11aa-22bb-cc33-0000aaaa3333.json`), wobei das Remote-Verzeichnis zwei Dateien und zwei Unterverzeichnisse (Pfade) enthält.

```
{
    "files": [
        {
            "filePath": "/home/what.txt",
            "modifiedTimestamp": "2024-01-30T20:34:54Z",
            "size" : 2323
        },
        {
            "filePath": "/home/how.pgp",
            "modifiedTimestamp": "2024-01-30T20:34:54Z",
            "size" : 4691
        }
    ],
    "paths": [
        {
            "path": "/home/magic"
        },
        {
            "path": "/home/aws"
```

```
        },
    ],
    "truncated": "false"
}
```

# SFTP-Konnektoren verwalten

In diesem Thema wird beschrieben, wie SFTP-Konnektoren angezeigt und aktualisiert werden, und es werden Kontingente aufgeführt, die für SFTP-Konnektoren relevant sind.

> **Note**
>
> Jedem Connector werden automatisch statische IP-Adressen zugewiesen, die während der Lebensdauer des Connectors unverändert bleiben. Auf diese Weise können Sie eine Verbindung zu Remote-SFTP-Servern herstellen, die nur eingehende Verbindungen von bekannten IP-Adressen akzeptieren. Ihren Connectoren wird ein Satz statischer IP-Adressen zugewiesen, die von allen Connectoren gemeinsam genutzt werden, die dasselbe Protokoll (SFTP oder AS2) in Ihrem verwenden. AWS-Konto

Themen


- Aktualisieren Sie die SFTP-Konnektoren
- Details zum SFTP-Connector anzeigen
- Kontingente für SFTP-Konnektoren


## Aktualisieren Sie die SFTP-Konnektoren

Um die vorhandenen Parameterwerte für Ihre Konnektoren zu ändern, können Sie den `update-connector` Befehl ausführen. Mit dem folgenden Befehl wird das Geheimnis für den Connector *`connector-id`* in der Region *`region-id`* auf aktualisiert*`secret-ARN`*. Wenn Sie diesen Beispielbefehl verwenden möchten, ersetzen Sie *`user input placeholders`* durch Ihre Informationen.

```
aws transfer update-connector --sftp-config '{"UserSecretId":"secret-ARN"}' \
   --connector-id connector-id --region region-id
```

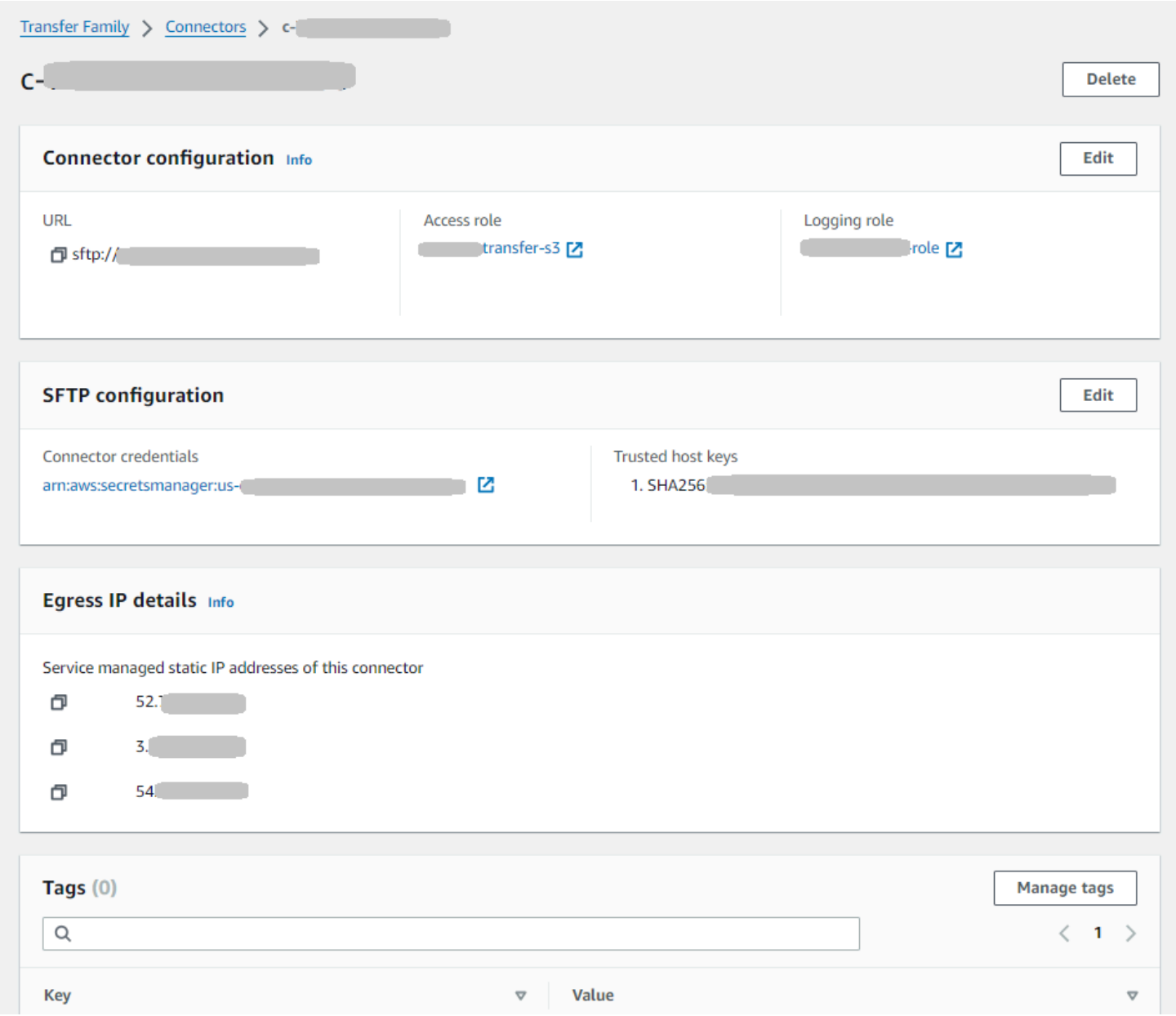
# Details zum SFTP-Connector anzeigen

In der Konsole finden Sie eine Liste mit Details und Eigenschaften für einen SFTP-Connector. AWS Transfer Family

Um die Connector-Details anzuzeigen

1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Wählen Sie im linken Navigationsbereich die Option Connectors aus.
3. Wählen Sie den Bezeichner in der Spalte Connector-ID aus, um die Detailseite für den ausgewählten Connector aufzurufen.

Sie können die Eigenschaften für den SFTP-Connector ändern, indem Sie auf der Connector-Detailseite auf Bearbeiten klicken.




> **Note**
>
> Sie können viele dieser Informationen abrufen, wenn auch in einem anderen Format, indem Sie den folgenden Befehl AWS Command Line Interface (AWS CLI) ausführen. Wenn Sie diesen Beispielbefehl verwenden möchten, ersetzen Sie *user input placeholders* durch Ihre Informationen.
>
> ```
> aws transfer describe-connector --connector-id your-connector-id
> ```

Weitere Informationen finden Sie DescribeConnectorin der API-Referenz.

## Kontingente für SFTP-Konnektoren

Die folgenden Kontingente gelten für SFTP-Konnektoren.

> **Note**
>
> Weitere Dienstkontingente für SFTP-Konnektoren sind unter AWS Transfer Family Endpunkte und Kontingente in der aufgeführt. Allgemeine Amazon Web Services-Referenz

Kontingente für SFTP-Konnektoren

| Name | Standard | Anpassbar |
|---|---|---|
| Maximale Anzahl an Testverbindungstransaktionen pro Sekunde (TPS) | 1 Anfrage pro Sekunde, pro Konto | Nein |
| Maximale Warteschlangengröße für ausstehende Dateiübertragungen | 1000 | Nein |
| Maximale Dateigröße | 50 Gibibyte (GiB) | Nein |
| Maximale Übertragungszeit pro Datei | 6 Stunden | Nein |
| Maximale Wartezeit für Anfragen pro Datei | 6 Stunden | Nein |
| Maximale Bandbreite für Konnektoren pro Konto (sowohl SFTP- als auch AS2-Konnektoren tragen zu diesem Wert bei) | 50 Mbit/s | Nein |

Für das Speichern der Anmeldeinformationen für SFTP-Konnektoren sind mit jedem Secrets Manager Manager-Geheimnis Kontingente verknüpft. Wenn Sie dasselbe Geheimnis zum Speichern mehrerer Schlüsseltypen für verschiedene Zwecke verwenden, können Sie auf diese Kontingente stoßen.

- Gesamtlänge eines einzelnen Geheimnisses: 12.000 Zeichen
- Maximale Länge der **`Password`** Zeichenfolge: 1024 Zeichen
- Maximale Länge der **`PrivateKey`** Zeichenfolge: 8192 Zeichen
- Maximale Länge der **`Username`** Zeichenfolge: 100 Zeichen

# AWS Transfer Family für AS2

Applicability Statement 2 (AS2) ist eine RFC-definierte Spezifikation für die Dateiübertragung, die starke Mechanismen zum Schutz und zur Überprüfung von Nachrichten umfasst. Das AS2-Protokoll ist entscheidend für Workflows mit Compliance-Anforderungen, die darauf beruhen, dass Datenschutz- und Sicherheitsfunktionen in das Protokoll integriert sind.

> **Note**
>
> AS2 for Transfer Family ist Drummond-zertifiziert.

Kunden aus Branchen wie Einzelhandel, Biowissenschaften, Fertigung, Finanzdienstleistungen und Versorgungsunternehmen, die sich bei Lieferketten-, Logistik- und Zahlungsabläufen auf AS2 verlassen, können AWS Transfer Family AS2-Endpunkte verwenden, um sichere Transaktionen mit ihren Geschäftspartnern abzuwickeln. Die Transaktionsdaten sind AWS für die Verarbeitung, Analyse und maschinelles Lernen nativ zugänglich. Diese Daten sind auch für Integrationen mit ERP- (Enterprise Resource Planning) und CRM-Systemen (Customer Relationship Management) verfügbar, auf denen sie ausgeführt werden. AWS Mit AS2 können Kunden ihre business-to-business (B2B-) Transaktionen in großem Umfang durchführen und AWS gleichzeitig die bestehenden Geschäftspartnerintegrationen und die Einhaltung der Vorschriften beibehalten.

Wenn Sie ein Transfer Family Family-Kunde sind und Dateien mit einem Partner austauschen möchten, der über einen konfigurierten AS2-fähigen Server verfügt, umfasst das Setup die Generierung eines öffentlich-privaten key pair für die Verschlüsselung und eines weiteren zum Signieren und Austauschen der öffentlichen Schlüssel mit dem Partner.

Transfer Family bietet einen Workshop, an dem Sie teilnehmen können, in dem Sie einen Transfer Family-Endpunkt mit aktiviertem AS2 und einen Transfer Family AS2-Connector konfigurieren können. Die Details zu diesem Workshop finden Sie hier.

Der Schutz einer AS2-Nutzlast während der Übertragung erfordert in der Regel die Verwendung von Cryptographic Message Syntax (CMS) und verwendet häufig Verschlüsselung und eine digitale Signatur, um Datenschutz und Peer-Authentifizierung zu gewährleisten. Eine signierte MDN-Antwortnutzlast (Message Disposition Notice) dient der Bestätigung (Unwiderlegbarkeit), dass eine Nachricht empfangen und erfolgreich entschlüsselt wurde.

Der Transport dieser CMS-Payloads und MDN-Antworten erfolgt über HTTP.

> **Note**
>
> HTTPS AS2-Serverendpunkte werden derzeit nicht unterstützt. Die TLS-Terminierung liegt derzeit in der Verantwortung des Kunden.

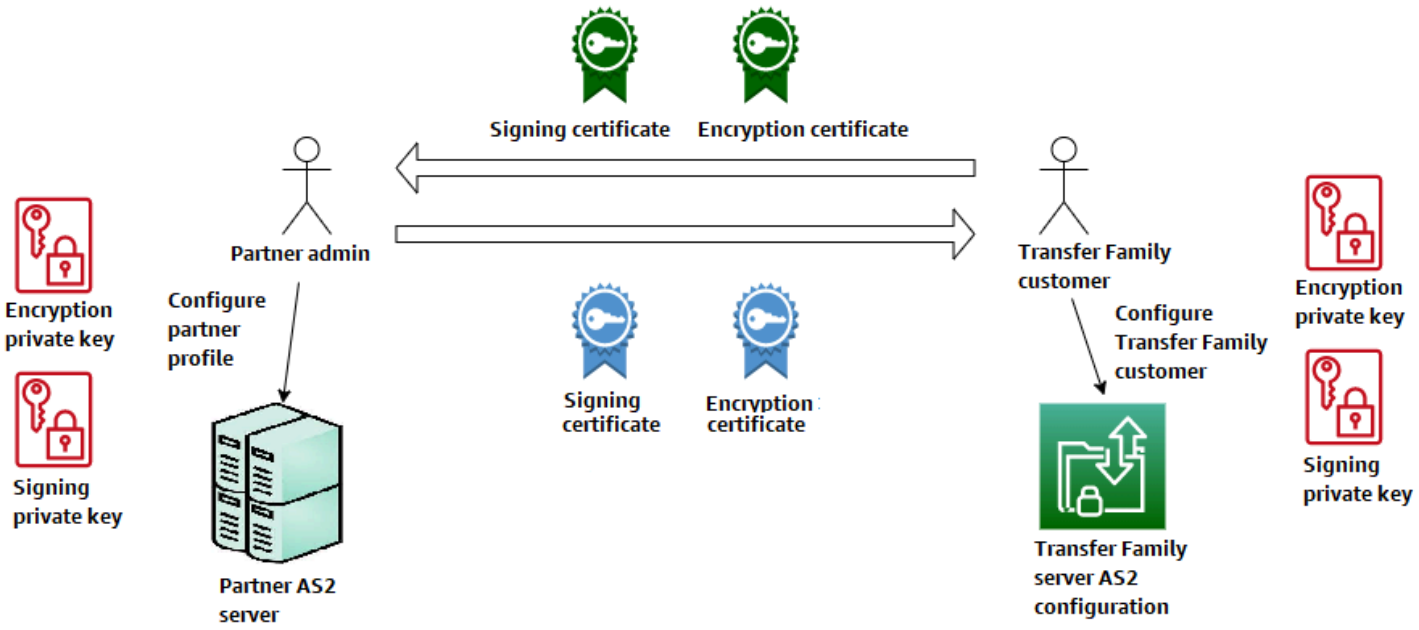
Eine ausführliche step-by-step Anleitung zur Einrichtung einer AS2-Konfiguration (Applicability Statement 2) finden Sie im Tutorial. Einrichtung einer AS2-Konfiguration

Themen


- AS2-Anwendungsfälle
- AS2 konfigurieren
- AS2-Konnektoren konfigurieren
- AS2-Partner verwalten
- Senden und Empfangen von AS2-Nachrichten
- Überwachen der AS2-Nutzung


# AS2-Anwendungsfälle

Wenn Sie ein AWS Transfer Family Kunde sind, der Dateien mit einem Partner austauschen möchte, der über einen konfigurierten AS2-Server verfügt, besteht der komplexeste Teil der Einrichtung darin, ein öffentlich-privates key pair für die Verschlüsselung und ein weiteres für das Signieren und Austauschen der öffentlichen Schlüssel mit dem Partner zu generieren.




Ziehen Sie die folgenden Varianten für die Verwendung AWS Transfer Family mit AS2 in Betracht.

> **Note**
>
> Handelspartner ist der Partner, der mit diesem Partnerprofil verknüpft ist.
> Bei allen Erwähnungen von MDN in der folgenden Tabelle wird von signierten mDNs ausgegangen.

## AS2-Anwendungsfälle

### Anwendungsfälle nur für eingehende Nachrichten

- Übertragen Sie verschlüsselte AS2-Nachrichten von einem Handelspartner auf einen Transfer Family Family-Server.

  Führen Sie in diesem Fall folgende Schritte aus:

  1. Erstellen Sie Profile für Ihren Handelspartner und sich selbst.
  2. Erstellen Sie einen Transfer Family Family-Server, der das AS2-Protokoll verwendet.
  3. Erstellen Sie eine Vereinbarung und fügen Sie sie Ihrem Server hinzu.
  4. Importieren Sie ein Zertifikat mit einem privaten Schlüssel und fügen Sie es Ihrem Profil hinzu. Importieren Sie dann den öffentlichen Schlüssel zur Verschlüsselung in Ihr Partnerprofil.

5. Nachdem Sie diese Elemente erhalten haben, senden Sie den öffentlichen Schlüssel für Ihr Zertifikat an Ihren Handelspartner.

Jetzt kann Ihr Partner Ihnen verschlüsselte Nachrichten senden und Sie können sie entschlüsseln und in Ihrem Amazon S3 S3-Bucket speichern.

- Übertragen Sie verschlüsselte AS2-Nachrichten von einem Handelspartner auf einen Transfer Family Family-Server und fügen Sie die Signatur hinzu.

  In diesem Szenario führen Sie immer noch nur eingehende Übertragungen durch, aber jetzt möchten Sie, dass Ihr Partner die von ihm gesendeten Nachrichten signiert. Importieren Sie in diesem Fall den öffentlichen Signaturschlüssel des Handelspartners (als Signaturzertifikat, das dem Profil Ihres Partners hinzugefügt wurde).
- Übertragen Sie verschlüsselte AS2-Nachrichten von einem Handelspartner auf einen Transfer Family Family-Server und fügen Sie das Signieren und Senden einer MDN-Antwort hinzu.

  In diesem Szenario führen Sie immer noch nur eingehende Übertragungen durch, aber Ihr Handelspartner möchte jetzt nicht nur signierte Payloads empfangen, sondern auch eine signierte MDN-Antwort erhalten.

  1. Importieren Sie Ihre öffentlichen und privaten Signaturschlüssel (als Signaturzertifikat in Ihr Profil).
  2. Senden Sie den öffentlichen Signaturschlüssel an Ihren Handelspartner.

Anwendungsfälle nur für ausgehende Anrufe

- Übertragen Sie verschlüsselte AS2-Nachrichten von einem Transfer Family Family-Server an einen Handelspartner.

  Dieser Fall ähnelt dem Anwendungsfall für eingehende Übertragungen, mit dem Unterschied, dass Sie, anstatt Ihrem AS2-Server eine Vereinbarung hinzuzufügen, einen Connector erstellen. In diesem Fall importieren Sie den öffentlichen Schlüssel Ihres Handelspartners in sein Profil.
- Übertragen Sie verschlüsselte AS2-Nachrichten von einem Transfer Family Family-Server an einen Handelspartner und fügen Sie die Signatur hinzu.

  Sie führen immer noch nur ausgehende Übertragungen durch, aber jetzt möchte Ihr Handelspa rtner, dass Sie die Nachricht, die Sie an ihn senden, signieren.

  1. Importieren Sie Ihren privaten Signaturschlüssel (als Signaturzertifikat, das Ihrem Profil hinzugefügt wurde).
  2. Senden Sie Ihrem Handelspartner Ihren öffentlichen Schlüssel.
- Übertragen Sie verschlüsselte AS2-Nachrichten von einem Transfer Family Family-Server an einen Handelspartner, fügen Sie die Signatur hinzu und senden Sie eine MDN-Antwort.

  Sie führen immer noch nur ausgehende Übertragungen durch, aber jetzt möchten Sie nicht nur signierte Payloads senden, sondern auch eine signierte MDN-Antwort von Ihrem Handelspartner erhalten.

  1. Ihr Handelspartner sendet Ihnen seinen öffentlichen Signaturschlüssel.
  2. Importieren Sie den öffentlichen Schlüssel Ihres Handelspartners (als Signaturzertifikat, das Ihrem Partnerprofil hinzugefügt wurde).

Anwendungsfälle für eingehenden und ausgehenden Datenverkehr

- Übertragen Sie verschlüsselte AS2-Nachrichten in beide Richtungen zwischen einem Transfer Family Family-Server und einem Handelspartner.

  Führen Sie in diesem Fall folgende Schritte aus:

  1. Erstellen Sie Profile für Ihren Handelspartner und sich selbst.
  2. Erstellen Sie einen Transfer Family Family-Server, der das AS2-Protokoll verwendet.
  3. Erstellen Sie eine Vereinbarung und fügen Sie sie Ihrem Server hinzu.
  4. Erstellen Sie einen Konnektor.
  5. Importieren Sie ein Zertifikat mit einem privaten Schlüssel und fügen Sie es Ihrem Profil hinzu. Importieren Sie dann den öffentlichen Schlüssel zur Verschlüsselung in Ihr Partnerpr ofil.
  6. Erhalten Sie einen öffentlichen Schlüssel von Ihrem Handelspartner und fügen Sie ihn zur Verschlüsselung zu seinem Profil hinzu.
  7. Nachdem Sie diese Artikel erhalten haben, senden Sie den öffentlichen Schlüssel für Ihr Zertifikat an Ihren Handelspartner.

  Jetzt können Sie und Ihr Handelspartner verschlüsselte Nachrichten austauschen, und Sie können sie beide entschlüsseln. Sie können die Nachrichten, die Sie erhalten, in Ihrem Amazon S3 S3-Bucket speichern, und Ihr Partner kann die Nachrichten, die Sie an ihn senden, entschlüs seln und speichern.
- Übertragen Sie verschlüsselte AS2-Nachrichten in beide Richtungen zwischen einem Transfer Family Family-Server und einem Handelspartner und fügen Sie die Signatur hinzu.

  Jetzt möchten Sie und Ihr Partner signierte Nachrichten.

  1. Importieren Sie Ihren privaten Signaturschlüssel (als Signaturzertifikat, das Ihrem Profil hinzugefügt wurde).
  2. Senden Sie Ihrem Handelspartner Ihren öffentlichen Schlüssel.
  3. Importieren Sie den öffentlichen Signaturschlüssel Ihres Handelspartners und fügen Sie ihn seinem Profil hinzu.
- Übertragen Sie verschlüsselte AS2-Nachrichten in beide Richtungen zwischen einem Transfer Family Family-Server und einem Handelspartner, fügen Sie die Signatur hinzu und senden Sie eine MDN-Antwort.

Jetzt möchten Sie signierte Payloads austauschen, und sowohl Sie als auch Ihr Handelspartner wünschen MDN-Antworten.

1. Ihr Handelspartner sendet Ihnen seinen öffentlichen Signaturschlüssel.
2. Importieren Sie den öffentlichen Schlüssel Ihres Handelspartners (als Signaturzertifikat in Ihr Partnerprofil).
3. Senden Sie Ihren öffentlichen Schlüssel an Ihren Handelspartner.

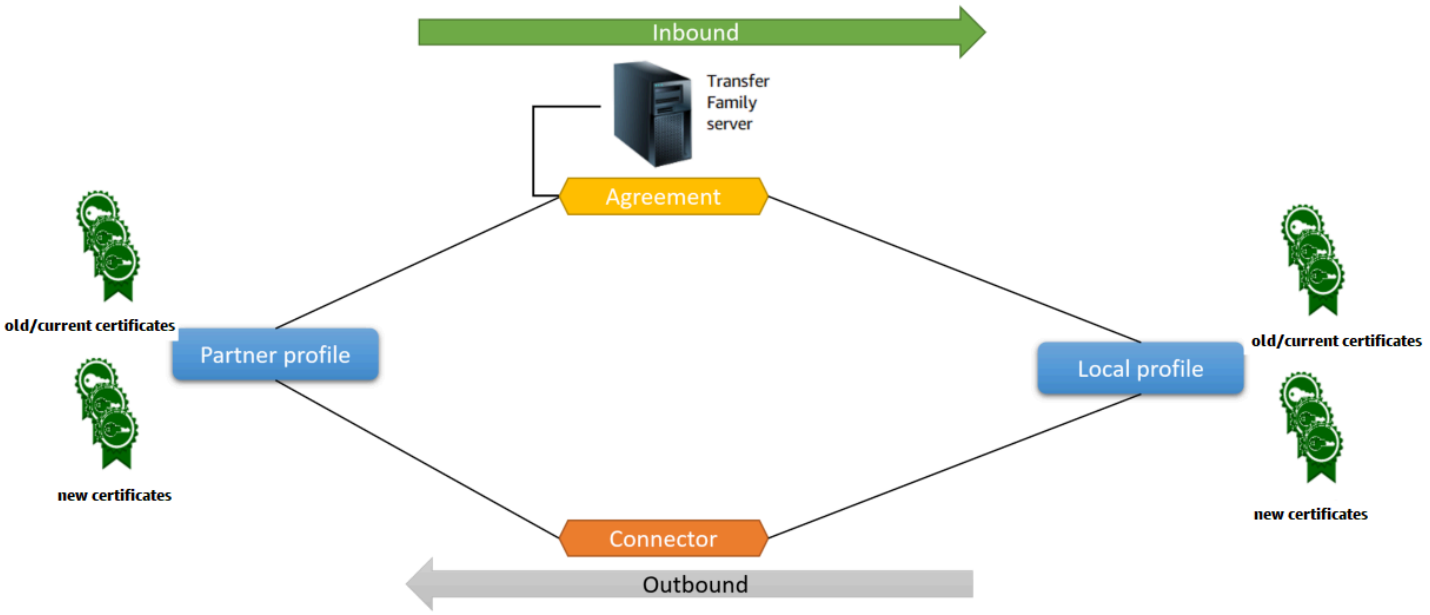
# AS2 konfigurieren

Um einen AS2-fähigen Server zu erstellen, müssen Sie außerdem die folgenden Komponenten angeben:

- Abkommen — Bilaterale Handelspartnerabkommen oder Partnerschaften definieren die Beziehung zwischen den beiden Parteien, die Nachrichten (Dateien) austauschen. Um eine Vereinbarung zu definieren, kombiniert Transfer Family Server-, lokale Profil-, Partnerprofil- und Zertifikatsinformationen. Transfer Family AS2-Inbound-Prozesse verwenden Vereinbarungen.
- Zertifikate — Zertifikate mit öffentlichem Schlüssel (X.509) werden in der AS2-Kommunikation zur Verschlüsselung und Überprüfung von Nachrichten verwendet. Zertifikate werden auch für Connector-Endpunkte verwendet.
- Lokale Profile und Partnerprofile — Ein lokales Profil definiert die lokale Organisation oder „Partei“ (AS2-fähiger Transfer Family Family-Server). In ähnlicher Weise definiert ein Partnerprofil die Remote-Partnerorganisation außerhalb von Transfer Family.

Obwohl nicht für alle AS2-fähigen Server erforderlich, benötigen Sie für ausgehende Übertragungen einen Connector. Ein Connector erfasst die Parameter für eine ausgehende Verbindung. Der Connector ist erforderlich, um Dateien an einen externen Server eines Kunden zu senden, der kein AWS Server ist.

Das folgende Diagramm zeigt die Beziehung zwischen den AS2-Objekten, die an den eingehenden und ausgehenden Prozessen beteiligt sind.




Ein end-to-end Beispiel für eine AS2-Konfiguration finden Sie unter. Einrichtung einer AS2-Konfiguration

Themen

- Erstellen Sie einen AS2-Server mit der Transfer Family Family-Konsole
- Verwenden Sie eine Vorlage, um einen Demo-Stack der Transfer Family AS2 zu erstellen
- AS2-Konfigurationen und Kontingente
- AS2-Funktionen und -Fähigkeiten

## Erstellen Sie einen AS2-Server mit der Transfer Family Family-Konsole

In diesem Verfahren wird erklärt, wie Sie mithilfe der Transfer Family Family-Konsole einen AS2-fähigen Server erstellen. Wenn Sie AWS CLI stattdessen den verwenden möchten, finden Sie weitere Informationen unter. the section called “Schritt 2: Erstellen Sie einen Transfer Family Family-Server, der das AS2-Protokoll verwendet”

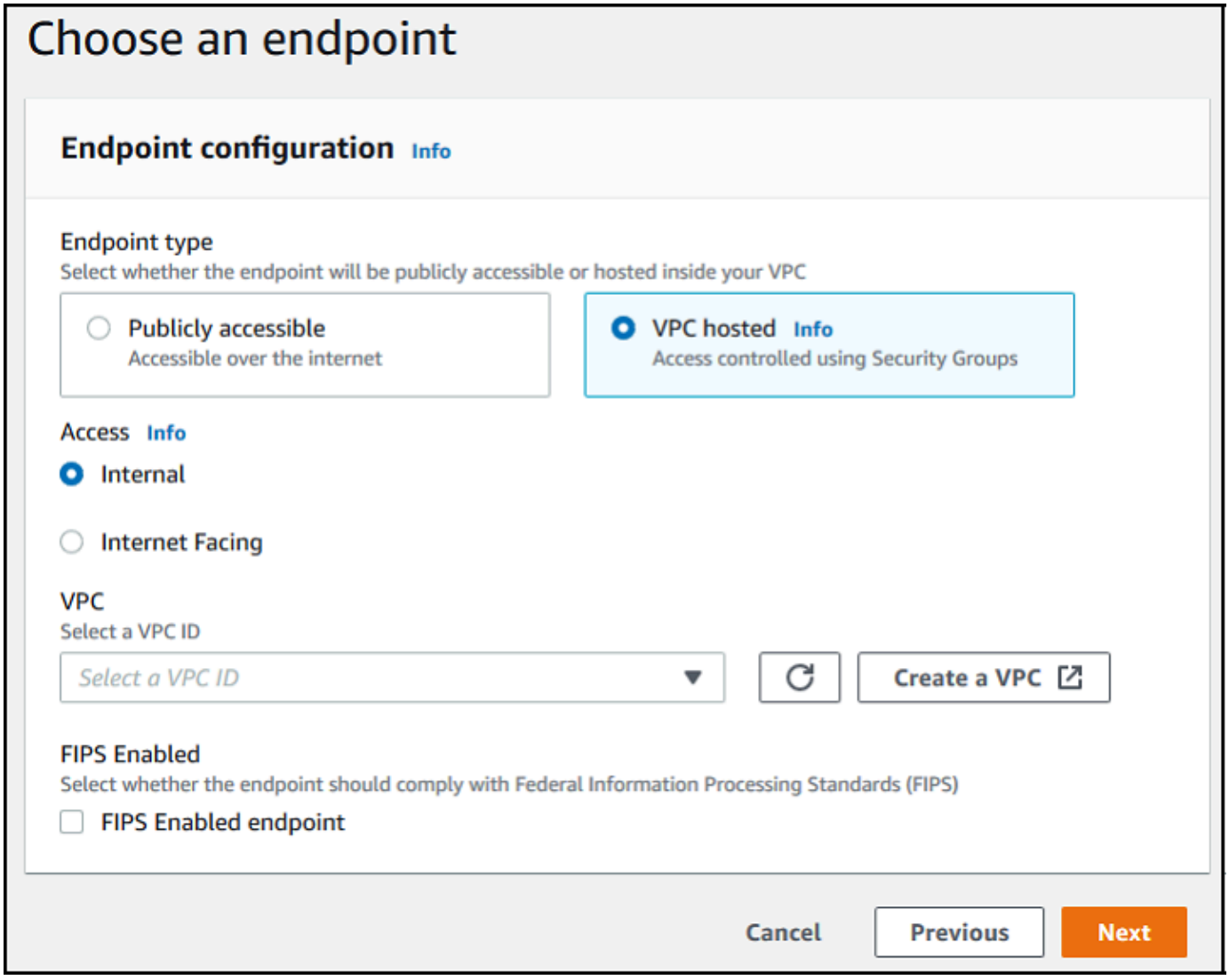
So erstellen Sie einen AS2-fähigen Server

1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Wählen Sie im linken Navigationsbereich Server und dann Server erstellen aus.
3. Wählen Sie auf der Seite „Protokolle auswählen“ die Option AS2 (Applicability Statement 2) und dann Weiter aus.

4. Wählen Sie auf der Seite „Identitätsanbieter auswählen“ die Option Weiter aus.

> **Note**
>
> Für AS2 können Sie keinen Identitätsanbieter auswählen, da die Standardauthentifizierung für das AS2-Protokoll nicht unterstützt wird. Stattdessen kontrollieren Sie den Zugriff über Virtual Private Cloud (VPC) -Sicherheitsgruppen.

5. Gehen Sie auf der Seite „Endpunkt auswählen“ wie folgt vor:




a. Wählen Sie als Endpunkttyp die Option VPC Hosted aus, um den Endpunkt Ihres Servers zu hosten. Informationen zur Einrichtung Ihres VPC-gehosteten Endpunkts finden Sie unter. Erstellen Sie einen Server in einer virtuellen privaten Cloud

> **Note**
>
> Öffentlich zugängliche Endpunkte werden für das AS2-Protokoll nicht unterstützt. Um Ihren VPC-Endpunkt über das Internet zugänglich zu machen, wählen Sie Internet Facing unter Access und geben Sie dann Ihre Elastic IP-Adressen ein.

b. Wählen Sie für Access eine der folgenden Optionen:

- Intern — Wählen Sie diese Option, um den Zugriff innerhalb Ihrer VPC und VPC-verbundenen Umgebungen bereitzustellen, z. B. über ein lokales Rechenzentrum oder VPN. AWS Direct Connect
- Internet Facing — Wählen Sie diese Option, um den Zugriff über das Internet und innerhalb Ihrer VPC- und VPC-verbundenen Umgebungen bereitzustellen, z. B. ein lokales Rechenzentrum über oder VPN. AWS Direct Connect

  Wenn Sie sich für Internet Facing entscheiden, geben Sie Ihre Elastic IP-Adressen ein, wenn Sie dazu aufgefordert werden.

c. Wählen Sie für VPC entweder eine vorhandene VPC aus oder wählen Sie Create VPC, um eine neue VPC zu erstellen.

d. Lassen Sie für FIPS Enabled das Kontrollkästchen FIPS Enabled Endpoint deaktiviert.

> **Note**
>
> FIPS-fähige Endpunkte werden für das AS2-Protokoll nicht unterstützt.

e. Wählen Sie Weiter aus.

6. Wählen Sie auf der Seite „Domain auswählen“ Amazon S3 aus, um Ihre Dateien mithilfe des ausgewählten Protokolls als Objekte zu speichern und darauf zuzugreifen.

   Wählen Sie Weiter aus.

7. Wählen Sie auf der Seite Zusätzliche Details konfigurieren die Einstellungen aus, die Sie benötigen.

> **Note**
>
> Wenn Sie zusammen mit AS2 weitere Protokolle konfigurieren, gelten alle zusätzlichen Detaileinstellungen. Für das AS2-Protokoll gelten jedoch nur die Einstellungen in den Abschnitten CloudWatch Protokollierung und Tags.
> Auch wenn die Einrichtung einer CloudWatch Protokollierungsrolle optional ist, empfehlen wir dringend, sie so einzurichten, dass Sie den Status Ihrer Nachrichten einsehen und Konfigurationsprobleme beheben können.

8. Überprüfen Sie auf der Seite Überprüfen und erstellen Ihre Einstellungen, um sicherzustellen, dass sie korrekt sind.

   - Wenn Sie eine Ihrer Einstellungen bearbeiten möchten, wählen Sie neben dem Schritt, den Sie ändern möchten, die Option Bearbeiten aus.

     > **Note**
     >
     > Wenn Sie einen Schritt bearbeiten, empfehlen wir Ihnen, jeden Schritt nach dem Schritt, den Sie bearbeiten möchten, zu überprüfen.

   - Wenn Sie keine Änderungen vorgenommen haben, wählen Sie Server erstellen, um Ihren Server zu erstellen. Sie gelangen zur Seite Servers (Server) (siehe unten), auf der der neue Server aufgelistet ist.

     Es kann mehrere Minuten dauern, bis sich der Status Ihres neuen Servers auf Online ändert. Ab diesem Zeitpunkt kann der Server Dateioperationen für die Benutzer ausführen.

## Verwenden Sie eine Vorlage, um einen Demo-Stack der Transfer Family AS2 zu erstellen

Wir liefern eine eigenständige AWS CloudFormation Vorlage, mit der Sie schnell einen AS2-fähigen Transfer Family Family-Server erstellen können. Die Vorlage konfiguriert den Server mit einem öffentlichen Amazon VPC-Endpunkt, Zertifikaten, lokalen Profilen und Partnerprofilen, einer Vereinbarung und einem Connector.

Bevor Sie diese Vorlage verwenden, sollten Sie Folgendes beachten:

- Wenn Sie anhand dieser Vorlage einen Stack erstellen, werden Ihnen die verwendeten AWS Ressourcen in Rechnung gestellt.
- Die Vorlage erstellt mehrere Zertifikate und platziert sie AWS Secrets Manager , um sie sicher zu speichern. Sie können diese Zertifikate bei Bedarf aus Secrets Manager löschen, da Ihnen die Nutzung dieses Dienstes in Rechnung gestellt wird. Durch das Löschen dieser Zertifikate in Secrets Manager werden sie nicht vom Transfer Family Family-Server gelöscht. Daher wird die Funktionalität des Demo-Stacks nicht beeinträchtigt. Für Zertifikate, die Sie mit einem AS2-Produktionsserver verwenden möchten, möchten Sie jedoch möglicherweise Secrets Manager verwenden, um Ihre gespeicherten Zertifikate zu verwalten und regelmäßig zu rotieren.
- Wir empfehlen, die Vorlage nur als Grundlage und hauptsächlich zu Demonstrationszwecken zu verwenden. Wenn Sie diesen Demo-Stack in der Produktion verwenden möchten, empfehlen wir Ihnen, den YAML-Code der Vorlage zu ändern, um einen robusteren Stack zu erstellen. Erstellen Sie beispielsweise Zertifikate auf Produktionsebene und erstellen Sie eine AWS Lambda Funktion, die Sie in der Produktion verwenden können.

So erstellen Sie einen AS2-fähigen Transfer Family Family-Server aus einer Vorlage CloudFormation

1. Öffnen Sie die AWS CloudFormation Konsole unter https://console.aws.amazon.com/cloudformation.
2. Wählen Sie im linken Navigationsbereich Stack aus.
3. Wählen Sie Create stack (Stack erstellen) und dann With new resources (standard) (Mit neuen Ressourcen (Standard)).
4. Wählen Sie im Abschnitt Voraussetzung — Vorlage vorbereiten die Option Vorlage ist bereit aus.
5. Kopieren Sie diesen Link, die AS2-Demo-Vorlage, und fügen Sie ihn in das Amazon S3 S3-URL-Feld ein.
6. Wählen Sie Weiter aus.
7. Geben Sie auf der Seite „Stack-Details angeben“ Ihrem Stack einen Namen und geben Sie dann die folgenden Parameter an:
   - Geben Sie unter AS2 Werte für Lokale AS2-ID und Partner-AS2-ID ein, oder akzeptieren Sie die Standardwerte und. `local` `partner`
   - Geben Sie unter Netzwerk einen Wert für die CIDR-IP des Sicherheitsgruppeneingangs ein, oder akzeptieren Sie die Standardeinstellung. `0.0.0.0/0`

> **Note**
>
> Dieser Wert im CIDR-Format gibt an, welche IP-Adressen für eingehenden Datenverkehr zum AS2-Server zulässig sind. Der Standardwert,`0.0.0.0/0`, erlaubt alle IP-Adressen.

- Geben Sie unter Allgemein einen Wert für Präfix ein, oder akzeptieren Sie den Standardwert`transfer-as2`. Dieses Präfix steht vor allen Ressourcennamen, die vom Stack erstellt werden. Wenn Sie beispielsweise das Standardpräfix verwenden, wird Ihr Amazon S3 S3-Bucket benannt`transfer-as2-`*`TransferS3BucketName`*.

8. Wählen Sie Weiter aus. Wählen Sie auf der Seite „Stack-Optionen konfigurieren“ erneut Weiter aus.
9. Überprüfen Sie die Details für den Stack, den Sie gerade erstellen, und wählen Sie dann Stapel erstellen aus.

> **Note**
>
> Unten auf der Seite müssen Sie unter Funktionen angeben, dass dadurch AWS CloudFormation möglicherweise Ressourcen AWS Identity and Access Management (IAM) erstellt werden.

Nachdem der Stack erstellt wurde, können Sie mithilfe von AWS Command Line Interface (AWS CLI) eine AS2-Testnachricht vom Partnerserver an Ihren lokalen Transfer Family Family-Server senden. Ein AWS CLI Beispielbefehl zum Senden einer Testnachricht wird zusammen mit allen anderen Ressourcen im Stack erstellt.

Um diesen Beispielbefehl zu verwenden, gehen Sie zur Registerkarte Outputs Ihres Stacks und kopieren Sie den TransferExampleAs2Command. Sie können den Befehl dann mit dem AWS CLI ausführen. Falls Sie das noch nicht installiert haben AWS CLI, finden Sie weitere Informationen unter Installieren oder Aktualisieren der neuesten Version von AWS CLI im AWS Command Line Interface Benutzerhandbuch.

Der Beispielbefehl hat das folgende Format:

```
aws s3api put-object --bucket TransferS3BucketName --key test.txt && aws transfer
 start-file-transfer --region aws-region --connector-id TransferConnectorId --send-
file-paths /TransferS3BucketName/test.txt
```

> **Note**
>
> Ihre Version dieses Befehls enthält die tatsächlichen Werte für die *TransferConnectorId* Ressourcen *TransferS3BucketName* und in Ihrem Stack.

Dieser Beispielbefehl besteht aus zwei separaten Befehlen, die mithilfe der && Zeichenfolge miteinander verkettet sind.

Der erste Befehl erstellt eine neue, leere Textdatei in Ihrem Bucket:

```
aws s3api put-object --bucket TransferS3BucketName --key test.txt
```

Dann verwendet der zweite Befehl den Connector, um die Datei vom Partnerprofil an das lokale Profil zu senden. Auf dem Transfer Family Family-Server wurde eine Vereinbarung eingerichtet, die es dem lokalen Profil ermöglicht, Nachrichten vom Partnerprofil anzunehmen.

```
aws transfer start-file-transfer --region aws-region --connector-id TransferConnectorId
 --send-file-paths /TransferS3BucketName/test.txt
```

Nachdem Sie den Befehl ausgeführt haben, können Sie zu Ihrem Amazon S3 S3-Bucket (*TransferS3BucketName*) wechseln und den Inhalt anzeigen. Wenn der Befehl erfolgreich ist, sollten Sie die folgenden Objekte in Ihrem Bucket sehen:

- `processed/`— Dieser Ordner enthält eine JSON-Datei, die die übertragene Datei und die MDN-Antwort beschreibt.
- `processing/`— Dieser Ordner enthält vorübergehend Dateien, während sie verarbeitet werden. Nach Abschluss einer Übertragung sollte dieser Ordner jedoch leer sein.
- *server-id*/— Dieser Ordner ist nach Ihrer Transfer Family Family-Server-ID benannt. Er enthält `from-`*partner* (dieser Ordner wird dynamisch benannt, basierend auf der AS2-ID des Partners), der wiederum `processing/` Ordner `failed/processed/`, und enthält. Der /*server-id*/`from-`*partner*`/processed/` Ordner enthält eine Kopie der übertragenen Textdatei und die entsprechenden JSON- und MDN-Dateien.

- `test.txt`— Dieses Objekt ist die (leere) Datei, die übertragen wurde.

# AS2-Konfigurationen und Kontingente

In diesem Thema werden die unterstützten Konfigurationen, Funktionen und Funktionen für Übertragungen beschrieben, die das AS2-Protokoll (Applicability Statement 2) verwenden, einschließlich der akzeptierten Chiffren und Digests. In diesem Abschnitt werden auch die Beschränkungen und bekannten Probleme für AS2-Übertragungen beschrieben.

Themen


- Von AS2 unterstützte Konfigurationen
- AS2-Kontingente und Einschränkungen


## Von AS2 unterstützte Konfigurationen

Signierung, Verschlüsselung, Komprimierung, MDN

Sowohl für eingehende als auch für ausgehende Übertragungen sind die folgenden Elemente entweder erforderlich oder optional:

- Verschlüsselung — Erforderlich (für HTTP-Transport, die einzige derzeit unterstützte Transportmethode). Unverschlüsselte Nachrichten werden nur akzeptiert, wenn sie von einem TLS-terminierenden Proxy wie einem Application Load Balancer (ALB) weitergeleitet werden und der Header vorhanden ist. `X-Forwarded-Proto: https`
- Signieren — optional
- Komprimierung — Optional (der einzige derzeit unterstützte Komprimierungsalgorithmus ist ZLIB)
- Hinweis zur Disposition von Nachrichten (MDN) — Optional

Chiffren

Die folgenden Verschlüsselungen werden sowohl für eingehende als auch für ausgehende Übertragungen unterstützt:

- AES128_CBC
- AES192_CBC
- AES256_CBC

- 3DES (nur aus Gründen der Abwärtskompatibilität)

Zusammenfassungen

Die folgenden Digests werden unterstützt:

- Eingehendes Signieren und MDN — SHA1, SHA256, SHA384, SHA512
- Ausgehendes Signieren und MDN — SHA1, SHA256, SHA384, SHA512

MDN

Für MDN-Antworten werden bestimmte Typen wie folgt unterstützt:

- Eingehende Übertragungen — Synchron und asynchron
- Ausgehende Übertragungen — Nur synchron
- Simple Mail Transfer Protocol (SMTP) (E-Mail MDN) — Wird nicht unterstützt

Transporte

- Eingehende Übertragungen — HTTP ist der einzige derzeit unterstützte Transport, und Sie müssen ihn explizit angeben.

  > **Note**
  >
  > Wenn Sie HTTPS für eingehende Übertragungen verwenden müssen, können Sie TLS auf einem Application Load Balancer oder einem Network Load Balancer beenden. Dies wird unter beschrieben. Empfangen Sie AS2-Nachrichten über HTTPS

- Ausgehende Übertragungen — Wenn Sie eine HTTP-URL angeben, müssen Sie auch einen Verschlüsselungsalgorithmus angeben. Wenn Sie eine HTTPS-URL angeben, haben Sie die Möglichkeit, NONE für Ihren Verschlüsselungsalgorithmus anzugeben.

## AS2-Kontingente und Einschränkungen

In diesem Abschnitt werden Kontingente und Einschränkungen für AS2 beschrieben

Themen

- AS2-Kontingente
- Kontingente für den Umgang mit Geheimnissen
- Bekannte Beschränkungen

## AS2-Kontingente

Die folgenden Kontingente gelten für AS2-Dateiübertragungen. Informationen zur Beantragung einer Erhöhung für ein anpassbares Kontingent finden Sie unter AWS-Service Kontingente im Allgemeine AWS-Referenz.

AS2-Kontingente

| Name | Standard | Anpassbar |
|---|---|---|
| Maximale Anzahl an empfangenen eingehenden Dateien pro Sekunde | 100 | Nein |
| Maximale Anzahl ausgehender Dateien, die pro Sekunde gesendet werden | 100 | Nein |
| Maximale Anzahl gleichzeitig eingehender Dateien | 400 | Nein |
| Maximale Anzahl gleichzeitiger ausgehender Dateien | 400 | Nein |
| Maximale Größe der eingehenden Datei (unkomprimiert) | 1 GB | Nein |
| Maximale Größe der ausgehenden Datei (unkomprimiert) | 1 GB | Nein |
| Maximale Anzahl von Dateien pro ausgehender Anfrage | 10 | Nein |

| Name | Standard | Anpassbar |
| --- | --- | --- |
| Maximale Anzahl ausgehender Anfragen pro Sekunde | 100 | Nein |
| Maximale Anzahl eingehender Anfragen pro Sekunde | 100 | Nein |
| Maximale ausgehende Bandbreite pro Konto (ausgehende SFTP- und AS2-Anfragen tragen beide zu diesem Wert bei) | 50 MB pro Sekunde | Nein |
| Maximale Anzahl der Vereinbarungen pro Server | 100 | Ja |
| Maximale Anzahl von Anschlüssen pro Konto (sowohl SFTP- als auch AS2-Konnektoren tragen zu diesem Limit bei) | 100 | Ja |
| Maximale Anzahl von Zertifikaten pro Partnerprofil | 10 | Nein |
| Maximale Anzahl der Zertifikate pro Konto | 1000 | Ja |
| Maximale Anzahl von Partnerprofilen pro Konto | 1000 | Ja |

Kontingente für den Umgang mit Geheimnissen

AWS Transfer Family ruft im Namen von AS2-Kunden AWS Secrets Manager an, die die Standardauthentifizierung verwenden. Zusätzlich ruft Secrets Manager auf AWS KMS.

> **Note**
>
> Diese Kontingente sind nicht spezifisch für Ihre Verwendung von Geheimnissen für Transfer Family: Sie werden von allen Diensten in Ihrem gemeinsam genutzt AWS-Konto.

Für Secrets Manager `GetSecretValue` gilt das Kontingent Kombinierte Rate von Anfragen DescribeSecret und GetSecretValue API-Anfragen, wie unter AWS Secrets Manager Kontingente beschrieben.

Secrets Manager **`GetSecretValue`**

| Name | Wert | Beschreibung |
|---|---|---|
| Kombinierte Rate von DescribeSecret und GetSecret Value API-Anfragen | Jede unterstützte Region: 10 000 pro Sekunde | Die maximale Anzahl an Transaktionen pro Sekunde für DescribeSecret GetSecretValue API-Anfragen zusammen. |

Für gelten AWS KMS die folgenden Kontingente für`Decrypt`. Einzelheiten finden Sie unter Kontingente für jeden AWS KMS API-Vorgang anfordern

AWS KMS **`Decrypt`**

| Kontingentname | Standardwert (Anforderungen pro Sekunde) |
|---|---|
| Anforderungsrate für kryptografische Operation en (symmetrisch) | Diese gemeinsamen Kontingente variieren je nach dem AWS-Region und der Art des in der Anfrage verwendeten AWS KMS Schlüssels. Jedes Kontingent wird separat berechnet.<br>• 5 500 (freigegeben)<br>• 10 000 (gemeinsam) in den folgenden Regionen:<br>  • USA Ost (Ohio), us-east-2<br>  • Asien-Pazifik (Singapur), ap-southeast-1<br>  • Asien-Pazifik (Sydney), ap-southeast-2 |

| Kontingentname | Standardwert (Anforderungen pro Sekunde) |
| --- | --- |
| | • Asien-Pazifik (Tokio), ap-northeast-1<br>• Europa (Frankfurt) eu-central-1<br>• Europa (London) eu-west-2<br>• 50 000 (gemeinsam) in den folgenden Regionen:<br>• USA Ost (Nord-Virginia), us-east-1<br>• USA West (Oregon), us-west-2<br>• Europa (Irland), eu-west-1 |
| Anforderungskontingente für benutzerdefinierte Schlüsselspeicher<br><br>**Note**<br>Dieses Kontingent gilt nur, wenn Sie einen externen Schlüsselspeicher verwenden. | Kontingente für benutzerdefinierte Schlüssel speicher-Anfragen werden für jeden benutzerd efinierten Schlüsselspeicher separat berechnet.<br>• 1.800 (gemeinsam genutzt) für jeden AWS CloudHSM Schlüsselspeicher<br>• 1 800 (freigegeben) für jeden externen Schlüsselspeicher. |

Bekannte Beschränkungen

- Serverseitiges TCP-Keep-Alive wird nicht unterstützt. Die Verbindung wird nach 350 Sekunden Inaktivität unterbrochen, es sei denn, der Client sendet Keep-Alive-Pakete.
- Damit eine aktive Vereinbarung vom Service akzeptiert wird und in den CloudWatch Amazon-Protokollen erscheint, müssen Nachrichten gültige AS2-Header enthalten.
- Der Server, der Nachrichten AWS Transfer Family für AS2 empfängt, muss das Schutzattribut des Algorithmus Cryptographic Message Syntax (CMS) zur Validierung von Nachrichtensignaturen unterstützen, wie in RFC 6211 definiert. Dieses Attribut wird in einigen älteren IBM Sterling-Produkten nicht unterstützt.
- Doppelte Nachrichten-IDs führen zu einer verarbeiteten Nachricht (Warnung: Doppeltes Dokument).
- Die Schlüssellänge für AS2-Zertifikate muss mindestens 2048 Bit und höchstens 4096 Bit betragen.

- Wenn AS2-Nachrichten oder asynchrone mDNS an den HTTPS-Endpunkt eines Handelspartners gesendet werden, müssen die Nachrichten oder mDNS ein gültiges SSL-Zertifikat verwenden, das von einer öffentlich vertrauenswürdigen Zertifizierungsstelle (CA) signiert ist. Selbstsignierte Zertifikate werden derzeit nur für ausgehende Übertragungen unterstützt.
- Der Endpunkt muss das TLS-Protokoll der Version 1.2 und einen kryptografischen Algorithmus unterstützen, der gemäß der Sicherheitsrichtlinie zulässig ist (wie unter beschrieben). Sicherheitsrichtlinien für AWS Transfer Family Server
- Mehrere Anlagen und Certificate Exchange Messaging (CEM) aus AS2 Version 1.2 werden derzeit nicht unterstützt.
- Die Standardauthentifizierung wird derzeit nur für ausgehende Nachrichten unterstützt.

## AS2-Funktionen und -Fähigkeiten

In den folgenden Tabellen sind die Funktionen und Fähigkeiten aufgeführt, die für Transfer Family Family-Ressourcen verfügbar sind, die AS2 verwenden.

### AS2-Funktionen

Transfer Family bietet die folgenden Funktionen für AS2.

| Funktion | Unterstützt von AWS Transfer Family |
| --- | --- |
| Drummond-Zertifizierung | Ja |
| AWS CloudFormation Unterstützung | Ja |
| CloudWatchAmazon-Metriken | Ja |
| Kryptografische SHA-2-Algorithmen | Ja |
| Support für Amazon S3 | Ja |
| Support für Amazon EFS | Nein |
| Geplante Nachrichten | Ja [1] |
| AWS Transfer Family Verwaltete Workflows | Nein |
| Zertifikatsaustausch-Messaging (CEM) | Nein |

| Funktion | Unterstützt von AWS Transfer Family |
| --- | --- |
| Gegenseitiges TLS (mTLS) | Nein |
| Support für selbstsignierte Zertifikate | Ja |

1. Ausgehende geplante Nachrichten sind über AWS Lambda Planungsfunktionen von Amazon verfügbar EventBridge

## AS2-Sende- und Empfangsfunktionen

Die folgende Tabelle enthält eine Liste der AWS Transfer Family AS2-Sende- und Empfangsfunktionen.

| Funktion | Eingehend: Empfangen mit dem Server | Ausgehend: Senden mit Connector |
| --- | --- | --- |
| TLS-verschlüsselter Transport (HTTPS) | Ja [1] | Ja |
| Nicht-TLS-Transport (HTTP) | Ja | Ja 2 |
| Synchrones MDN | Ja | Ja |
| Komprimierung von Nachrichten | Ja | Ja |
| Asynchrones MDN | Ja | Nein |
| Statische IP-Adresse | Ja | Ja |
| Bringen Sie Ihre eigene IP-Adresse mit | Ja | Nein |
| Mehrere Dateianhänge | Nein | Nein |
| Standardauthentifizierung | Nein | Ja |
| AS2 neu starten | Nicht zutreffend | Nein |

| Funktion | Eingehend: Empfangen mit dem Server | Ausgehend: Senden mit Connector |
|---|---|---|
| AS2-Zuverlässigkeit | Nein | Nein |
| Benutzerdefinierter Betreff pro Nachricht | Nicht zutreffend | Nein |

1. Eingehender TLS-verschlüsselter Transport mit Network Load Balancer (NLB) verfügbar

2. Ausgehender Nicht-TLS-Transport ist nur verfügbar, wenn die Verschlüsselung aktiviert ist

# AS2-Konnektoren konfigurieren

Der Zweck eines Connectors besteht darin, eine Beziehung zwischen Handelspartnern für ausgehende Übertragungen herzustellen, indem AS2-Dateien von einem Transfer Family Family-Server an ein externes, partnereigenes Ziel gesendet werden. Für den Connector geben Sie die lokale Partei, den Remote-Partner und deren Zertifikate an (indem Sie lokale Profile und Partnerprofile erstellen).

Sobald Sie einen Konnektor eingerichtet haben, können Sie Informationen an Ihre Handelspartner übertragen. Jedem AS2-Server werden drei statische IP-Adressen zugewiesen. AS2-Konnektoren verwenden diese IP-Adressen, um asynchrone mDNS über AS2 an Ihre Handelspartner zu senden.

> **Note**
>
> Die von einem Handelspartner empfangene Nachrichtengröße entspricht nicht der Objektgröße in Amazon S3. Diese Diskrepanz ist darauf zurückzuführen, dass die AS2-Nachricht die Datei vor dem Senden in einen Umschlag verpackt. Daher kann sich die Dateigröße erhöhen, auch wenn die Datei komprimiert gesendet wird. Stellen Sie daher sicher, dass die maximale Dateigröße des Handelspartners größer ist als die Größe der Datei, die Sie senden.

## Erstellen Sie einen AS2-Connector

In diesem Verfahren wird erklärt, wie AS2-Konnektoren mithilfe der AWS Transfer Family Konsole erstellt werden. Wenn Sie AWS CLI stattdessen den verwenden möchten, finden Sie weitere

Informationen unterthe section called “Schritt 6: Stellen Sie eine Verbindung zwischen Ihnen und Ihrem Partner her”.

So erstellen Sie einen AS2-Connector

1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Wählen Sie im linken Navigationsbereich Connectors und dann Create Connector aus.
3. Geben Sie im Abschnitt Connector-Konfiguration die folgenden Informationen an:

   - URL — Geben Sie die URL für ausgehende Verbindungen ein.
   - Zugriffsrolle — Wählen Sie den Amazon-Ressourcennamen (ARN) der zu AWS Identity and Access Management verwendenden (IAM) -Rolle aus. Stellen Sie sicher, dass diese Rolle Lese- und Schreibzugriff auf das übergeordnete Verzeichnis des Dateispeicherorts bietet, der in der `StartFileTransfer` Anfrage verwendet wird. Stellen Sie außerdem sicher, dass die Rolle Lese- und Schreibzugriff auf das übergeordnete Verzeichnis der Dateien bietet, die Sie mit versenden möchten`StartFileTransfer`.

     > **Note**
     >
     > Wenn Sie die Standardauthentifizierung für Ihren Connector verwenden, benötigt die Zugriffsrolle die `secretsmanager:GetSecretValue` Erlaubnis für den geheimen Schlüssel. Wenn das Geheimnis mithilfe eines vom Kunden verwalteten Schlüssels anstelle von Von AWS verwalteter Schlüssel PIN verschlüsselt wird AWS Secrets Manager, benötigt die Rolle auch die `kms:Decrypt` Erlaubnis für diesen Schlüssel. Wenn Sie Ihr Geheimnis mit dem Präfix benennen`aws/transfer/`, können Sie die erforderliche Berechtigung mit einem Platzhalterzeichen (`*`) hinzufügen, wie im Beispiel für eine Berechtigung zum Erstellen von Geheimnissen gezeigt.

   - Rolle für die Protokollierung (optional) — Wählen Sie die IAM-Rolle aus, die der Connector verwenden soll, um Ereignisse in Ihre CloudWatch Logs zu übertragen.
4. Wählen Sie im Abschnitt AS2-Konfiguration die lokalen Profile und Partnerprofile, die Verschlüsselungs- und Signierungsalgorithmen aus und legen Sie fest, ob die übertragenen Informationen komprimiert werden sollen. Beachten Sie Folgendes:

   - Wählen Sie für den Verschlüsselungsalgorithmus nur, `DES_EDE3_CBC` wenn Sie einen Legacy-Client unterstützen müssen, der ihn benötigt, da es sich um einen schwachen Verschlüsselungsalgorithmus handelt.

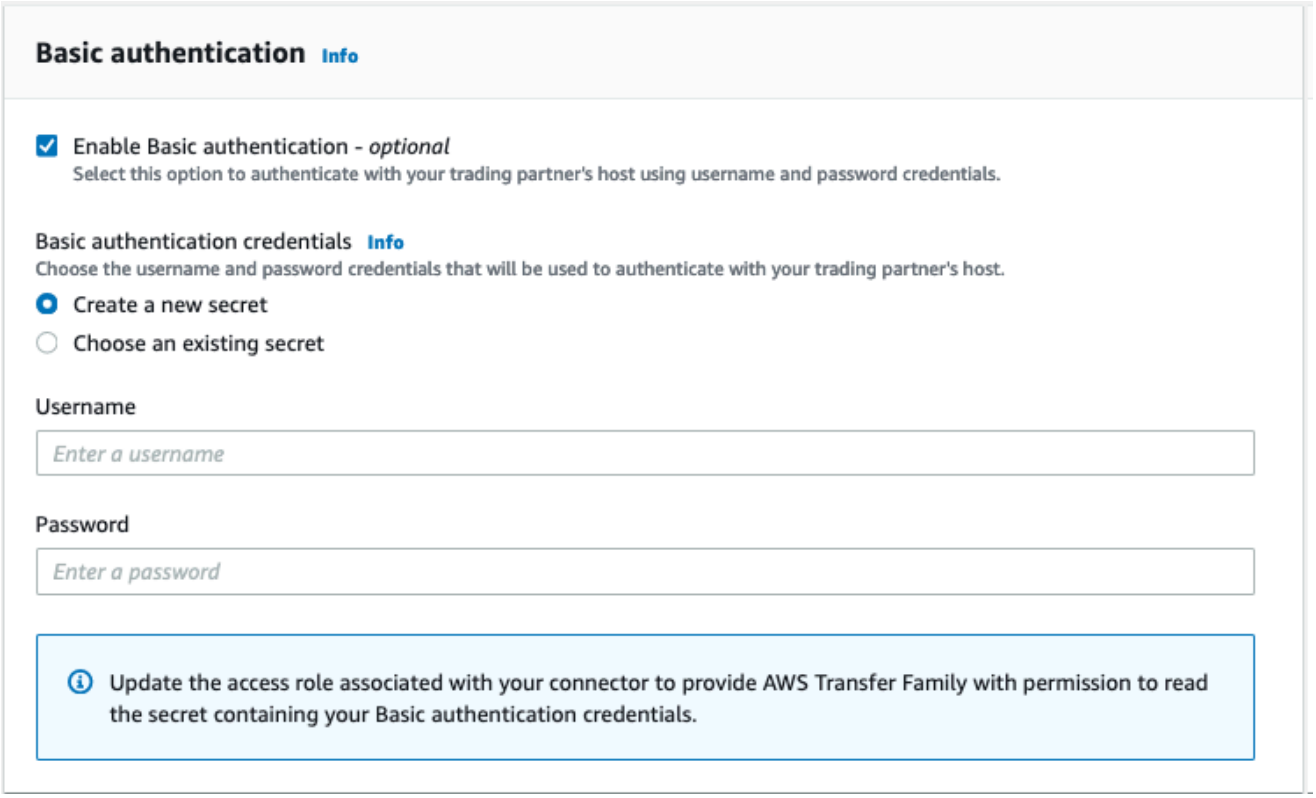
- Der Betreff wird als `subject` HTTP-Header-Attribut in AS2-Nachrichten verwendet, die mit dem Connector gesendet werden.
- Wenn Sie einen Connector ohne Verschlüsselungsalgorithmus erstellen möchten, müssen Sie dies `HTTPS` als Protokoll angeben.

5. Geben Sie im Abschnitt MDN-Konfiguration die folgenden Informationen an:

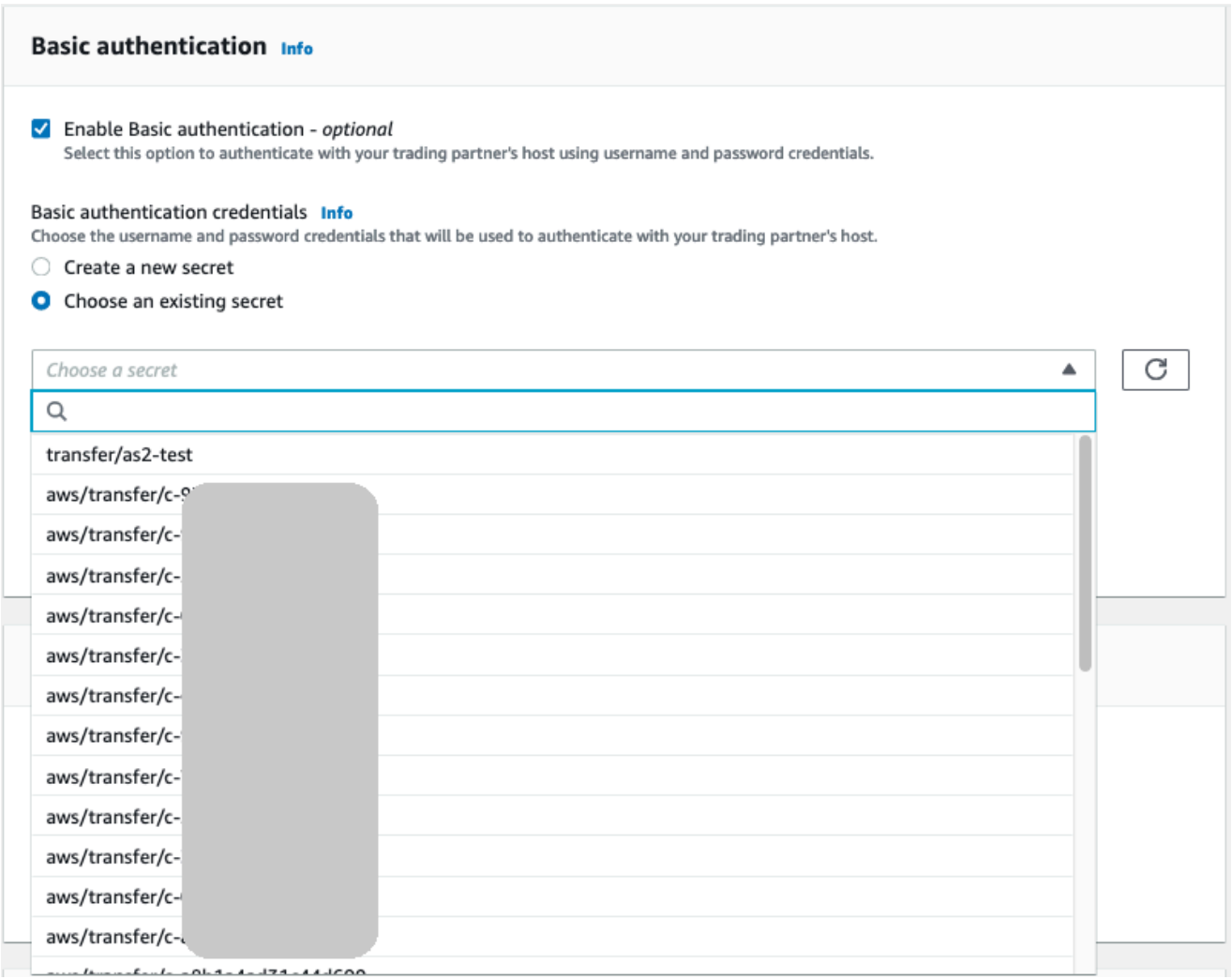
   - MDN anfordern — Sie haben die Möglichkeit, von Ihrem Handelspartner zu verlangen, dass er Ihnen eine MDN sendet, nachdem er Ihre Nachricht erfolgreich über AS2 erhalten hat.
   - Signiertes MDN — Sie haben die Möglichkeit, zu verlangen, dass mDNS signiert wird. Diese Option ist nur verfügbar, wenn Sie MDN anfordern ausgewählt haben.

6. Geben Sie im Abschnitt Standardauthentifizierung die folgenden Informationen an.

   - Um Anmeldeinformationen zusammen mit ausgehenden Nachrichten zu senden, wählen Sie Standardauthentifizierung aktivieren aus. Wenn Sie keine Anmeldeinformationen zusammen mit ausgehenden Nachrichten senden möchten, lassen Sie die Option Standardauthentifizierung aktivieren deaktiviert.
   - Wenn Sie die Authentifizierung verwenden, wählen oder erstellen Sie ein Geheimnis.
     - Um ein neues Geheimnis zu erstellen, wählen Sie Neues Geheimnis erstellen und geben Sie dann einen Benutzernamen und ein Passwort ein. Diese Anmeldeinformationen müssen mit dem Benutzer übereinstimmen, der eine Verbindung zum Endpunkt des Partners herstellt.




- Um ein vorhandenes Geheimnis zu verwenden, wählen Sie „Bestehendes Geheimnis auswählen“ und wählen Sie dann ein Geheimnis aus dem Dropdownmenü aus. Einzelheiten zur Erstellung eines korrekt formatierten Secrets in Secrets Manager finden Sie unterAktivieren Sie die Standardauthentifizierung für AS2-Konnektoren.




7. Nachdem Sie alle Ihre Einstellungen bestätigt haben, wählen Sie Connector erstellen, um den Connector zu erstellen.

Die Seite Connectors wird angezeigt, auf der die ID Ihres neuen Connectors zur Liste hinzugefügt wurde. Einzelheiten zu Ihren Konnektoren finden Sie unterDetails zum AS2-Konnektor anzeigen.

## Algorithmen für AS2-Konnektoren

Wenn Sie einen AS2-Connector erstellen, werden die folgenden Sicherheitsalgorithmen an den Connector angehängt.

| Typ | Algorithmus |
| --- | --- |
| TLS-Chiffre | TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 |

| Typ | Algorithmus |
| --- | --- |
| | TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256<br><br>TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256<br><br>TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256<br><br>TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384<br><br>TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384<br><br>TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384<br><br>TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 |

## Standardauthentifizierung für AS2-Konnektoren

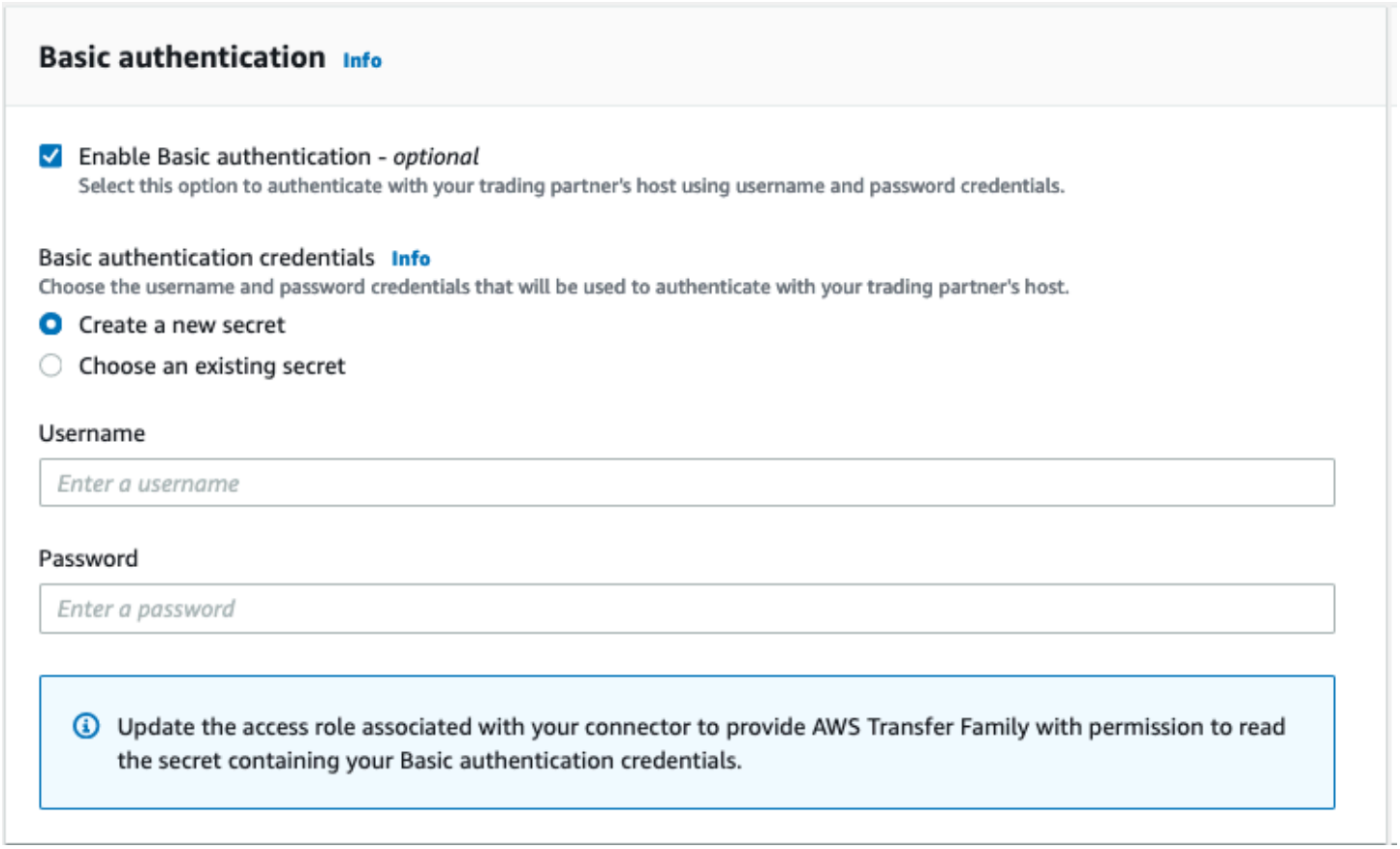
Wenn Sie einen Transfer Family Family-Server erstellen oder aktualisieren, der das AS2-Protokoll verwendet, können Sie die Standardauthentifizierung für ausgehende Nachrichten hinzufügen. Dazu fügen Sie einem Connector Authentifizierungsinformationen hinzu.

> **Note**
>
> Die Standardauthentifizierung ist nur verfügbar, wenn Sie HTTPS verwenden.

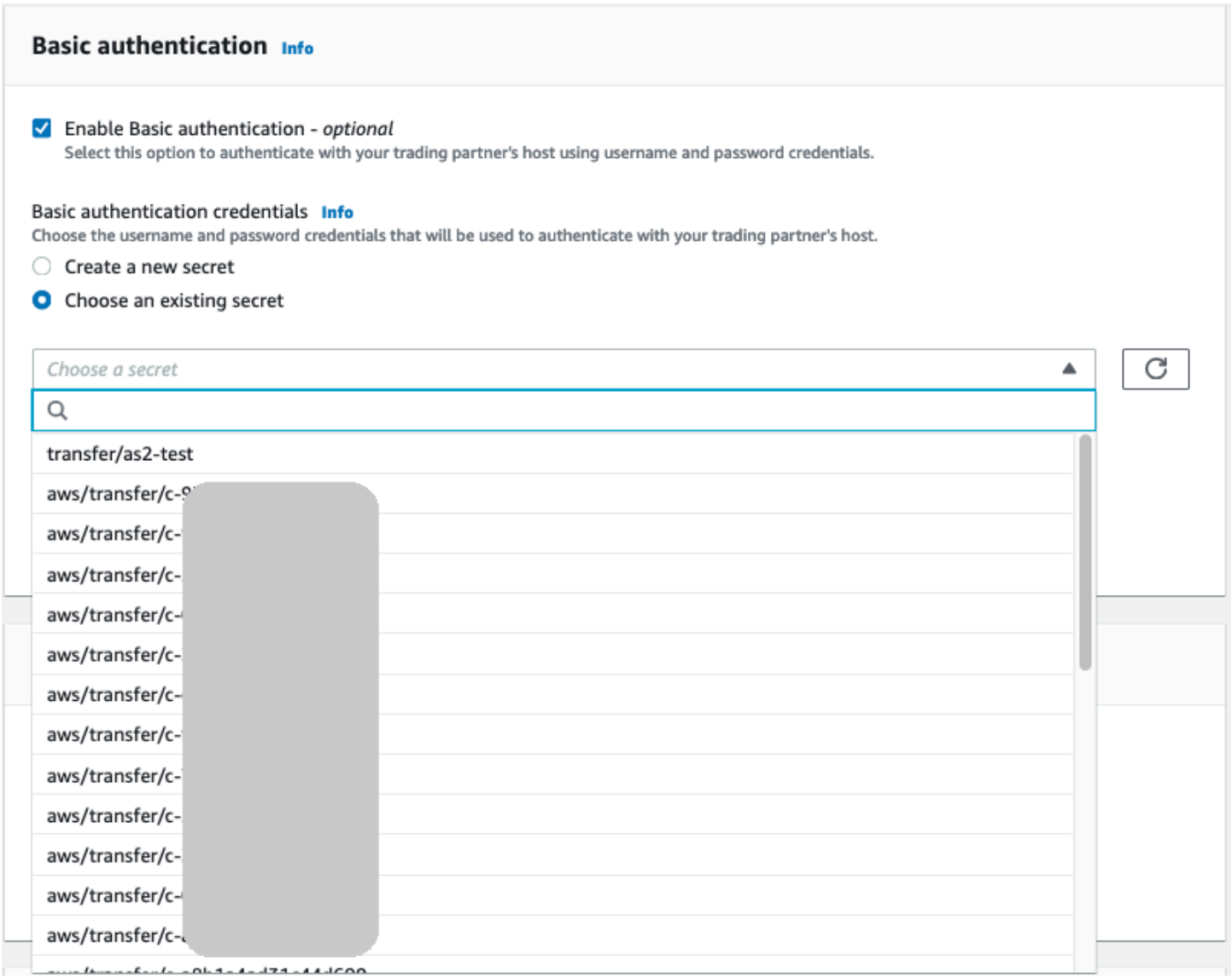
Um die Authentifizierung für Ihren Connector zu verwenden, wählen Sie im Abschnitt Standardauthentifizierung die Option Standardauthentifizierung aktivieren aus. Nachdem Sie die Standardauthentifizierung aktiviert haben, können Sie wählen, ob Sie ein neues Geheimnis erstellen oder ein vorhandenes verwenden möchten. In beiden Fällen werden die Anmeldeinformationen im Secret zusammen mit ausgehenden Nachrichten gesendet, die diesen Connector verwenden. Die

Anmeldeinformationen müssen mit dem Benutzer übereinstimmen, der versucht, eine Verbindung zum Remote-Endpunkt des Handelspartners herzustellen.

Der folgende Screenshot zeigt, wie „Standardauthentifizierung aktivieren“ und „Neuen geheimen Schlüssel erstellen“ ausgewählt sind. Nachdem Sie diese Optionen getroffen haben, können Sie einen Benutzernamen und ein Passwort für das Geheimnis eingeben.




Der folgende Screenshot zeigt, wie „Standardauthentifizierung aktivieren“ und „Vorhandenes Geheimnis auswählen“ ausgewählt sind. Ihr Geheimnis muss das richtige Format haben, wie unter beschriebenAktivieren Sie die Standardauthentifizierung für AS2-Konnektoren.




# Aktivieren Sie die Standardauthentifizierung für AS2-Konnektoren

Wenn Sie die Standardauthentifizierung für AS2-Connectoren aktivieren, können Sie entweder ein neues Geheimnis in der Transfer Family Family-Konsole erstellen oder ein Geheimnis verwenden, das Sie in AWS Secrets Manager erstellen. In beiden Fällen wird Ihr Geheimnis im Secrets Manager gespeichert.

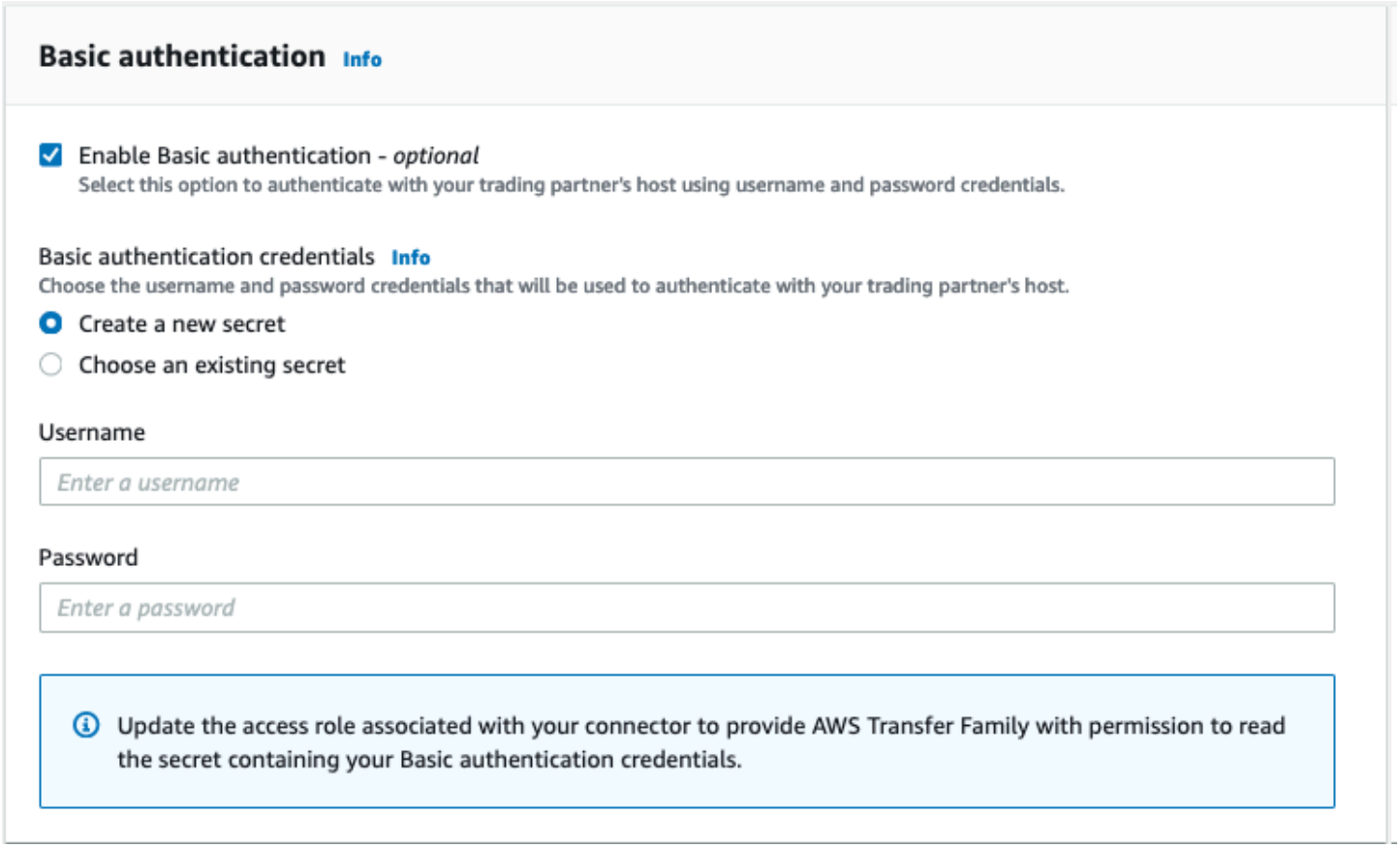
Themen

- Erstellen Sie ein neues Geheimnis in der Konsole
- Verwenden eines vorhandenen -Secrets
- Erstellen Sie ein Geheimnis in AWS Secrets Manager

## Erstellen Sie ein neues Geheimnis in der Konsole

Wenn Sie einen Connector in der Konsole erstellen, können Sie ein neues Geheimnis erstellen.

Um ein neues Geheimnis zu erstellen, wählen Sie Neues Geheimnis erstellen und geben Sie dann einen Benutzernamen und ein Passwort ein. Diese Anmeldeinformationen müssen mit dem Benutzer übereinstimmen, der eine Verbindung zum Endpunkt des Partners herstellt.

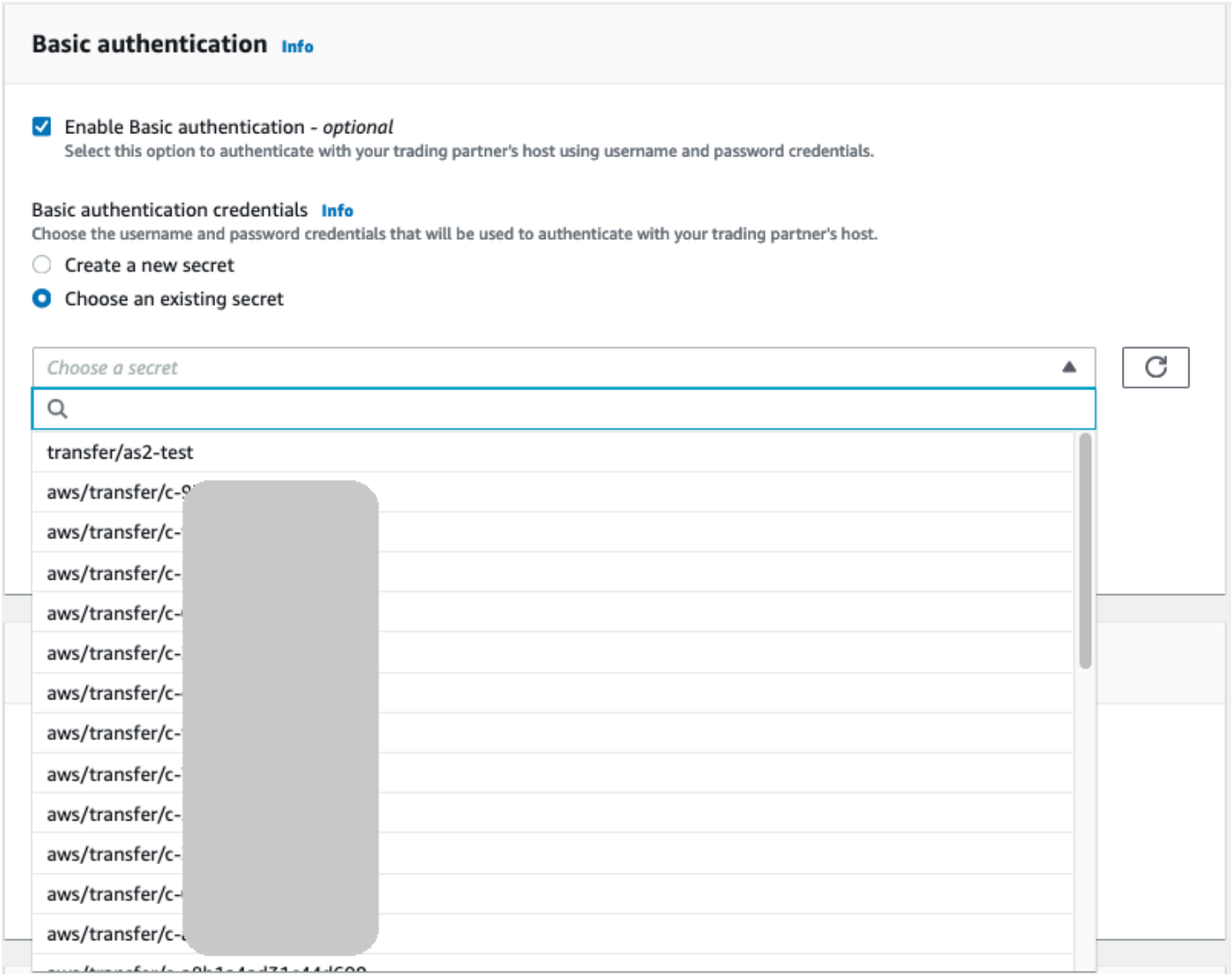



> **Note**
>
> Wenn Sie in der Konsole ein neues Geheimnis erstellen, folgt der Name des Geheimnisses dieser Benennungskonvention:**`/aws/transfer/`*`connector-id`***, wobei *`Connector-ID`* die ID des Connectors ist, den Sie erstellen. Beachten Sie dies, wenn Sie versuchen, das Geheimnis in zu finden. AWS Secrets Manager

## Verwenden eines vorhandenen -Secrets

Wenn Sie einen Connector in der Konsole erstellen, können Sie ein vorhandenes Geheimnis angeben.

Um ein vorhandenes Geheimnis zu verwenden, wählen Sie Choose an existing secret und wählen Sie dann ein Secret aus dem Dropdownmenü aus. Einzelheiten zur Erstellung eines korrekt formatierten Secrets in Secrets Manager finden Sie unterErstellen Sie ein Geheimnis in AWS Secrets Manager.




## Erstellen Sie ein Geheimnis in AWS Secrets Manager

Das folgende Verfahren beschreibt, wie Sie ein geeignetes Geheimnis für die Verwendung mit Ihrem AS2-Connector erstellen.

> **Note**
>
> Die Standardauthentifizierung ist nur verfügbar, wenn Sie HTTPS verwenden.

Um Benutzeranmeldeinformationen in Secrets Manager für die AS2 Basic-Authentifizierung zu speichern

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die AWS Secrets Manager Konsole unter https://console.aws.amazon.com/secretsmanager/.
2. Wählen Sie im linken Navigationsbereich Secrets aus.
3. Wählen Sie auf der Seite Secrets die Option Neues Geheimnis speichern aus.
4. Wählen Sie auf der Seite Geheimtyp auswählen für Geheimtyp die Option Anderer Geheimtyp aus.
5. Wählen Sie im Abschnitt Schlüssel/Wert-Paare die Registerkarte Schlüssel/Wert aus.
   - Schlüssel — Geben Sie ein. **`Username`**
   - Wert — Geben Sie den Namen des Benutzers ein, der berechtigt ist, eine Verbindung zum Server des Partners herzustellen.
6. Wenn Sie ein Passwort angeben möchten, wählen Sie Zeile hinzufügen und wählen Sie im Abschnitt Schlüssel/Wert-Paare die Registerkarte Schlüssel/Wert.

   Wählen Sie Zeile hinzufügen und wählen Sie im Abschnitt Schlüssel/Wert-Paare die Registerkarte Schlüssel/Wert-Paare aus.
   - Schlüssel — Geben Sie ein. **`Password`**
   - Wert — Geben Sie das Passwort für den Benutzer ein.
7. Wenn Sie einen privaten Schlüssel angeben möchten, wählen Sie Zeile hinzufügen und wählen Sie im Abschnitt Schlüssel/Wert-Paare die Registerkarte Schlüssel/Wert.
   - Schlüssel — Geben Sie ein. **`PrivateKey`**
   - Wert — Geben Sie einen privaten Schlüssel für den Benutzer ein. Dieser Wert muss im OpenSSH-Format gespeichert werden und dem öffentlichen Schlüssel entsprechen, der für diesen Benutzer auf dem Remoteserver gespeichert ist.
8. Wählen Sie Weiter aus.

9. Geben Sie auf der Seite Geheimen Schlüssel konfigurieren einen Namen und eine Beschreibung für Ihr Geheimnis ein. Wir empfehlen, **`aws/transfer/`** für den Namen das Präfix von zu verwenden. Sie könnten beispielsweise Ihr Geheimnis benennen**`aws/transfer/connector-1`**.
10. Wählen Sie Weiter und akzeptieren Sie dann die Standardeinstellungen auf der Seite Rotation konfigurieren. Wählen Sie anschließend Weiter.
11. Wählen Sie auf der Seite „Überprüfen“ die Option Speichern aus, um das Geheimnis zu erstellen und zu speichern.

Nachdem Sie das Geheimnis erstellt haben, können Sie es beim Erstellen eines Connectors auswählen (sieheAS2-Konnektoren konfigurieren). Wählen Sie in dem Schritt, in dem Sie die Standardauthentifizierung aktivieren, das Geheimnis aus der Dropdownliste der verfügbaren Geheimnisse aus.

## Details zum AS2-Konnektor anzeigen

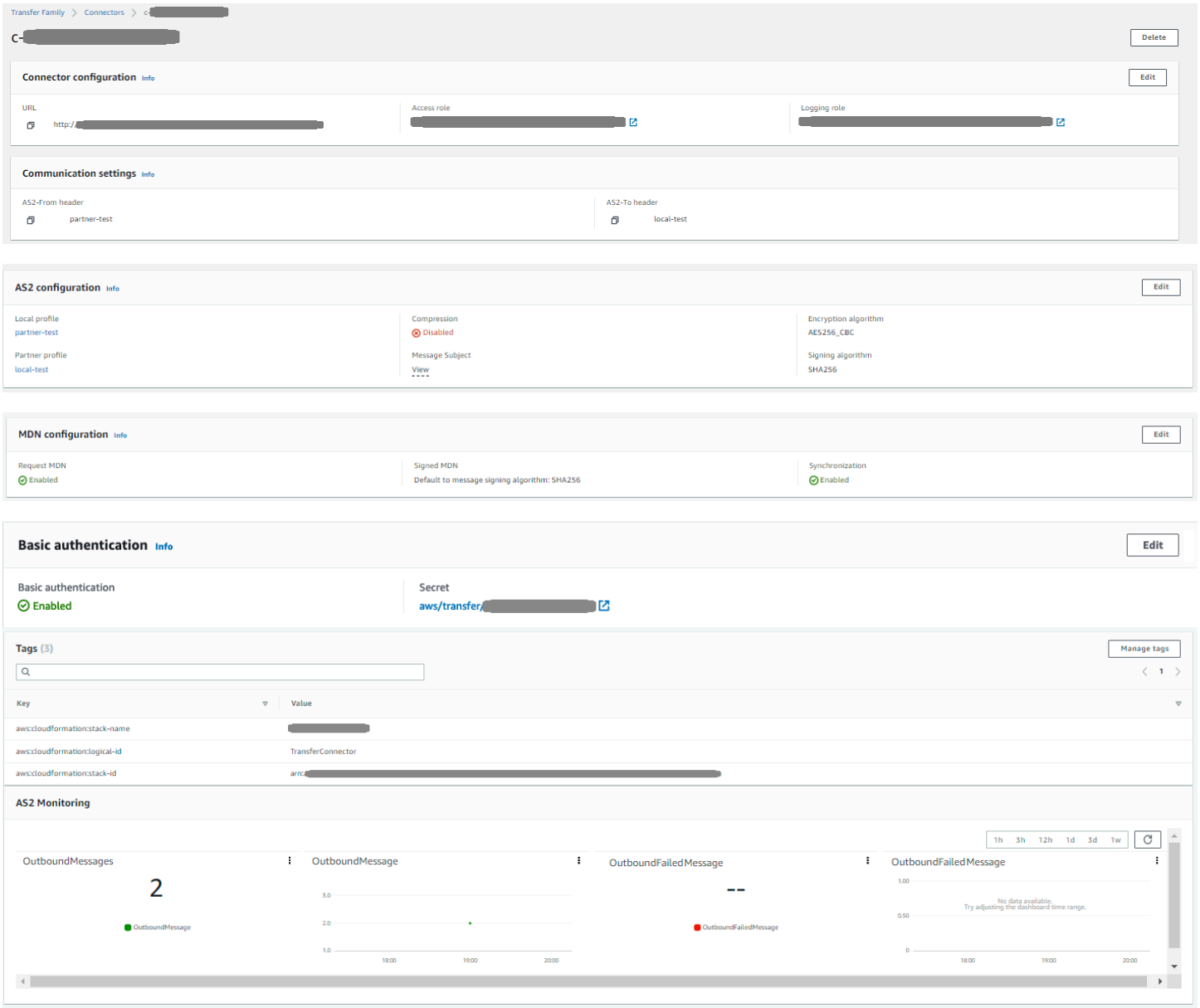
In der AWS Transfer Family Konsole finden Sie eine Liste mit Details und Eigenschaften für einen AWS Transfer Family AS2-Connector. Zu den Eigenschaften eines AS2-Connectors gehören seine URL, Rollen, Profile, mDNs, Tags und Überwachungsmetriken.

Dies ist das Verfahren zum Anzeigen von Konnektordetails.

So zeigen Sie die Konnektordetails an

1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Wählen Sie im linken Navigationsbereich die Option Connectors aus.
3. Wählen Sie den Bezeichner in der Spalte Connector-ID aus, um die Detailseite für den ausgewählten Connector aufzurufen.

Sie können die Eigenschaften für den AS2-Connector auf der Detailseite des Connectors ändern, indem Sie Bearbeiten wählen.




> **Note**
>
> Sie können viele dieser Informationen, wenn auch in einem anderen Format, abrufen, indem Sie den folgenden Befehl ausführen AWS Command Line Interface ()AWS CLI :
>
> ```
> aws transfer describe-connector --connector-id your-connector-id
> ```
>
> Weitere Informationen finden Sie DescribeConnectorin der API-Referenz.

# AS2-Partner verwalten

In diesem Thema wird die Verwaltung von AS2-Zertifikaten, -Profilen und -Vereinbarungen beschrieben.

## AS2-Zertifikate importieren

Der Transfer Family AS2-Prozess verwendet Zertifikatsschlüssel sowohl für die Verschlüsselung als auch für die Signierung der übertragenen Informationen. Partner können für beide Zwecke denselben Schlüssel oder für jeden einen separaten Schlüssel verwenden. Wenn Sie über gemeinsame Verschlüsselungsschlüssel verfügen, die von einem vertrauenswürdigen Drittanbieter treuhänderisch aufbewahrt werden, sodass Daten im Notfall oder bei einer Sicherheitsverletzung entschlüsselt werden können, empfehlen wir, separate Signaturschlüssel zu verwenden. Durch die Verwendung separater Signaturschlüssel (die Sie nicht hinterlegen) gefährden Sie nicht die Funktionen Ihrer digitalen Signaturen, die nicht zurückgewiesen werden können.

> **Note**
>
> Die Schlüssellänge für AS2-Zertifikate muss mindestens 2048 Bit und höchstens 4096 Bit betragen.

In den folgenden Punkten wird detailliert beschrieben, wie AS2-Zertifikate während des Prozesses verwendet werden.

- Eingehender AS2-Versand
  - Der Handelspartner sendet seinen öffentlichen Schlüssel für das Signaturzertifikat, und dieser Schlüssel wird in das Partnerprofil importiert.
  - Die lokale Partei sendet den öffentlichen Schlüssel für ihre Verschlüsselungs- und Signaturzertifikate. Der Partner importiert dann den oder die privaten Schlüssel. Die lokale Partei kann separate Zertifikatsschlüssel zum Signieren und Verschlüsseln senden oder denselben Schlüssel für beide Zwecke verwenden.
- Ausgehender AS2
  - Der Partner sendet den öffentlichen Schlüssel für sein Verschlüsselungszertifikat, und dieser Schlüssel wird in das Partnerprofil importiert.
  - Die lokale Partei sendet den öffentlichen Schlüssel für das Zertifikat zum Signieren und importiert den privaten Schlüssel des Zertifikats zum Signieren.

- Wenn Sie HTTPS verwenden, können Sie ein selbstsigniertes Transport Layer Security (TLS) - Zertifikat importieren.

Einzelheiten zum Erstellen von Zertifikaten finden Sie unterthe section called “Schritt 1: Zertifikate für AS2 erstellen”.

In diesem Verfahren wird erklärt, wie Zertifikate mithilfe der Transfer Family Family-Konsole importiert werden. Wenn Sie AWS CLI stattdessen das verwenden möchten, finden Sie weitere Informationen unterthe section called “Schritt 3: Zertifikate als Transfer Family Family-Zertifikatsressourcen importieren”.

So geben Sie ein AS2-fähiges Zertifikat an

1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Wählen Sie im linken Navigationsbereich unter AS2 Trading Partners die Option Certificates aus.
3. Wählen Sie Import certificate (Zertifikat importieren).
4. Geben Sie im Abschnitt Beschreibung des Zertifikats einen leicht identifizierbaren Namen für das Zertifikat ein. Stellen Sie sicher, dass Sie den Zweck des Zertifikats anhand seiner Beschreibung identifizieren können. Wählen Sie außerdem die Rolle für das Zertifikat aus.
5. Geben Sie im Abschnitt Inhalt des Zertifikats ein öffentliches Zertifikat eines Handelspartners oder die öffentlichen und privaten Schlüssel für ein lokales Zertifikat an.
6. Wählen Sie im Abschnitt Verwendung des Zertifikats den Zweck für dieses Zertifikat aus. Es kann zur Verschlüsselung, Signierung oder für beides verwendet werden.

   > **Note**
   >
   > Wenn Sie Verschlüsselung und Signierung für die Verwendung wählen, erstellt Transfer Family zwei identische Zertifikate (jedes hat seine eigene ID): eines mit einem Verwendungswert von `ENCRYPTION` und eines mit einem Nutzungswert von`SIGNING`.

7. Füllen Sie den Abschnitt Inhalt des Zertifikats mit den entsprechenden Details aus.

   - Wenn Sie Selbstsigniertes Zertifikat wählen, geben Sie die Zertifikatskette nicht an.
   - Fügen Sie den Inhalt des Zertifikats ein.
   - Wenn es sich bei dem Zertifikat nicht um ein selbstsigniertes Zertifikat handelt, geben Sie die Zertifikatskette an.

- Wenn es sich bei diesem Zertifikat um ein lokales Zertifikat handelt, fügen Sie seinen privaten Schlüssel ein.

8. Wählen Sie Zertifikat importieren, um den Vorgang abzuschließen und die Details für das importierte Zertifikat zu speichern.

> **Note**
>
> TLS-Zertifikate können nur als öffentliches Zertifikat eines Partners importiert werden. Wenn Sie Öffentliches Zertifikat von einem Partner und dann Transport Layer Security (TLS) für die Verwendung auswählen, erhalten Sie eine Warnung. Außerdem müssen TLS-Zertifikate selbstsigniert sein (d. h. Sie müssen Self Signed Certificate auswählen, um ein TLS-Zertifikat zu importieren).

## Rotation der AS2-Zertifikate

Oft sind Zertifikate für einen Zeitraum von sechs Monaten bis zu einem Jahr gültig. Möglicherweise haben Sie Profile eingerichtet, die Sie für einen längeren Zeitraum beibehalten möchten. Um dies zu erleichtern, bietet Transfer Family eine Zertifikatsrotation. Sie können mehrere Zertifikate für ein Profil angeben, sodass Sie das Profil mehrere Jahre lang verwenden können. Transfer Family verwendet Zertifikate zum Signieren (optional) und Verschlüsseln (verpflichtend). Sie können ein einzelnes Zertifikat für beide Zwecke angeben, wenn Sie möchten.

Bei der Zertifikatsrotation wird ein altes, ablaufendes Zertifikat durch ein neueres Zertifikat ersetzt. Die Umstellung erfolgt schrittweise, um zu vermeiden, dass Übertragungen unterbrochen werden, wenn ein Vertragspartner noch kein neues Zertifikat für ausgehende Übertragungen konfiguriert hat oder während eines Zeitraums, in dem möglicherweise auch ein neueres Zertifikat verwendet wird, Payloads sendet, die mit einem alten Zertifikat signiert oder verschlüsselt sind. Die Zwischenzeit, in der sowohl alte als auch neue Zertifikate gültig sind, wird als Karenzzeit bezeichnet.

X.509-Zertifikate haben ein `Not Before` `Not After` Datum. Diese Parameter bieten Administratoren jedoch möglicherweise nicht genügend Kontrolle. Transfer Family bietet `Active Date` `Inactive Date` Einstellungen, mit denen gesteuert werden kann, welches Zertifikat für ausgehende Payloads verwendet und welches für eingehende Payloads akzeptiert wird.

Bei der Auswahl ausgehender Zertifikate wird der Höchstwert verwendet, der vor dem Datum der Übertragung liegt. `Inactive Date` Eingehende Prozesse akzeptieren Zertifikate im Bereich von `Not Before` `Not After` und und. `Active Date` `Inactive Date`

In der folgenden Tabelle wird eine Möglichkeit beschrieben, zwei Zertifikate für ein einzelnes Profil zu konfigurieren.

Zwei Zertifikate im Wechsel

| Name | NOT BEFORE(wird von der Zertifizierungsstelle kontrolliert) | ACTIVE DATE(gesetzt von Transfer Family) | INACTIVE DATE(gesetzt von Transfer Family) | NOT AFTER(von der Zertifizierungsstelle festgelegt) |
|---|---|---|---|---|
| Cert1 (älteres Zertifikat) | 2019-11-01 | 01.01.2020 | 31.12.2020 | 2024-01-01 |
| Cert2 (neueres Zertifikat) | 01.11.2020 | 2020-06-01 | 01.06.2021 | 01.01.2025 |

Beachten Sie Folgendes:

- Wenn Sie ein `Active Date` und `Inactive Date` für ein Zertifikat angeben, muss der Bereich innerhalb des Bereichs zwischen und liegen. `Not Before` `Not After`
- Es wird empfohlen, mehrere Zertifikate für jedes Profil zu konfigurieren und dabei sicherzustellen, dass der aktive Datumsbereich für alle Zertifikate zusammen den Zeitraum abdeckt, für den Sie das Profil verwenden möchten.
- Wir empfehlen Ihnen, eine Übergangszeit zwischen dem Zeitpunkt, zu dem Ihr älteres Zertifikat inaktiv wird, und dem Zeitpunkt, zu dem Ihr neueres Zertifikat aktiv wird, festzulegen. Im vorherigen Beispiel wird das erste Zertifikat erst am 31.12.2020 inaktiv, während das zweite Zertifikat am 01.06.2020 aktiv wird, was eine Kulanzzeit von 6 Monaten bietet. Im Zeitraum vom 01.06.2020 bis 31.12.2020 sind beide Zertifikate aktiv.

## Erstellen Sie AS2-Profile

Gehen Sie wie folgt vor, um sowohl lokale Profile als auch Partnerprofile zu erstellen. In diesem Verfahren wird erklärt, wie AS2-Profile mithilfe der Transfer Family Family-Konsole erstellt werden. Wenn Sie AWS CLI stattdessen das verwenden möchten, finden Sie weitere Informationen unterthe section called “Schritt 4: Erstellen Sie Profile für Sie und Ihren Handelspartner”.

So erstellen Sie ein AS2-Profil

1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Wählen Sie im linken Navigationsbereich unter AS2 Trading Partners die Option Profile und dann Create profile aus.
3. Geben Sie im Abschnitt Profilkonfiguration die AS2-ID für das Profil ein. Dieser Wert wird für die für das AS2-Protokoll spezifischen HTTP-Header `as2-from` und `as2-to` zur Identifizierung der Handelspartnerschaft verwendet, die bestimmt, welche Zertifikate verwendet werden sollen usw.
4. Wählen Sie im Abschnitt Profiltyp die Option Lokales Profil oder Partnerprofil aus.
5. Wählen Sie im Abschnitt Zertifikate ein oder mehrere Zertifikate aus dem Drop-down-Menü aus.

   > **Note**
   >
   > Wenn Sie ein Zertifikat importieren möchten, das nicht im Dropdownmenü aufgeführt ist, wählen Sie Neues Zertifikat importieren aus. Dadurch wird ein neues Browserfenster auf dem Bildschirm Zertifikat importieren geöffnet. Informationen zum Importieren von Zertifikaten finden Sie unterAS2-Zertifikate importieren.

6. (Optional) Geben Sie im Abschnitt Tags ein oder mehrere Schlüssel-Wert-Paare an, um dieses Profil leichter identifizieren zu können.
7. Wählen Sie Profil erstellen, um den Vorgang abzuschließen und das neue Profil zu speichern.

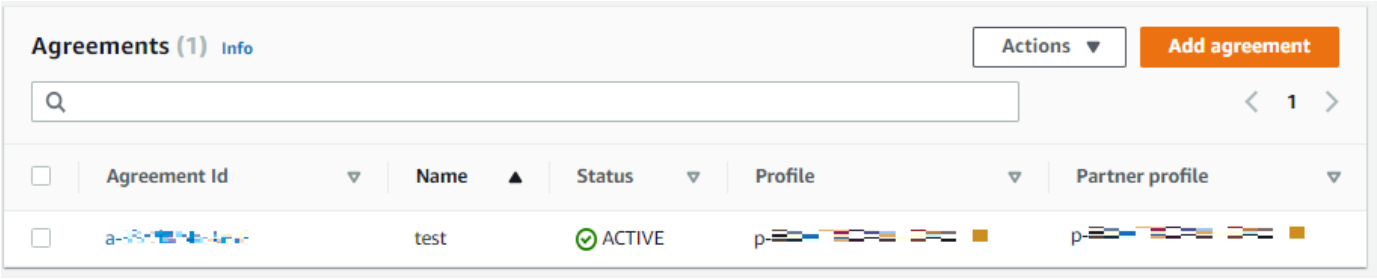
## Erstellen Sie AS2-Vereinbarungen

Vereinbarungen sind mit Transfer Family Family-Servern verknüpft. Sie spezifizieren die Details für Geschäftspartner, die das AS2-Protokoll verwenden, um Nachrichten oder Dateien mithilfe von Transfer Family auszutauschen, für eingehende Übertragungen, d. h. das Senden von AS2-Dateien von einer externen, partnereigenen Quelle an einen Transfer Family Family-Server.

In diesem Verfahren wird erklärt, wie AS2-Vereinbarungen mithilfe der Transfer Family Family-Konsole erstellt werden. Wenn Sie AWS CLI stattdessen das verwenden möchten, finden Sie weitere Informationen unterthe section called “Schritt 5: Erstellen Sie eine Vereinbarung zwischen Ihnen und Ihrem Partner”.

So erstellen Sie eine Vereinbarung für einen Transfer Family Family-Server

1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Wählen Sie im linken Navigationsbereich Server und dann einen Server aus, der das AS2-Protokoll verwendet.
3. Scrollen Sie auf der Seite mit den Serverdetails nach unten zum Abschnitt Vereinbarungen.




4. Wählen Sie Vereinbarung hinzufügen.
5. Geben Sie die Vereinbarungsparameter wie folgt ein:

   a. Geben Sie im Abschnitt Vereinbarungskonfiguration einen aussagekräftigen Namen ein. Stellen Sie sicher, dass Sie den Zweck der Vereinbarung anhand des Namens identifizieren können. Legen Sie außerdem den Status für die Vereinbarung fest: entweder Aktiv (standardmäßig ausgewählt) oder Inaktiv.

   b. Wählen Sie im Abschnitt Kommunikationskonfiguration ein lokales Profil und ein Partnerprofil aus.

   c. Wählen Sie im Abschnitt Konfiguration des Posteingangsordners einen Amazon S3 S3-Bucket zum Speichern eingehender Dateien und eine IAM-Rolle aus, die auf den Bucket zugreifen kann. Optional können Sie ein Präfix (Ordner) eingeben, das zum Speichern von Dateien im Bucket verwendet werden soll.

      Wenn Sie beispielsweise für Ihren Bucket und **`DOC-EXAMPLE-BUCKET`** **`incoming`** für Ihr Präfix eingeben, werden Ihre eingehenden Dateien in dem `/DOC-EXAMPLE-BUCKET/incoming` Ordner gespeichert.

   d. (Optional) Fügen Sie im Abschnitt „Tags“ Tags hinzu.

e. Nachdem Sie alle Informationen für die Vereinbarung eingegeben haben, wählen Sie Vereinbarung erstellen aus.

Die neue Vereinbarung wird im Abschnitt Vereinbarungen der Serverdetailseite angezeigt.

# Senden und Empfangen von AS2-Nachrichten

In diesem Abschnitt werden die Prozesse zum Senden und Empfangen von AS2-Nachrichten beschrieben. Es enthält auch Einzelheiten zu Dateinamen und Speicherorten, die mit AS2-Nachrichten verknüpft sind.

In der folgenden Tabelle sind die verfügbaren Verschlüsselungsalgorithmen für AS2-Nachrichten aufgeführt und es wird angegeben, wann Sie sie verwenden können.

| Verschlüsselungsalgorithmus | HTTP | HTTPS | Hinweise |
|---|---|---|---|
| AES128_CBC | Ja | Ja | |
| AES192_CBC | Ja | Ja | |
| AES256_CBC | Ja | Ja | |
| DES_EDE3_CBC | Ja | Ja | Verwenden Sie diesen Algorithmus nur, wenn Sie einen Legacy-Client unterstützen müssen, der ihn benötigt, da es sich um einen schwachen Verschlüsselungsalgorithmus handelt. |
| NONE | Nein | Ja | Wenn Sie Nachrichten an einen Transfer Family Family-Server senden, können Sie |

| Verschlüsselungsalgorithmus | HTTP | HTTPS | Hinweise |
| --- | --- | --- | --- |
| | | | nur auswählen, `NONE` ob Sie einen Application Load Balancer (ALB) verwenden. |

Themen

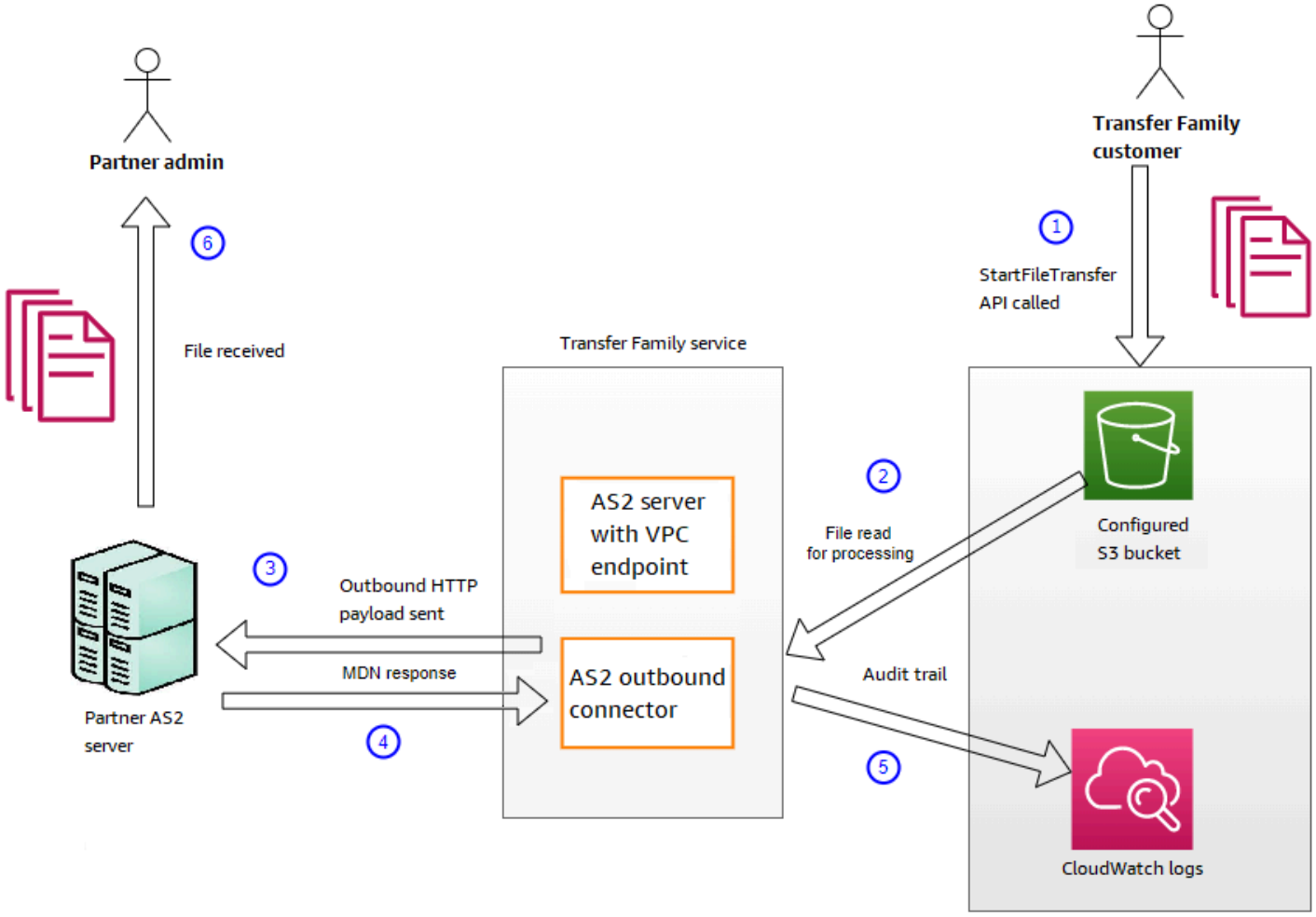
- AS2-Nachrichtenprozess senden
- Prozess zum Empfangen einer AS2-Nachricht
- Senden und Empfangen von AS2-Nachrichten über HTTPS
- Übertragung von Dateien mithilfe eines AS2-Connectors
- Dateinamen und Speicherorte
- Statuscodes
- JSON-Beispieldateien

## AS2-Nachrichtenprozess senden

Der ausgehende Prozess ist definiert als eine Nachricht oder Datei, die von AWS einem externen Client oder Dienst gesendet wird. Die Reihenfolge für ausgehende Nachrichten ist wie folgt:

1. Ein Administrator ruft den Befehl `start-file-transfer` AWS Command Line Interface (AWS CLI) oder die `StartFileTransfer` API-Operation auf. Diese Operation verweist auf eine `connector` Konfiguration.
2. Transfer Family erkennt eine neue Dateianfrage und lokalisiert die Datei. Die Datei ist komprimiert, signiert und verschlüsselt.
3. Ein Transfer-HTTP-Client führt eine HTTP-POST-Anfrage durch, um die Nutzdaten an den AS2-Server des Partners zu übertragen.
4. Der Prozess gibt die signierte MDN-Antwort entsprechend der HTTP-Antwort zurück (synchrones MDN).
5. Während sich die Datei zwischen den verschiedenen Übertragungsphasen bewegt, übermittelt der Prozess dem Kunden die MDN-Antwort, den Empfang und die Verarbeitungsdetails.

6. Der Remote-AS2-Server stellt die entschlüsselte und verifizierte Datei dem Partneradministrator zur Verfügung.



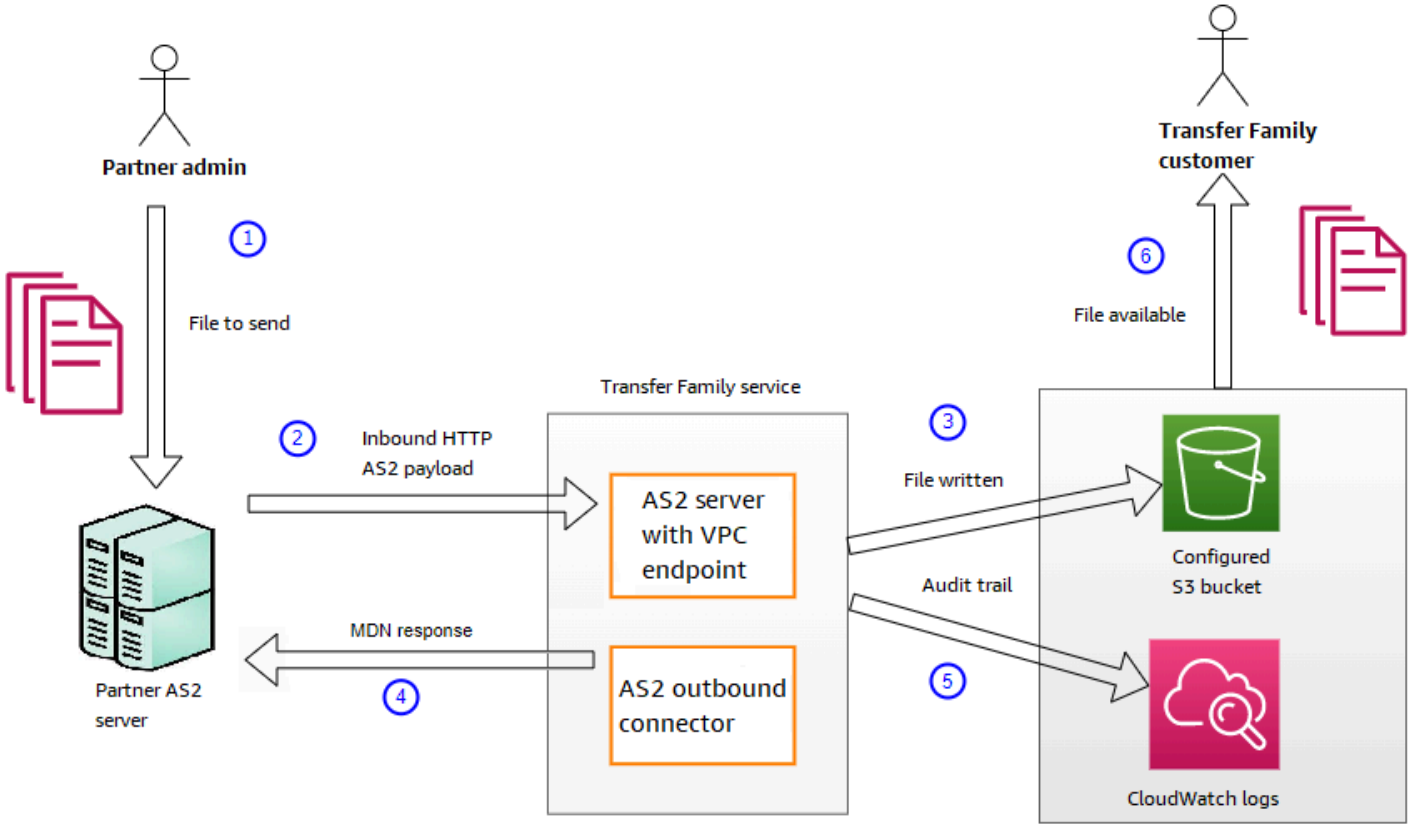

Die AS2-Verarbeitung unterstützt viele der RFC 4130-Protokolle, wobei der Schwerpunkt auf allgemeinen Anwendungsfällen und der Integration mit bestehenden AS2-fähigen Serverimplementierungen liegt. Einzelheiten zu den unterstützten Konfigurationen finden Sie unter. Von AS2 unterstützte Konfigurationen

## Prozess zum Empfangen einer AS2-Nachricht

Der eingehende Prozess ist als eine Nachricht oder Datei definiert, die auf Ihren AWS Transfer Family Server übertragen wird. Die Reihenfolge für eingehende Nachrichten ist wie folgt:

1. Ein Administrator- oder automatisierter Prozess startet eine AS2-Dateiübertragung auf dem AS2-Remote-Server des Partners.

2. Der Remote-AS2-Server des Partners signiert und verschlüsselt den Dateiinhalt und sendet dann eine HTTP-POST-Anforderung an einen AS2-Eingangsendpunkt, der auf Transfer Family gehostet wird.
3. Mithilfe der konfigurierten Werte für den Server, die Partner, Zertifikate und die Vereinbarung entschlüsselt und verifiziert Transfer Family die AS2-Payload. Der Dateiinhalt wird im konfigurierten Amazon S3 S3-Dateispeicher gespeichert.
4. Die signierte MDN-Antwort wird entweder direkt mit der HTTP-Antwort oder asynchron über eine separate HTTP-POST-Anfrage an den ursprünglichen Server zurückgegeben.
5. Ein Prüfprotokoll CloudWatch mit Einzelheiten zum Austausch wird an Amazon geschrieben.
6. Die entschlüsselte Datei ist in einem Ordner mit dem Namen `inbox/processed` verfügbar.




## Senden und Empfangen von AS2-Nachrichten über HTTPS

In diesem Abschnitt wird beschrieben, wie Sie einen Transfer Family Family-Server konfigurieren, der das AS2-Protokoll zum Senden und Empfangen von Nachrichten über HTTPS verwendet.

Themen

- Senden Sie AS2-Nachrichten über HTTPS
- Empfangen Sie AS2-Nachrichten über HTTPS

## Senden Sie AS2-Nachrichten über HTTPS

Um AS2-Nachrichten über HTTPS zu senden, erstellen Sie einen Connector mit den folgenden Informationen:

- Geben Sie für die URL eine HTTPS-URL an
- Wählen Sie für den Verschlüsselungsalgorithmus einen der verfügbaren Algorithmen aus.

> **Note**
>
> Um Nachrichten an einen Transfer Family Family-Server zu senden, ohne Verschlüsselung zu verwenden (d. h. Sie wählen `NONE` den Verschlüsselungsalgorithmus), müssen Sie einen Application Load Balancer (ALB) verwenden.

- Geben Sie die verbleibenden Werte für den Connector ein, wie unter beschrieben. AS2-Konnektoren konfigurieren

## Empfangen Sie AS2-Nachrichten über HTTPS

AWS Transfer Family AS2-Server bieten derzeit nur HTTP-Transport über Port 5080. Sie können TLS jedoch auf einem Netzwerk oder einem Application Load Balancer vor Ihrem Transfer Family Family-Server-VPC-Endpunkt beenden, indem Sie einen Port und ein Zertifikat Ihrer Wahl verwenden. Mit diesem Ansatz können Sie festlegen, dass eingehende AS2-Nachrichten HTTPS verwenden.

Voraussetzungen

- Die VPC muss sich auf demselben Server AWS-Region wie Ihr Transfer Family Family-Server befinden.
- Die Subnetze Ihrer VPC müssen sich innerhalb der Availability Zones befinden, in denen Sie Ihren Server verwenden möchten.

> **Note**
>
> Jeder Transfer Family Family-Server kann bis zu drei Availability Zones unterstützen.

- Weisen Sie bis zu drei Elastic IP-Adressen in derselben Region wie Ihr Server zu. Sie können sich auch dafür entscheiden, Ihren eigenen IP-Adressbereich (BYOIP) mitzubringen.

> **Note**
>
> Die Anzahl der Elastic IP-Adressen muss der Anzahl der Availability Zones entsprechen, die Sie mit Ihren Serverendpunkten verwenden.

Sie können entweder einen Network Load Balance (NLB) oder einen Application Load Balancer (ALB) konfigurieren. In der folgenden Tabelle sind die Vor- und Nachteile der einzelnen Ansätze aufgeführt.

In der folgenden Tabelle sind die Unterschiede in den Funktionen aufgeführt, wenn Sie einen NLB und einen ALB zum Beenden von TLS verwenden.

| Funktion | Network Load Balancer (NLB) | Application Load Balancer (ALB) |
|---|---|---|
| Latency | Geringere Latenz, da er auf Netzwerkebene arbeitet. | Höhere Latenz, da es auf der Anwendungsebene arbeitet. |
| Unterstützung für statische IPs | Kann elastische IP-Adressen anhängen, die statisch sein können. | Elastic IP-Adressen können nicht angehängt werden: Stellt eine Domain bereit, deren zugrunde liegende IP-Adressen sich ändern können. |
| Erweitertes Routing | Unterstützt kein erweitertes Routing. | Unterstützt erweitertes Routing. Kann den für AS2 erforderlichen `X-Forwarded-Proto` Header ohne Verschlüsselung einfügen.<br><br>Dieser Header wird in X-Forwarded-Proto auf der Website developer.mozilla.org beschrieben. |

| Funktion | Network Load Balancer (NLB) | Application Load Balancer (ALB) |
| --- | --- | --- |
| TLS/SSL-Kündigung | Unterstützt TLS/SSL-T erminierung | Unterstützt TLS/SSL-T erminierung |
| Gegenseitiges TLS (mTLS) | Transfer Family unterstützt derzeit nicht die Verwendung eines NLB für mTLS | Support für mTLS |

### Configure NLB

Dieses Verfahren beschreibt, wie Sie einen mit dem Internet verbundenen Network Load Balancer (NLB) in Ihrer VPC einrichten.

Um einen Network Load Balancer zu erstellen und den VPC-Endpunkt des Servers als Ziel des Load Balancers zu definieren

1. Öffnen Sie die Amazon Elastic Compute Cloud-Konsole unter https://console.aws.amazon.com/ec2/.
2. Wählen Sie im Navigationsbereich Load Balancers und dann Create Load Balancer aus.
3. Wählen Sie im Bereich Network Load Balancer die Option Erstellen.
4. Geben Sie im Abschnitt Grundkonfiguration die folgenden Informationen ein:
   - Geben Sie unter Name einen aussagekräftigen Namen für den Load Balancer ein.
   - Für Scheme, wählen Sie  Internet-facing.
   - Wählen Sie als IP-Adresstyp IPv4 aus.
5. Geben Sie im Abschnitt Netzwerkzuordnung die folgenden Informationen ein:
   - Wählen Sie für VPC die Virtual Private Cloud (VPC) aus, die Sie erstellt haben.
   - Wählen Sie unter Zuordnungen die Availability Zones aus, die den öffentlichen Subnetzen zugeordnet sind, die in derselben VPC verfügbar sind, die Sie mit Ihren Serverendpunkten verwenden.
   - Wählen Sie für die IPv4-Adresse jedes Subnetzes eine der Elastic IP-Adressen aus, die Sie zugewiesen haben.
6. Geben Sie im Abschnitt Listener und Routing die folgenden Informationen ein:

- Wählen Sie als Protokoll die Option TLS aus.
- Geben Sie im Feld Port **`5080`** ein.
- Wählen Sie für Standardaktion die Option Zielgruppe erstellen aus. Einzelheiten zum Erstellen einer neuen Zielgruppe finden Sie unterErstellen einer Zielgruppe.

Nachdem Sie eine Zielgruppe erstellt haben, geben Sie ihren Namen in das Feld Standardaktion ein.

7. Wählen Sie im Bereich Secure Listener Settings Ihr Zertifikat im Bereich Standard-SSL/TLS-Zertifikat aus.
8. Wählen Sie Create Load Balancer aus, um Ihren NLB zu erstellen.
9. (Optional, aber empfohlen) Aktivieren Sie die Zugriffsprotokolle für den Network Load Balancer, um einen vollständigen Audit-Trail zu führen, wie unter Zugriffsprotokolle für Ihren Network Load Balancer beschrieben.

   Wir empfehlen diesen Schritt, da die TLS-Verbindung im NLB beendet wird. Daher ist die Quell-IP-Adresse, die in Ihren CloudWatch AS2-Protokollgruppen von Transfer Family wiedergegeben wird, die private IP-Adresse der NLB und nicht die externe IP-Adresse Ihres Handelspartners.

Configure ALB

Dieses Verfahren beschreibt, wie Sie einen Application Load Balancer (NLB) in Ihrer VPC einrichten.

Um einen Application Load Balancer zu erstellen und den VPC-Endpunkt des Servers als Ziel des Load Balancers zu definieren

1. Öffnen Sie die Amazon Elastic Compute Cloud-Konsole unter https://console.aws.amazon.com/ec2/.
2. Wählen Sie im Navigationsbereich Load Balancers und dann Create Load Balancer aus.
3. Wählen Sie unter Application Load Balancer Create (Erstellen) aus.
4. Erstellen Sie in der ALB-Konsole einen neuen HTTP-Listener auf Port 443 (HTTPS).
5. (Optional). Wenn Sie die gegenseitige Authentifizierung (mTLS) einrichten möchten, konfigurieren Sie Sicherheitseinstellungen und einen Trust Store.

a. Hängen Sie Ihr SSL/TLS-Zertifikat an den Listener an.

b. Wählen Sie unter Behandlung von Client-Zertifikaten die Option Mutual Authentication (mTLS) aus.

c. Wählen Sie Mit Trust Store verifizieren aus.

d. Wählen Sie unter Erweiterte mTLS-Einstellungen einen Vertrauensspeicher aus oder erstellen Sie ihn, indem Sie Ihre CA-Zertifikate hochladen.

6. Erstellen Sie eine neue Zielgruppe und fügen Sie die privaten IP-Adressen Ihrer Transfer Family AS2-Serverendpunkte als Ziele auf Port 5080 hinzu. Einzelheiten zur Erstellung einer neuen Zielgruppe finden Sie unter. Erstellen einer Zielgruppe
7. Konfigurieren Sie Integritätsprüfungen für die Zielgruppe, um das TCP-Protokoll auf Port 5080 zu verwenden.
8. Erstellen Sie eine neue Regel, um HTTPS-Verkehr vom Listener an die Zielgruppe weiterzuleiten.
9. Konfigurieren Sie den Listener so, dass er Ihr SSL/TLS-Zertifikat verwendet.

Nachdem Sie den Load Balancer eingerichtet haben, kommunizieren die Clients mit dem Load Balancer über den benutzerdefinierten Port-Listener. Anschließend kommuniziert der Load Balancer über Port 5080 mit dem Server.

Erstellen einer Zielgruppe

1. Nachdem Sie im vorherigen Verfahren „Zielgruppe erstellen“ ausgewählt haben, werden Sie zur Seite „Gruppendetails angeben“ für eine neue Zielgruppe weitergeleitet.
2. Geben Sie im Abschnitt Grundkonfiguration die folgenden Informationen ein.
   - Wählen Sie für Wählen Sie einen Zieltyp die Option IP-Adressen aus.
   - Geben Sie unter Zielgruppenname einen Namen für die Zielgruppe ein.
   - Wählen Sie für Protocol TCP aus.
   - Geben Sie im Feld Port **`5080`** ein.
   - Wählen Sie als IP-Adresstyp IPv4 aus.
   - Wählen Sie für VPC die VPC aus, die Sie für Ihren Transfer Family AS2-Server erstellt haben.
3. Wählen Sie im Abschnitt Health Checks TCP für das Health Check-Protokoll aus.
4. Wählen Sie Weiter aus.

5. Geben Sie auf der Seite Ziele registrieren die folgenden Informationen ein:

   - Vergewissern Sie sich, dass für Network die VPC angegeben ist, die Sie für Ihren Transfer Family AS2-Server erstellt haben.
   - Geben Sie für IPv4-Adresse die private IPv4-Adresse der Endpunkte Ihres Transfer Family AS2-Servers ein.

     Wenn Sie mehr als einen Endpunkt für Ihren Server haben, wählen Sie IPv4-Adresse hinzufügen, um eine weitere Zeile für die Eingabe einer anderen IPv4-Adresse hinzuzufügen. Wiederholen Sie diesen Vorgang, bis Sie die privaten IP-Adressen für alle Endpunkte Ihres Servers eingegeben haben.
   - Stellen Sie sicher, dass Ports auf **`5080`** eingestellt ist.
   - Wählen Sie unten „Als ausstehend einbeziehen“ aus, um Ihre Einträge zum Abschnitt „Ziele überprüfen“ hinzuzufügen.
6. Überprüfen Sie im Abschnitt Ziele überprüfen Ihre IP-Ziele.
7. Wählen Sie Zielgruppe erstellen aus, kehren Sie dann zum vorherigen Verfahren zur Erstellung Ihrer NLB zurück und geben Sie die neue Zielgruppe an der angegebenen Stelle ein.

Testen Sie den Zugriff auf den Server von einer Elastic IP-Adresse

Stellen Sie über den benutzerdefinierten Port eine Connect zum Server her, indem Sie eine Elastic IP-Adresse oder den DNS-Namen des Network Load Balancer verwenden.

> **Important**
>
> Verwalten Sie den Zugriff auf Ihren Server von Client-IP-Adressen aus, indem Sie die Network Access Control Lists (Netzwerk-ACLs) für die auf dem Load Balancer konfigurierten Subnetze verwenden. Netzwerk-ACL-Berechtigungen werden auf Subnetzebene festgelegt, sodass die Regeln für alle Ressourcen gelten, die das Subnetz verwenden. Sie können den Zugriff von Client-IP-Adressen aus nicht mithilfe von Sicherheitsgruppen steuern, da der Zieltyp des Load Balancers auf IP-Adressen statt auf Instances festgelegt ist. Daher speichert der Load Balancer keine Quell-IP-Adressen. Wenn die Integritätsprüfungen des Network Load Balancers fehlschlagen, bedeutet dies, dass der Load Balancer keine Verbindung zum Serverendpunkt herstellen kann. Überprüfen Sie Folgendes, um dieses Problem zu beheben:
>
> - Vergewissern Sie sich, dass die dem Serverendpunkt zugeordnete Sicherheitsgruppe eingehende Verbindungen aus den Subnetzen zulässt, die auf dem Load Balancer

konfiguriert sind. Der Load Balancer muss in der Lage sein, über Port 5080 eine Verbindung zum Serverendpunkt herzustellen.

- Vergewissern Sie sich, dass der Serverstatus Online ist.

## Übertragung von Dateien mithilfe eines AS2-Connectors

AS2-Konnektoren stellen eine Beziehung zwischen Handelspartnern für die Übertragung von AS2-Nachrichten von einem Transfer Family Family-Server an ein externes, partnereigenes Ziel her.

Sie können Transfer Family verwenden, um AS2-Nachrichten zu senden, indem Sie auf die Connector-ID und die Pfade zu den Dateien verweisen, wie im folgenden Befehl `start-file-transfer` AWS Command Line Interface (AWS CLI) dargestellt:

```
aws transfer start-file-transfer --connector-id c-1234567890abcdef0 \
--send-file-paths "/DOC-EXAMPLE-SOURCE-BUCKET/myfile1.txt" "/DOC-EXAMPLE-SOURCE-BUCKET/
myfile2.txt"
```

Führen Sie den folgenden Befehl aus, um die Details für Ihre Konnektoren abzurufen:

```
aws transfer list-connectors
```

Der `list-connectors` Befehl gibt die Connector-IDs, URLs und Amazon-Ressourcennamen (ARNs) für Ihre Konnektoren zurück.

Um die Eigenschaften eines bestimmten Connectors zurückzugeben, führen Sie den folgenden Befehl mit der ID aus, die Sie verwenden möchten:

```
aws transfer describe-connector --connector-id your-connector-id
```

Der `describe-connector` Befehl gibt alle Eigenschaften für den Connector zurück, einschließlich seiner URL, Rollen, Profile, MDNs (Message Disposition Notices), Tags und Überwachungsmetriken.

Sie können überprüfen, ob der Partner die Dateien erfolgreich erhalten hat, indem Sie sich die JSON- und MDN-Dateien ansehen. Diese Dateien werden gemäß den unter beschriebenen Konventionen benannt. Dateinamen und Speicherorte Wenn Sie bei der Erstellung des Connectors eine Protokollierungsrolle konfiguriert haben, können Sie Ihre CloudWatch Protokolle auch auf den Status von AS2-Nachrichten überprüfen.

Einzelheiten zum AS2-Connector finden Sie unter. Details zum AS2-Konnektor anzeigen Weitere Informationen zum Erstellen von AS2-Konnektoren finden Sie unter. AS2-Konnektoren konfigurieren

## Dateinamen und Speicherorte

In diesem Abschnitt werden die Konventionen zur Benennung von Dateien für AS2-Übertragungen beschrieben.

Beachten Sie bei eingehenden Dateiübertragungen Folgendes:

- Sie geben das Basisverzeichnis in einer Vereinbarung an. Das Basisverzeichnis ist der Amazon S3 S3-Bucket-Name in Kombination mit einem Präfix, falls vorhanden. z. B. `/DOC-EXAMPLE-BUCKET/AS2-folder`.
- Wenn eine eingehende Datei erfolgreich verarbeitet wurde, wird die Datei (und die entsprechende JSON-Datei) in dem `/processed` Ordner gespeichert. z. B. `/DOC-EXAMPLE-BUCKET/AS2-folder/processed`.

  Die JSON-Datei enthält die folgenden Felder:

  - `agreement-id`
  - `as2-from`
  - `as2-to`
  - `as2-message-id`
  - `transfer-id`
  - `client-ip`
  - `connector-id`
  - `failure-message`
  - `file-path`
  - `message-subject`
  - `mdn-message-id`
  - `mdn-subject`
  - `requester-file-name`
  - `requester-content-type`
  - `server-id`
  - `status-code`

  - `failure-code`
  - `transfer-size`
- Wenn eine eingehende Datei nicht erfolgreich verarbeitet werden kann, wird die Datei (und die entsprechende JSON-Datei) in dem `/failed` Ordner gespeichert. z. B. `/DOC-EXAMPLE-BUCKET/AS2-folder/failed`.
- Die übertragene Datei wird im `processed` Ordner als gespeichert*`original_filename`*`.`*`messageId`*`.`*`original_extension`*. Das heißt, die Nachrichten-ID für die Übertragung wird vor der ursprünglichen Erweiterung an den Namen der Datei angehängt.
- Eine JSON-Datei wird erstellt und gespeichert als*`original_filename`*`.`*`messageId`*`.`*`original_extension`*`.json`. Zusätzlich zur hinzugefügten Nachrichten-ID wird die Zeichenfolge `.json` an den Namen der übertragenen Datei angehängt.
- Eine MDN-Datei (Message Disposition Notice) wird erstellt und gespeichert als. *`original_filename`*`.`*`messageId`*`.`*`original_extension`*`.mdn` Zusätzlich zur hinzugefügten Nachrichten-ID wird die Zeichenfolge `.mdn` an den Namen der übertragenen Datei angehängt.
- Wenn es eine eingehende Datei mit dem Namen gibt`ExampleFileInS3Payload.dat`, werden die folgenden Dateien erstellt:
  - Datei — `ExampleFileInS3Payload.c4d6b6c7-23ea-4b8c-9ada-0cb811dc8b35@44313c54b0a46a36.`
  - JSON — `ExampleFileInS3Payload.c4d6b6c7-23ea-4b8c-9ada-0cb811dc8b35@44313c54b0a46a36.`
  - MDN — `ExampleFileInS3Payload.c4d6b6c7-23ea-4b8c-9ada-0cb811dc8b35@44313c54b0a46a36.`

Bei ausgehenden Übertragungen ist die Benennung ähnlich, mit dem Unterschied, dass es keine Datei für eingehende Nachrichten gibt und außerdem die Übertragungs-ID für die übertragene Nachricht zum Dateinamen hinzugefügt wird. Die Übertragungs-ID wird durch den `StartFileTransfer` API-Vorgang zurückgegeben (oder wenn ein anderer Prozess oder ein anderes Skript diesen Vorgang aufruft).

- Das `transfer-id` ist eine Kennung, die mit einer Dateiübertragung verknüpft ist. Alle Anfragen, die Teil eines `StartFileTransfer` Anrufs sind, teilen sich eine`transfer-id`.

- Das Basisverzeichnis entspricht dem Pfad, den Sie für die Quelldatei verwenden. Das heißt, das Basisverzeichnis ist der Pfad, den Sie in der `StartFileTransfer` API-Operation oder im `start-file-transfer` AWS CLI API-Befehl angeben. Beispielsweise:

  ```
  aws transfer start-file-transfer --send-file-paths /DOC-EXAMPLE-BUCKET/AS2-folder/file-to-send.txt
  ```

  Wenn Sie diesen Befehl ausführen, werden MDN- und JSON-Dateien in `/DOC-EXAMPLE-BUCKET/AS2-folder/processed` (für erfolgreiche Übertragungen) oder `/DOC-EXAMPLE-BUCKET/AS2-folder/failed` (für erfolglose Übertragungen) gespeichert.
- Eine JSON-Datei wird erstellt und gespeichert als*original_filename*.*transferId*.*messageId*.*original_extension*.json.
- Eine MDN-Datei wird erstellt und gespeichert als*original_filename*.*transferId*.*messageId*.*original_extension*.mdn.
- Wenn es eine ausgehende Datei mit dem Namen gibt`ExampleFileOutTestOutboundSyncMdn.dat`, werden die folgenden Dateien erstellt:
  - JSON — `ExampleFileOutTestOutboundSyncMdn.dedf4601-4e90-4043-b16b-579af35e0d83.fbe18db8-7361-42ff-8ab6-49ec1e435f34@c9c705f0baaaabaa.dat.j`
  - MDN — `ExampleFileOutTestOutboundSyncMdn.dedf4601-4e90-4043-b16b-579af35e0d83.fbe18db8-7361-42ff-8ab6-49ec1e435f34@c9c705f0baaaabaa.dat.m`

Sie können auch die CloudWatch Protokolle überprüfen, um die Details Ihrer Übertragungen einzusehen, einschließlich fehlgeschlagener Übertragungen.

## Statuscodes

In der folgenden Tabelle sind alle Statuscodes aufgeführt, die in den CloudWatch Protokollen protokolliert werden können, wenn Sie oder Ihr Partner eine AS2-Nachricht senden. Verschiedene Schritte zur Nachrichtenverarbeitung gelten für verschiedene Nachrichtentypen und dienen nur der Überwachung. Die Status COMPLETED und FAILED stellen den letzten Verarbeitungsschritt dar und sind in JSON-Dateien sichtbar.

| Code | Beschreibung | Verarbeitung abgeschlossen? |
| --- | --- | --- |
| VERARBEITUNG | Die Nachricht wird gerade in ihr endgültiges Format | Nein |

| Code | Beschreibung | Verarbeitung abgeschlossen? |
| --- | --- | --- |
| | konvertiert. Beispielsweise haben sowohl die Dekomprim ierungs- als auch die Entschlüsselungsschritte diesen Status. | |
| MDN_TRANSMIT | Die Nachrichtenverarbeitung sendet eine MDN-Antwort. | Nein |
| MDN_RECEIVE | Die Nachrichtenverarbeitung empfängt eine MDN-Antwort. | Nein |
| COMPLETED | Die Nachrichtenverarbeitung wurde erfolgreich abgeschlo ssen. Dieser Status gilt auch, wenn ein MDN für eine eingehende Nachricht oder für die MDN-Überprüfung ausgehender Nachrichten gesendet wird. | Ja |
| FEHLGESCHLAGEN | Die Nachrichtenverarbeitung ist fehlgeschlagen. Eine Liste der Fehlercodes finden Sie unterAS2-Fehlercodes. | Ja |

## JSON-Beispieldateien

In diesem Abschnitt werden JSON-Beispieldateien für eingehende und ausgehende Übertragungen aufgeführt, einschließlich Beispieldateien für erfolgreiche und fehlgeschlagene Übertragungen.

Beispiel für eine ausgehende Datei, die erfolgreich übertragen wurde:

```
{
  "requester-content-type": "application/octet-stream",
  "mesage-subject": "File xyzTest from MyCompany_OID to partner YourCompany",
  "requester-file-name": "TestOutboundSyncMdn-9lmCr79hV.dat",
```

```
  "as2-from": "MyCompany_OID",
  "connector-id": "c-c21c63ceaaf34d99b",
  "status-code": "COMPLETED",
  "disposition": "automatic-action/MDN-sent-automatically; processed",
  "transfer-size": 3198,
  "mdn-message-id": "OPENAS2-11072022063009+0000-df865189-1450-435b-9b8d-
d8bc0cee97fd@PartnerA_OID_MyCompany_OID",
  "mdn-subject": "Message be18db8-7361-42ff-8ab6-49ec1e435f34@c9c705f0baaaabaa has been
 accepted",
  "as2-to": "PartnerA_OID",
  "transfer-id": "dedf4601-4e90-4043-b16b-579af35e0d83",
  "file-path": "/DOC-EXAMPLE-BUCKET/as2testcell0000/openAs2/
TestOutboundSyncMdn-9lmCr79hV.dat",
  "as2-message-id": "fbe18db8-7361-42ff-8ab6-49ec1e435f34@c9c705f0baaaabaa",
  "timestamp": "2022-07-11T06:30:10.791274Z"
}
```

Beispiel für eine ausgehende Datei, die nicht erfolgreich übertragen wurde:

```
{
  "failure-code": "HTTP_ERROR_RESPONSE_FROM_PARTNER",
  "status-code": "FAILED",
  "requester-content-type": "application/octet-stream",
  "subject": "Test run from Id da86e74d6e57464aae1a55b8596bad0a to partner
 9f8474d7714e476e8a46ce8c93a48c6c",
  "transfer-size": 3198,
  "requester-file-name": "openAs2TestOutboundWrongAs2Ids-necco-3VYn5n8wE.dat",
  "as2-message-id": "9a9cc9ab-7893-4cb6-992a-5ed8b90775ff@718de4cec1374598",
  "failure-message": "http://Test123456789.us-east-1.elb.amazonaws.com:10080 returned
 status 500 for message with ID 9a9cc9ab-7893-4cb6-992a-5ed8b90775ff@718de4cec1374598",
  "transfer-id": "07bd3e07-a652-4cc6-9412-73ffdb97ab92",
  "connector-id": "c-056e15cc851f4b2e9",
  "file-path": "/testbucket-4c1tq6ohjt9y/as2IntegCell0002/openAs2/
openAs2TestOutboundWrongAs2Ids-necco-3VYn5n8wE.dat",
  "timestamp": "2022-07-11T21:17:24.802378Z"
}
```

Beispiel für eine eingehende Datei, die erfolgreich übertragen wurde:

```
{
  "requester-content-type": "application/EDI-X12",
  "subject": "File openAs2TestInboundAsyncMdn-necco-5Ab6bTfCO.dat sent from MyCompany
 to PartnerA",
```

```
  "client-ip": "10.0.109.105",
  "requester-file-name": "openAs2TestInboundAsyncMdn-necco-5Ab6bTfCO.dat",
  "as2-from": "MyCompany_OID",
  "status-code": "COMPLETED",
  "disposition": "automatic-action/MDN-sent-automatically; processed",
  "transfer-size": 1050,
  "mdn-subject": "Message Disposition Notification",
  "as2-message-id": "OPENAS2-11072022233606+0000-5dab0452-0ca1-4f9b-b622-
fba84effff3c@MyCompany_OID_PartnerA_OID",
  "as2-to": "PartnerA_OID",
  "agreement-id": "a-f5c5cbea5f7741988",
  "file-path": "processed/openAs2TestInboundAsyncMdn-
necco-5Ab6bTfCO.OPENAS2-11072022233606+0000-5dab0452-0ca1-4f9b-b622-
fba84effff3c@MyCompany_OID_PartnerA_OID.dat",
  "server-id": "s-5f7422b04c2447ef9",
  "timestamp": "2022-07-11T23:36:36.105030Z"
}
```


Beispiel für eine eingehende Datei, die nicht erfolgreich übertragen wurde:


```
{
  "failure-code": "INVALID_REQUEST",
  "status-code": "FAILED",
  "subject": "Sending a request from InboundHttpClientTests",
  "client-ip": "10.0.117.27",
  "as2-message-id": "testFailedLogs-TestRunConfig-Default-inbound-direct-
integ-0c97ee55-af56-4988-b7b4-a3e0576f8f9c@necco",
  "as2-to": "0beff6af56c548f28b0e78841dce44f9",
  "failure-message": "Unsupported date format: 2022/123/456T",
  "agreement-id": "a-0ceec8ca0a3348d6a",
  "as2-from": "ab91a398aed0422d9dd1362710213880",
  "file-path": "failed/01187f15-523c-43ac-9fd6-51b5ad2b08f3.testFailedLogs-
TestRunConfig-Default-inbound-direct-integ-0c97ee55-af56-4988-b7b4-a3e0576f8f9c@necco",
  "server-id": "s-0582af12e44540b9b",
  "timestamp": "2022-07-11T06:30:03.662939Z"
}
```


# Überwachen der AS2-Nutzung

Sie können die AS2-Aktivität mit Amazon CloudWatch und überwachenAWS CloudTrail. Weitere Transfer Family-Servermetriken finden Sie unter . Amazon CloudWatch loggt sich ein für AWS Transfer Family

## AS2-Metriken

| Kennzahl | Beschreibung |
| --- | --- |
| `InboundMessage` | Die Gesamtzahl der AS2-Nachrichten, die erfolgreich von einem Handelspartner empfangen wurden.<br><br>Einheiten: Anzahl<br><br>Dauer: 5 Minuten |
| `InboundFailedMessage` | Die Gesamtzahl der AS2-Nachrichten, die nicht erfolgreich von einem Handelspartner empfangen wurden. Das heißt, ein Handelspartner hat eine Nachricht gesendet, aber der Transfer Family-Server konnte sie nicht erfolgreich verarbeiten.<br><br>Einheiten: Anzahl<br><br>Dauer: 5 Minuten |
| `OutboundMessage` | Die Gesamtzahl der AS2-Nachrichten, die erfolgreich vom Transfer Family-Server an einen Handelspartner gesendet wurden.<br><br>Einheiten: Anzahl<br><br>Zeitraum: 5 Minuten |
| `OutboundFailedMessage` | Die Gesamtzahl der AS2-Nachrichten, die erfolglos an einen Handelspartner gesendet wurden. Das heißt, sie wurden vom Transfer Family-Server gesendet, aber vom Handelspartner nicht erfolgreich empfangen.<br><br>Einheiten: Anzahl<br><br>Dauer: 5 Minuten |

# AS2-Statuscodes

In der folgenden Tabelle sind alle Statuscodes aufgeführt, die in CloudWatch Protokollen protokolliert werden können, wenn Sie oder Ihr Partner eine AS2-Nachricht senden. Verschiedene Schritte zur Nachrichtenverarbeitung gelten für verschiedene Nachrichtentypen und sind nur für die Überwachung vorgesehen. Die Zustände COMPLETED und FAILED stellen den letzten Schritt der Verarbeitung dar und sind in JSON-Dateien sichtbar.

| Code | Beschreibung | Wird die Verarbeitung abgeschlossen? |
|---|---|---|
| VERFÜGUNG | Die Nachricht wird gerade in ihr endgültiges Format konvertiert. Beispielsweise haben die Schritte Dekomprimierung und Entschlüsselung beide diesen Status. | Nein |
| MDN_TRANSMIT | Die Nachrichtenverarbeitung sendet eine MDN-Antwort. | Nein |
| MDN_AUFLÖSCHEN | Die Nachrichtenverarbeitung empfängt eine MDN-Antwort. | Nein |
| COMPLETED | Die Nachrichtenverarbeitung wurde erfolgreich abgeschlossen. Dieser Status gilt auch, wenn ein MDN für eine eingehende Nachricht oder für die MDN-Verifizierung ausgehender Nachrichten gesendet wird. | Ja |
| FEHLGESCHLAGEN | Die Nachrichtenverarbeitung ist fehlgeschlagen. Eine Liste der Fehlercodes finden Sie unter AS2-Fehlercodes. | Ja |

# AS2-Fehlercodes

In der folgenden Tabelle sind Fehlercodes aufgeführt und beschrieben, die Sie möglicherweise von AS2-Dateiübertragungen erhalten.

AS2-Fehlercodes

| Code | Fehler | Beschreibung und Auflösung |
| --- | --- | --- |
| `ACCESS_DENIED` | • Zugriff verweigert. Überprüfen Sie, ob Ihre Zugriffsrolle über die erforderlichen Berechtig ungen verfügt.<br>• Ungültiger Dateipfad *`send-file-path`*<br>• Anmeldeinformationen konnten nicht mit abgerufen werden ErrorCode: *`error-code`* | Tritt auf, wenn eine `StartFileTransfer` Anforderung bearbeitet wird, bei der eine der ungültig oder falsch formatiert `SendFileP aths` ist. Das heißt, dem Pfad fehlt der Amazon S3-Bucket-Name oder der Pfad enthält ungültige Zeichen. Tritt auch auf, wenn Transfer Family die Zugriffsrolle oder Protokollierungsrolle nicht übernehmen kann.<br><br>Stellen Sie sicher, dass der Pfad einen gültigen Amazon S3-Bucket-Namen und Schlüsselnamen enthält. |
| `AGREEMENT_NOT_FOUND` | Die Vereinbarung wurde nicht gefunden. | Entweder wurde die Vereinbar ung nicht gefunden oder die Vereinbarung ist mit einem inaktiven Profil verknüpft.<br><br>Aktualisieren Sie die Vereinbar ung innerhalb des Transfer Family-Servers, um aktive Profile einzuschließen. |

| Code | Fehler | Beschreibung und Auflösung |
| --- | --- | --- |
| CONNECTOR_NOT_FOUND | Connector oder zugehörige Konfiguration wurde nicht gefunden. | Entweder wurde der Konnektor nicht gefunden oder der Konnektor ist einem inaktiven Profil zugeordnet.<br><br>Aktualisieren Sie den Konnektor, um aktive Profile einzuschließen. |

| Code | Fehler | Beschreibung und Auflösung |
| --- | --- | --- |
| `CREDENTIALS_RETRIEVAL_FAILED` | 1. Das Secret wurde in Secrets Manager nicht gefunden.<br>2. Kann nicht auf Secrets Manager zugreifen.<br>3. Secret konnte in Secrets Manager nicht entschlüsselt werden.<br>4. Secret-Wert kann aufgrund von Drosselung nicht abgerufen werden. | Für die AS2-Basisauthentifizierung muss das Secret korrekt formatiert sein.<br>Die folgenden Lösungen entsprechen den in der vorherigen Spalte aufgeführten Fehlern.<br>1. Stellen Sie sicher, dass die geheime ID korrekt ist.<br>2. Stellen Sie sicher, dass die Zugriffsrolle über die entsprechenden Berechtigungen zum Lesen des Secrets verfügt. Die Zugriffsrolle muss Lese- und Schreibzugriff auf das übergeordnete Verzeichnis des in der `StartFileTransfer` Anforderung verwendeten Dateispeicherorts gewähren. Stellen Sie außerdem sicher, dass die Rolle Lese- und Schreibzugriff auf das übergeordnete Verzeichnis der Dateien bietet, die Sie mit senden möchten`StartFileTransfer` .<br>3. Wenn ein vom Kunden verwalteter Schlüssel für das Secret verwendet wird, stellen Sie sicher, dass die |

| Code | Fehler | Beschreibung und Auflösung |
| --- | --- | --- |
| | | Zugriffsrolle über Berechtigungen für den AWS Key Management Service (AWS KMS)-Schlüssel verfügt.<br>4. Die entsprechenden Kontingente finden Sie unter Kontingente für den Umgang mit Geheimnissen. |
| `DECOMPRESSION_FAILED` | Nachricht konnte nicht dekomprimiert werden. | Entweder ist die gesendete Datei beschädigt oder der Komprimierungsalgorithmus ist ungültig.<br><br>Senden Sie die Nachricht erneut und überprüfen Sie, ob die ZLIB-Komprimierung verwendet wird, oder senden Sie die Nachricht erneut, ohne dass die Komprimierung aktiviert ist. |
| `DECRYPT_FAILED` | Nachrichten-*`ID konnte nicht entschlüsselt werden`*. Stellen Sie sicher, dass der Partner über den richtigen öffentlichen Verschlüsselungsschlüssel verfügt. | Die Entschlüsselung ist fehlgeschlagen.<br><br>Vergewissern Sie sich, dass der Partner eine Nutzlast mit einem gültigen Zertifikat gesendet hat und dass die Verschlüsselung mit einem gültigen Verschlüsselungsalgorithmus durchgeführt wurde. |

| Code | Fehler | Beschreibung und Auflösung |
| --- | --- | --- |
| `DECRYPT_FAILED_INVALID_SMIME_FORMAT` | Envelope mimePart konnte nicht analysiert werden. | MIME-Nutzlast ist entweder beschädigt oder hat ein nicht unterstütztes SMIME-Format.<br><br>Der Sender sollte sicherstellen, dass das verwendete Format unterstützt wird, und dann die Nutzlast erneut senden. |
| `DECRYPT_FAILED_NO_DECRYPTION_KEY_FOUND` | Es wurde kein passender Entschlüsselungsschlüssel gefunden. | Dem Partnerprofil wurde kein Zertifikat zugewiesen, das der Nachricht entspricht, oder die Zertifikate, die der Nachricht entsprechen, sind jetzt abgelaufen oder nicht mehr gültig.<br><br>Sie müssen das Partnerprofil aktualisieren und sicherstellen, dass es ein gültiges Zertifikat enthält. |
| `DECRYPT_FAILED_UNSUPPORTED_ENCRYPTION_ALG` | Die SMIME-Nutzlastentschlüsselung wurde mit einem nicht unterstützten Algorithmus mit der ID angefordert: *`encryption-ID`* . | Der Remote-Sender hat eine AS2-Nutzlast mit einem nicht unterstützten Verschlüsselungsalgorithmus gesendet.<br><br>Der Sender muss einen Verschlüsselungsalgorithmus auswählen, der von unterstützt wirdAWS Transfer Family. |

| Code | Fehler | Beschreibung und Auflösung |
| --- | --- | --- |
| `DUPLICATE_MESSAGE` | Doppelter oder doppelt verarbeiteter Schritt. | Die Nutzlast hat einen doppelten Verarbeitungsschritt. Beispielsweise gibt es zwei Verschlüsselungsschritte.<br><br>Senden Sie die Nachricht mit einem einzigen Schritt zum Signieren, Komprimieren und Verschlüsseln erneut. |
| `ENCRYPT_FAILED_NO_ENCRYPTION_KEY_FOUND` | Es wurden keine gültigen öffentlichen Verschlüsselungszertifikate im Profil gefunden: *`local-profile-ID`* | Transfer Family versucht, eine ausgehende Nachricht zu verschlüsseln, aber es werden keine Verschlüsselungszertifikate für das lokale Profil gefunden.<br><br>Lösungsoptionen:<br><br>• Stellen Sie sicher, dass dem lokalen Profil ein Zertifikat und ein privater Schlüssel für die Verschlüsselung angefügt sind.<br>• Stellen Sie sicher, dass das Verschlüsselungszertifikat derzeit aktiv ist. |

| Code | Fehler | Beschreibung und Auflösung |
| --- | --- | --- |
| ENCRYPTION_FAILED | *Dateidateiname konnte nicht verschlüsselt werden*. | Die zu sendende Datei ist nicht für die Verschlüsselung verfügbar.<br><br>Stellen Sie sicher, dass sich die Datei an ihrem erwarteten AS2-Speicherort befindet und über die Berechtigung zum Lesen der Datei AWS Transfer Family verfügt. |
| FILE_SIZE_TOO_LARGE | Die Dateigröße ist zu groß. | Dies tritt auf, wenn eine Datei gesendet oder empfangen wird, die die Dateigröß enbeschränkung überschre itet. |
| HTTP_ERROR_RESPONSE_FROM_PARTNER | *Partner-URL* hat den Status 400 für Nachricht mit ID=*message-ID* zurückgeg eben. | Die Kommunikation mit dem AS2-Server des Partners gab einen unerwarteten HTTP-Antwortcode zurück.<br><br>Der Partner kann möglicher weise mehr Diagnosen aus seinen AS2-Serverprotokollen bereitstellen. |
| INSUFFICENT_MESSAGE_SECURITY_UNENCRYPTED | Verschlüsselung ist erforderl ich. | Der Partner hat eine unverschl üsselte Nachricht an Transfer Family gesendet, die nicht unterstützt wird. Der Sender muss eine verschlüsselte Nutzlast verwenden. |

| Code | Fehler | Beschreibung und Auflösung |
| --- | --- | --- |
| `INVALID_ENDPOINT_PROTOCOL` | Nur HTTP und HTTPS werden unterstützt. | Sie müssen HTTP oder HTTPS als Protokoll in Ihrer AS2-Konnektor-Konfiguration angeben. |

| Code | Fehler | Beschreibung und Auflösung |
| --- | --- | --- |
| `INVALID_REQUEST` | 1. Es liegt ein Problem mit einem Nachrichten-Header vor.<br>2. Secret-JSON konnte nicht analysiert werden.<br><br>Secret JSON stimmte nicht mit dem erwarteten Format überein.<br>3. Secret muss eine JSON-Zeichenfolge sein.<br>4. Der Benutzername darf keinen Doppelpunkt enthalten.<br><br>Der Benutzername darf keine Steuerzeichen enthalten.<br><br>Der Benutzername darf nur ASCII-Zeichen enthalten.<br><br>Das Passwort darf keine Steuerzeichen enthalten.<br><br>Das Passwort darf nur ASCII-Zeichen enthalten. | Dieser Fehler hat mehrere Ursachen. Die folgenden Lösungen entsprechen den in der vorherigen Spalte aufgeführten Fehlern.<br><br>1. Überprüfen Sie die `as2-to` Felder `as2-from` und . Stellen Sie sicher, dass die ursprüngliche Nachricht en-ID für das MDN-Forma t korrekt ist. Stellen Sie außerdem sicher, dass dem Nachrichten-ID-Format keine AS2-Header fehlen.<br>2. Stellen Sie sicher, dass der Secret-Wert dem dokumenti erten Format entspricht, wie unter beschriebenAktiviere n Sie die Standardauthentifi zierung für AS2-Konne ktoren.<br>3. Stellen Sie sicher, dass das Secret als Zeichenfo lge und nicht als Binärdatei bereitgestellt wird.<br>4. Nehmen Sie die erforderl iche Korrektur am Benutzernamen oder Passwort vor. |

| Code | Fehler | Beschreibung und Auflösung |
| --- | --- | --- |
| `INVALID_URL_FORMAT` | Ungültiges URL-Format: *URL* | Dies tritt auf, wenn Sie eine ausgehende Nachricht mit einem Connector senden, der mit einer fehlerhaften URL konfiguriert ist.<br><br>Stellen Sie sicher, dass der Konnektor mit einer gültigen HTTP- oder HTTPS-URL konfiguriert ist. |
| `MDN_RESPONSE_INDICATES_AUTHENTICATION_FAILED` | Nicht zutreffend | Der Empfänger kann den Sender nicht authentifizieren. Der Handelspartner gibt einen MDN mit dem Dispositionsmodifikator Error: authentication-failed an Transfer Family zurück. |
| `MDN_RESPONSE_INDICATES_DECOMPRESSION_FAILED` | Nicht zutreffend | Dies tritt auf, wenn der Empfänger den Nachrichteninhalt nicht dekomprimieren kann. Der Handelspartner gibt einen MDN mit dem Dispositionsmodifikator Error: decompression-failed an Transfer Family zurück. |
| `MDN_RESPONSE_INDICATES_DECRYPTION_FAILED` | Nicht zutreffend | Der Empfänger kann den Nachrichteninhalt nicht entschlüsseln. Der Handelspartner gibt einen MDN mit dem Dispositionsmodifikator Error: authentication-failed an Transfer Family zurück. |

| Code | Fehler | Beschreibung und Auflösung |
| --- | --- | --- |
| `MDN_RESPONSE_INDICATES_INSUFFICIENT_MESSAGE_SECURITY` | Nicht zutreffend | Der Empfänger erwartet, dass die Nachricht signiert oder verschlüsselt ist, dies jedoch nicht. Der Handelspartner gibt einen MDN mit dem Dispositionsmodifikator Error an Transfer Family zurück: insufficient-message-security<br><br>Aktivieren Sie Signatur und/oder Verschlüsselung auf dem Konnektor, um den Erwartungen des Handelspartners zu entsprechen. |
| `MDN_RESPONSE_INDICATES_INTEGRITY_CHECK_FAILED` | Nicht zutreffend | Der Empfänger kann die Integrität von Inhalten nicht überprüfen. Der Handelspartner gibt einen MDN mit dem Dispositionsmodifikator Error an Transfer Family zurück: integrity-check-failed |
| `PATH_NOT_FOUND` | *`Verzeichnisdateipfad kann nicht erstellt werden`*. Der übergeordnete Pfad konnte nicht gefunden werden. | Transfer Family versucht, ein Verzeichnis im Amazon S3-Bucket des Kunden zu erstellen, der Bucket wird jedoch nicht gefunden.<br><br>Stellen Sie sicher, dass jeder im `StartFileTransfer` Befehl erwähnte Pfad den Namen eines vorhandenen Buckets enthält. |

| Code | Fehler | Beschreibung und Auflösung |
| --- | --- | --- |
| SEND_FILE_NOT_FOUND | Dateipfad-*Dateipfad* nicht gefunden. | Transfer Family kann die Datei im Vorgang Datei senden nicht finden.<br><br>Überprüfen Sie, ob das konfigurierte Stammverz eichnis und der Pfad gültig sind und ob Transfer Family über Leseberechtigungen für die Datei verfügt. |
| SERVER_NOT_FOUND | Der mit der Nachricht verknüpfte Server kann nicht gefunden werden. | Transfer Family konnte den Server beim Empfang einer Nachricht nicht finden. Dies kann passieren, wenn der Server während der Verarbeit ung einer eingehenden Nachricht gelöscht wird. |
| SERVER_NOT_ONLINE | Die *Server-ID* ist nicht online. | Der Transfer Family-Server ist offline.<br><br>Starten Sie den Server, damit er Nachrichten empfangen und verarbeiten kann. |

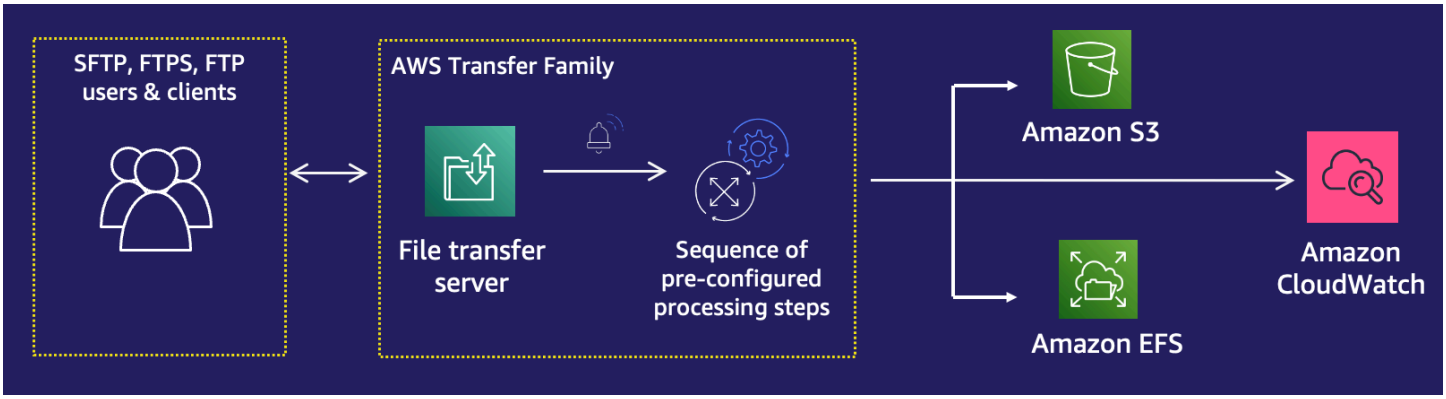
| Code | Fehler | Beschreibung und Auflösung |
| --- | --- | --- |
| `SIGNING_FAILED` | Datei konnte nicht signiert werden. | Die zu sendende Datei steht nicht zum Signieren zur Verfügung oder die Signierung konnte nicht durchgeführt werden.<br><br>Stellen Sie sicher, dass sich die Datei an ihrem erwarteten AS2-Speicherort befindet und über die Berechtigung zum Lesen der Datei AWS Transfer Family verfügt. |
| `SIGNING_FAILED_NO_SIGNING_KEY_FOUND` | Kein Zertifikat für Profil gefunden: *`local-profile-ID`* . | Der Versuch, eine ausgehende Nachricht zu signieren, es werden jedoch keine Signaturzertifikate für das lokale Profil gefunden.<br><br>Lösungsoptionen:<br><br>• Stellen Sie sicher, dass dem lokalen Profil ein Zertifikat und ein privater Schlüssel zum Signieren angefügt sind.<br>• Stellen Sie sicher, dass das Signaturzertifikat derzeit aktiv ist. |

| Code | Fehler | Beschreibung und Auflösung |
| --- | --- | --- |
| `UNABLE_RESOLVE_HOST_TO_IP_ADDRESS` | Hostname konnte nicht in IP-Adressen aufgelöst werden. | Transfer Family kann keine DNS-zu-IP-Adressauflösung auf dem öffentlichen DNS-Server durchführen, der im AS2-Konnektor konfiguriert ist.<br><br>Aktualisieren Sie den Konnektor so, dass er auf eine gültige Partner-URL verweist. |
| `UNABLE_TO_CONNECT_TO_REMOTE_HOST_OR_IP` | Bei der Verbindung mit dem Endpunkt ist eine Zeitüberschreitung aufgetreten. | Transfer Family kann keine Socket-Verbindung zum AS2-Server des konfigurierten Partners herstellen.<br><br>Überprüfen Sie, ob der AS2-Server des Partners unter der konfigurierten IP-Adresse verfügbar ist. |
| `UNABLE_TO_RESOLVE_HOSTNAME` | Hostname-*`Hostname kann nicht aufgelöst werden`*. | Der Transfer Family-Server konnte den Hostnamen des Partners nicht mithilfe eines öffentlichen DNS-Servers auflösen.<br><br>Überprüfen Sie, ob der konfigurierte Host registriert ist und ob der DNS-Datensatz Zeit hatte, zu veröffentlichen. |

| Code | Fehler | Beschreibung und Auflösung |
| --- | --- | --- |
| `VERIFICATION_FAILED` | Die Signaturüberprüfung für die Nachrichten-ID der AS2-Nachricht ist fehlgeschlagen oder ein MIC-Code stimmt nicht überein. | Überprüfen Sie, ob das Signaturzertifikat des Senders mit den Signaturzertifikaten für das Remote-Profil übereinst immt. Überprüfen Sie auch, ob die MIC-Algorithmen mit kompatibel sindAWS Transfer Family. |
| `VERIFICATION_FAILED_NO_MATCHING_KEY_FOUND` | • Im Profil konnte kein öffentliches Zertifikat gefunden werden, das mit der Nachrichtensignatur übereinstimmt: *`Partner-Profil-ID`* .<br>• Zertifikate für nicht vorhandene Profile können nicht abgerufen werden: *`Partner-Profil-ID`* .<br>• Im Profil wurde kein gültiges Zertifikat gefunden: *`Partner-Profil-ID`* . | AWS Transfer Family versucht, die Signatur für eine empfangene Nachricht zu überprüfen, aber es wird kein passendes Signaturzertifikat für das Partnerprofil gefunden.<br><br>Lösungsoptionen:<br><br>• Stellen Sie sicher, dass dem Partnerprofil ein Signaturz ertifikat angefügt ist.<br>• Stellen Sie sicher, dass das Zertifikat derzeit aktiv ist.<br>• Stellen Sie sicher, dass das Zertifikat das richtige Signaturzertifikat für den Partner ist. |

# AWS Transfer Family verwaltete Workflows

AWS Transfer Family unterstützt verwaltete Workflows für die Dateiverarbeitung. Mit verwalteten Workflows können Sie einen Workflow starten, nachdem eine Datei über SFTP, FTPS oder FTP übertragen wurde. Mit dieser Funktion können Sie Ihre Compliance-Anforderungen für den business-to-business (B2B-) Dateiaustausch sicher und kostengünstig erfüllen, indem Sie alle für die Dateiverarbeitung erforderlichen Schritte koordinieren. Darüber hinaus profitieren Sie von end-to-end Auditing und Transparenz.




Durch die Orchestrierung von Dateiverarbeitungsaufgaben helfen Ihnen verwaltete Workflows dabei, Daten vorzuverarbeiten, bevor sie von Ihren nachgelagerten Anwendungen verwendet werden. Zu diesen Aufgaben zur Dateiverarbeitung können gehören:

- Dateien in benutzerspezifische Ordner verschieben.
- Entschlüsseln von Dateien als Teil eines Workflows.
- Dateien taggen.
- Durchführung einer benutzerdefinierten Verarbeitung durch Erstellen und Anhängen einer AWS Lambda Funktion an einen Workflow.
- Senden von Benachrichtigungen, wenn eine Datei erfolgreich übertragen wurde. (Einen Blogbeitrag, in dem dieser Anwendungsfall detailliert beschrieben wird, finden Sie unter Anpassen von Benachrichtigungen zur Dateizustellung mithilfe AWS Transfer Family verwalteter Workflows.)

Um häufig auftretende Aufgaben zur Dateiverarbeitung nach dem Upload, die sich über mehrere Geschäftsbereiche in Ihrem Unternehmen erstrecken, schnell zu replizieren und zu standardisieren, können Sie Workflows mithilfe von Infrastructure as Code (IaC) bereitstellen. Sie können einen verwalteten Workflow angeben, der für vollständig hochgeladene Dateien initiiert wird. Sie können auch einen anderen verwalteten Workflow angeben, der für Dateien initiiert werden soll, die

aufgrund einer vorzeitigen Sitzungsunterbrechung nur teilweise hochgeladen wurden. Die integrierte Ausnahmebehandlung hilft Ihnen, schnell auf Ergebnisse der Dateiverarbeitung zu reagieren, und bietet Ihnen gleichzeitig die Kontrolle darüber, wie mit Fehlern umgegangen werden soll. Darüber hinaus erstellt jeder Workflow-Schritt detaillierte Protokolle, die Sie überprüfen können, um die Herkunft der Daten nachzuverfolgen.

Führen Sie zunächst die folgenden Aufgaben aus:

1. Richten Sie Ihren Workflow so ein, dass er je nach Ihren Anforderungen Vorverarbeitungsaktionen wie Kopieren, Markieren und andere Schritte enthält. Details dazu finden Sie unter Erstellen Sie einen Workflow.
2. Konfigurieren Sie eine Ausführungsrolle, die Transfer Family zur Ausführung des Workflows verwendet. Details dazu finden Sie unter IAM-Richtlinien für Workflows.
3. Ordnen Sie den Workflow einem Server zu, sodass die in diesem Workflow angegebenen Aktionen beim Eintreffen der Datei in Echtzeit ausgewertet und initiiert werden. Details dazu finden Sie unter Konfigurieren Sie einen Workflow und führen Sie ihn aus.

Ähnliche Informationen

- Informationen zur Überwachung Ihrer Workflow-Ausführungen finden Sie unter CloudWatch Metriken für Transfer Family verwenden.
- Ausführliche Ausführungsprotokolle und Informationen zur Fehlerbehebung finden Sie unterWorkflow-bezogene Fehler mithilfe von Amazon beheben CloudWatch.
- Transfer Family bietet einen Blogbeitrag und einen Workshop, die Sie durch den Aufbau einer Dateiübertragungslösung führen. Diese Lösung nutzt AWS Transfer Family verwaltete SFTP/FTPS-Endpunkte sowie Amazon Cognito und DynamoDB für die Benutzerverwaltung.

  Der Blogbeitrag ist unter Amazon Cognito als Identitätsanbieter mit AWS Transfer Family Amazon S3 verwenden verfügbar. Die Details zum Workshop finden Sie hier.
- Unter AWS Transfer Family Verwaltete Workflows finden Sie eine kurze Einführung in die Workflows von Transfer Family.

Themen

- Erstellen Sie einen Workflow
- Verwenden Sie vordefinierte Schritte
- Verwenden Sie benutzerdefinierte Schritte zur Dateiverarbeitung

- IAM-Richtlinien für Workflows
- Ausnahmebehandlung für einen Workflow
- Überwachen Sie die Workflow-Ausführung
- Erstellen Sie einen Workflow aus einer Vorlage
- Einen Workflow von einem Transfer Family Family-Server entfernen
- Einschränkungen und Einschränkungen verwalteter Workflows

Weitere Hilfe zu den ersten Schritten mit verwalteten Workflows finden Sie in den folgenden Ressourcen:

- AWS Transfer Family Demovideo zu verwalteten Workflows
- Blogbeitrag zum Aufbau einer Cloud-nativen Dateiübertragungsplattform mithilfe von AWS Transfer Family Workflows

# Erstellen Sie einen Workflow

Sie können einen verwalteten Workflow mithilfe von erstellen AWS Management Console, wie in diesem Thema beschrieben. Um den Prozess der Workflow-Erstellung so einfach wie möglich zu gestalten, sind für die meisten Abschnitte der Konsole kontextuelle Hilfebereiche verfügbar.

Ein Workflow besteht aus zwei Arten von Schritten:

- Nominale Schritte — Nominale Schritte sind Schritte zur Dateiverarbeitung, die Sie auf eingehende Dateien anwenden möchten. Wenn Sie mehr als einen nominalen Schritt auswählen, wird jeder Schritt in einer linearen Reihenfolge verarbeitet.
- Schritte zur Ausnahmebehandlung — Ausnahmehandler sind Schritte zur Dateiverarbeitung, die AWS Transfer Family ausgeführt werden, falls nominelle Schritte fehlschlagen oder zu Validierungsfehlern führen.

Erstellen Sie einen Workflow

1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Wählen Sie im linken Navigationsbereich Workflows aus.
3. Wählen Sie auf der Seite Workflows die Option Workflow erstellen aus.

4. Geben Sie auf der Seite Workflow erstellen eine Beschreibung ein. Diese Beschreibung wird auf der Seite Workflows angezeigt.
5. Wählen Sie im Abschnitt Nominale Schritte die Option Schritt hinzufügen aus. Fügen Sie einen oder mehrere Schritte hinzu.

   a. Wählen Sie einen Schritttyp aus den verfügbaren Optionen aus. Weitere Informationen zu den verschiedenen Schritttypen finden Sie unterthe section called “Verwenden Sie vordefinierte Schritte”.

   b. Wählen Sie Weiter und konfigurieren Sie dann die Parameter für den Schritt.

   c. Wählen Sie Weiter und überprüfen Sie dann die Details für den Schritt.

   d. Wählen Sie Schritt erstellen, um den Schritt hinzuzufügen und fortzufahren.

   e. Fügen Sie nach Bedarf weitere Schritte hinzu. Die maximale Anzahl von Schritten in einem Workflow beträgt 8.

   f. Nachdem Sie alle erforderlichen nominalen Schritte hinzugefügt haben, scrollen Sie nach unten zum Abschnitt Ausnahmebehandler — optional und wählen Sie Schritt hinzufügen aus.

   > **Note**
   >
   > Damit Sie in Echtzeit über Fehler informiert werden, empfehlen wir Ihnen, Ausnahmebehandlungsroutinen und Schritte einzurichten, die ausgeführt werden, wenn Ihr Workflow fehlschlägt.

6. Um Ausnahmehandler zu konfigurieren, fügen Sie Schritte auf die gleiche Weise wie zuvor beschrieben hinzu. Wenn eine Datei dazu führt, dass ein Schritt eine Ausnahme auslöst, werden Ihre Ausnahmebehandlungsroutinen nacheinander aufgerufen.
7. (Optional) Scrollen Sie nach unten zum Abschnitt „Tags“ und fügen Sie Tags für Ihren Workflow hinzu.
8. Überprüfen Sie die Konfiguration und wählen Sie Workflow erstellen aus.

> **Important**
>
> Nachdem Sie einen Workflow erstellt haben, können Sie ihn nicht mehr bearbeiten. Überprüfen Sie die Konfiguration daher sorgfältig.

# Konfigurieren Sie einen Workflow und führen Sie ihn aus

Bevor Sie einen Workflow ausführen können, müssen Sie ihn einem Transfer Family Family-Server zuordnen.

So konfigurieren Sie Transfer Family so, dass ein Workflow für hochgeladene Dateien ausgeführt wird

1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Wählen Sie im linken Navigationsbereich Server aus.
   - Um den Workflow zu einem vorhandenen Server hinzuzufügen, wählen Sie den Server aus, den Sie für Ihren Workflow verwenden möchten.
   - Alternativ können Sie einen neuen Server erstellen und den Workflow hinzufügen. Weitere Informationen finden Sie unter Konfiguration eines SFTP-, FTPS- oder FTP-Serverendpunkts.
3. Scrollen Sie auf der Detailseite für den Server nach unten zum Abschnitt Zusätzliche Details und wählen Sie dann Bearbeiten aus.

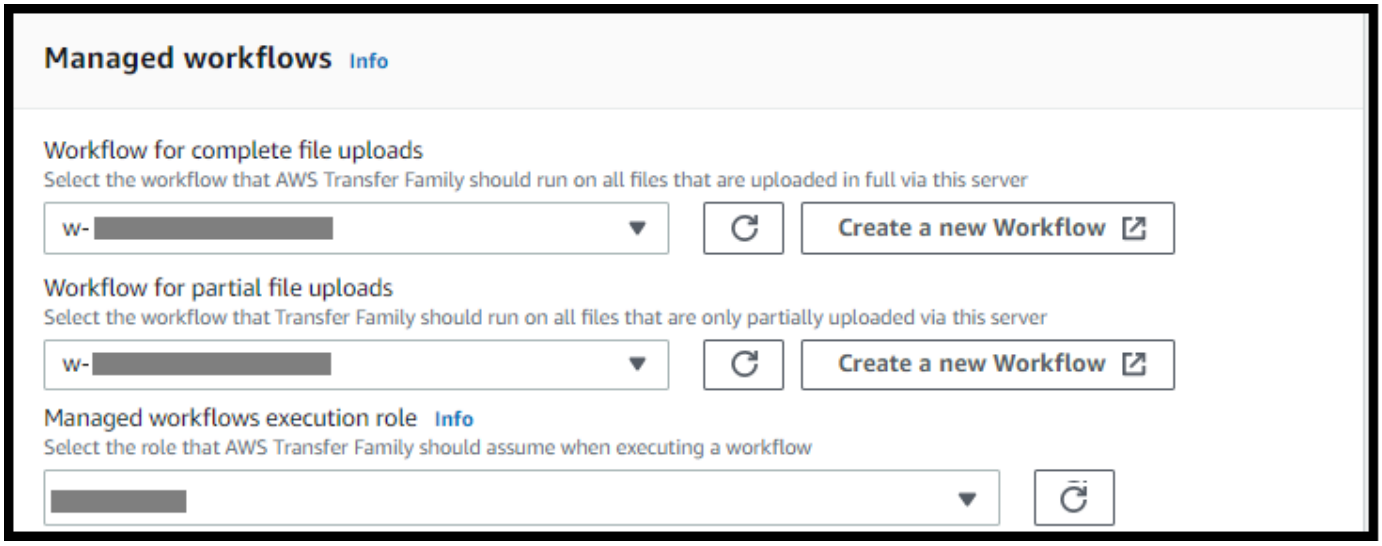
   > **Note**
   >
   > Standardmäßig sind Servern keine Workflows zugeordnet. Sie verwenden den Abschnitt Zusätzliche Details, um dem ausgewählten Server einen Workflow zuzuordnen.

4. Wählen Sie auf der Seite Zusätzliche Details bearbeiten im Abschnitt Verwaltete Workflows einen Workflow aus, der für alle Uploads ausgeführt werden soll.

   > **Note**
   >
   > Wenn Sie noch keinen Workflow haben, wählen Sie Neuen Workflow erstellen aus, um einen zu erstellen.

   a. Wählen Sie die zu verwendende Workflow-ID aus.

   b. Wählen Sie eine Ausführungsrolle aus. Dies ist die Rolle, die Transfer Family bei der Ausführung der Workflow-Schritte einnimmt. Weitere Informationen finden Sie unter IAM-Richtlinien für Workflows. Wählen Sie Save (Speichern) aus.




> **Note**
>
> Wenn Sie nicht mehr möchten, dass ein Workflow dem Server zugeordnet wird, können Sie die Zuordnung entfernen. Details hierzu finden Sie unter Einen Workflow von einem Transfer Family Family-Server entfernen.

Um einen Workflow auszuführen

Um einen Workflow auszuführen, laden Sie eine Datei auf einen Transfer Family Family-Server hoch, den Sie mit einem zugehörigen Workflow konfiguriert haben.

> **Note**
>
> Jedes Mal, wenn Sie einen Workflow von einem Server entfernen und ihn durch einen neuen ersetzen oder die Serverkonfiguration aktualisieren (was sich auf die Ausführungsrolle eines Workflows auswirkt), müssen Sie etwa 10 Minuten warten, bevor Sie den neuen Workflow ausführen. Der Transfer Family Family-Server speichert die Workflow-Details im Cache, und es dauert 10 Minuten, bis der Server seinen Cache aktualisiert hat.
> Darüber hinaus müssen Sie sich von allen aktiven SFTP-Sitzungen abmelden und sich nach Ablauf der 10-minütigen Wartezeit wieder anmelden, um die Änderungen zu sehen.

Example

```
# Execute a workflow
> sftp bob@s-1234567890abcdef0.server.transfer.us-east-1.amazonaws.com

Connected to s-1234567890abcdef0.server.transfer.us-east-1.amazonaws.com.
sftp> put doc1.pdf
Uploading doc1.pdf to /DOC-EXAMPLE-BUCKET/home/users/bob/doc1.pdf
doc1.pdf                                                                    100% 5013KB
 601.0KB/s   00:08
sftp> exit
>
```

Nachdem Ihre Datei hochgeladen wurde, wird die definierte Aktion an Ihrer Datei ausgeführt. Wenn Ihr Workflow beispielsweise einen Kopierschritt enthält, wird die Datei an den Speicherort kopiert, den Sie in diesem Schritt definiert haben. Sie können Amazon CloudWatch Logs verwenden, um die ausgeführten Schritte und ihren Ausführungsstatus nachzuverfolgen.

## Workflow-Details anzeigen

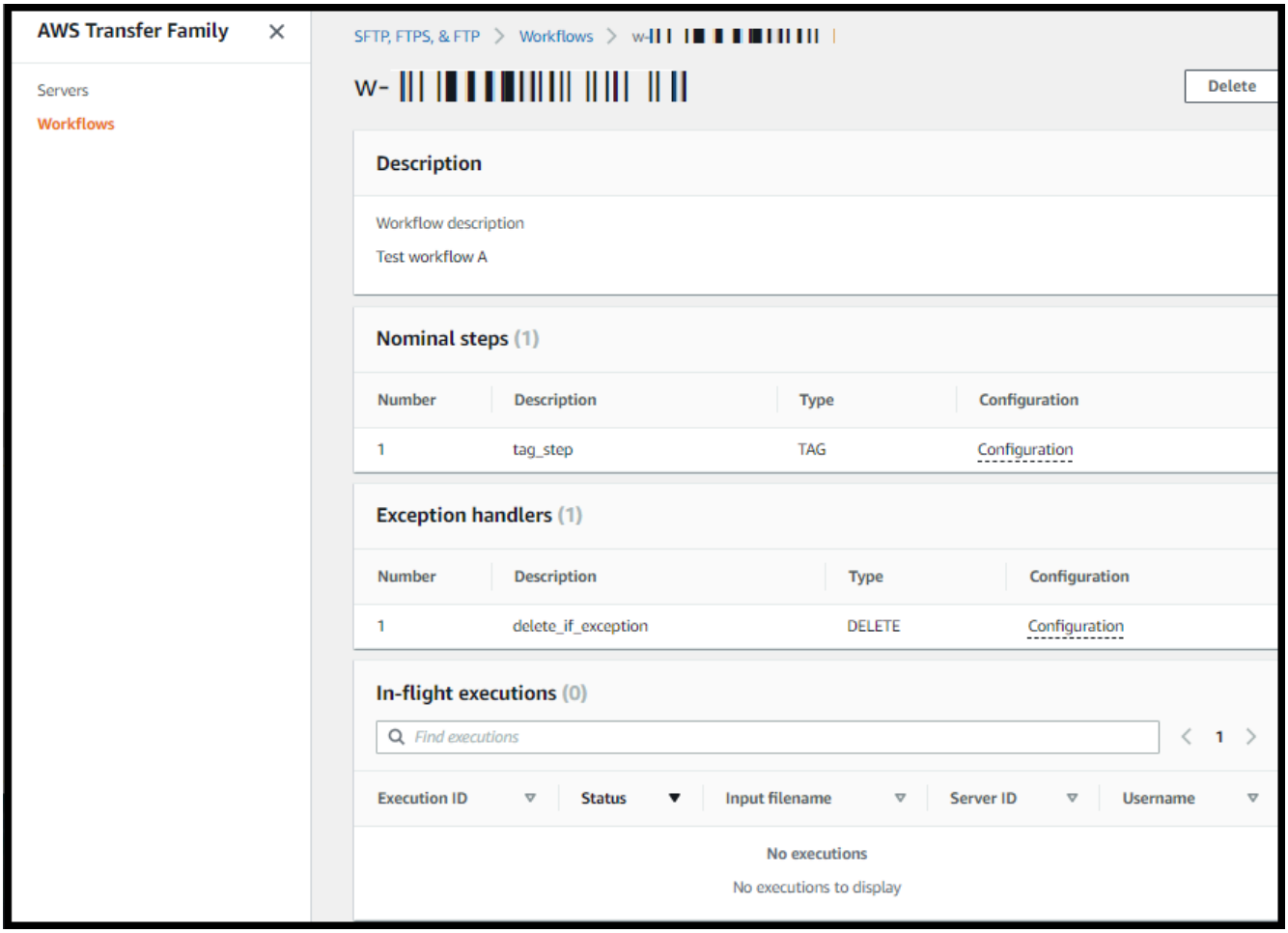
Sie können Details zu zuvor erstellten Workflows oder zu Workflow-Ausführungen anzeigen. Um diese Details anzuzeigen, können Sie die Konsole oder die AWS Command Line Interface (AWS CLI) verwenden.

Console

Workflow-Details anzeigen

1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Wählen Sie im linken Navigationsbereich Workflows aus.
3. Wählen Sie auf der Workflow-Seite einen Workflow aus.

   Die Seite mit den Workflow-Details wird geöffnet.



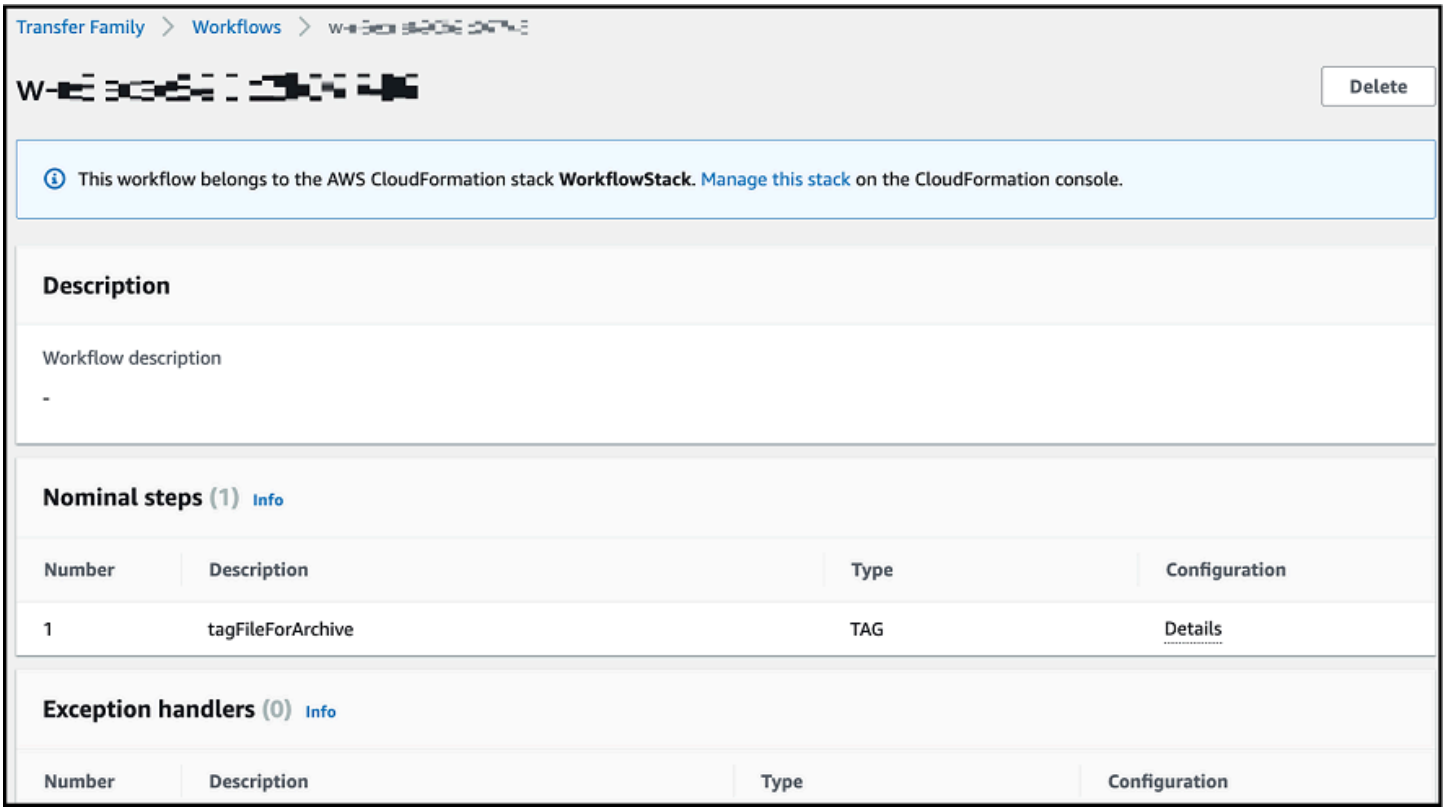

CLI

Verwenden Sie den `describe-workflow` CLI-Befehl, um die Workflow-Details anzuzeigen, wie im folgenden Beispiel gezeigt. Ersetzen Sie die Workflow-ID *w-1234567890abcdef0* durch Ihren eigenen Wert. Weitere Informationen finden Sie unter describe-workflow in der AWS CLI Befehlsreferenz.

```
# View Workflow details
> aws transfer describe-workflow --workflow-id w-1234567890abcdef0
{
    "Workflow": {
        "Arn": "arn:aws:transfer:us-east-1:111122223333:workflow/
w-1234567890abcdef0",
        "WorkflowId": "w-1234567890abcdef0",
        "Name": "Copy file to shared_files",
        "Steps": [
            {
```

```
                "Type": "COPY",
                "CopyStepDetails": {
                "Name": "Copy to shared",
                "FileLocation": {
                    "S3FileLocation": {
                        "Bucket": "DOC-EXAMPLE-BUCKET",
                        "Key": "home/shared_files/"
                    }
                }
                }
            }
        ],
        "OnException": {}
    }
}
```

Wenn Ihr Workflow als Teil eines AWS CloudFormation Stacks erstellt wurde, können Sie den Workflow über die AWS CloudFormation Konsole verwalten (https://console.aws.amazon.com/cloudformation).



# Verwenden Sie vordefinierte Schritte

Wenn Sie einen Workflow erstellen, können Sie wählen, ob Sie einen der folgenden vordefinierten Schritte hinzufügen möchten, die in diesem Thema beschrieben werden. Sie können sich auch dafür entscheiden, Ihre eigenen benutzerdefinierten Dateiverarbeitungsschritte hinzuzufügen. Weitere Informationen finden Sie unter the section called “Verwenden Sie benutzerdefinierte Schritte zur Dateiverarbeitung”.

Themen

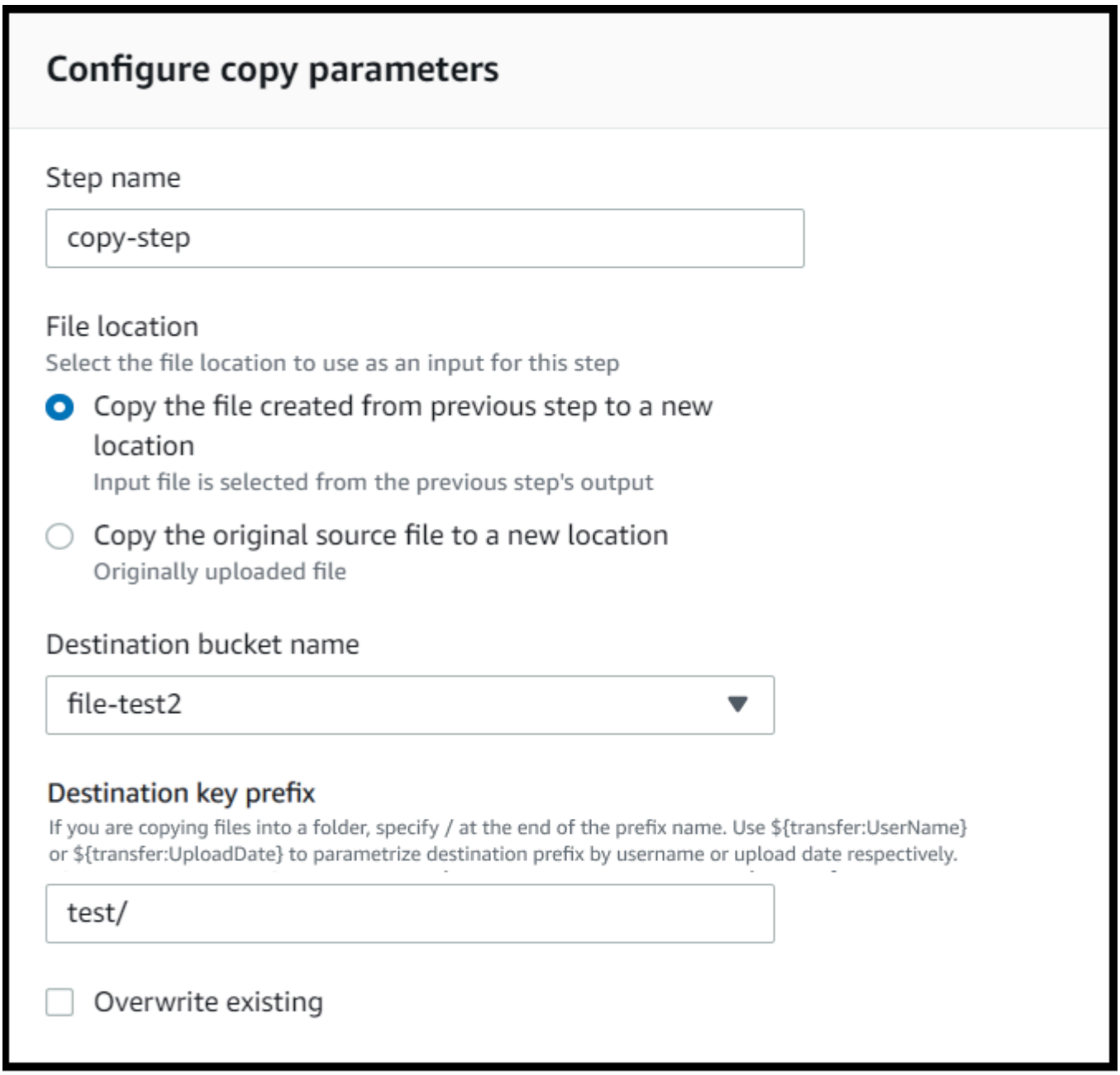

- Datei kopieren
- Datei entschlüsseln
- Datei kennzeichnen
- Datei löschen
- Benannte Variablen für Workflows
- Beispiel für einen Arbeitsablauf zum Markieren und Löschen


## Datei kopieren

Ein Schritt „Datei kopieren“ erstellt eine Kopie der hochgeladenen Datei an einem neuen Amazon S3 S3-Speicherort. Derzeit können Sie den Schritt „Datei kopieren“ nur mit Amazon S3 verwenden.

Der folgende Schritt zum Kopieren von Dateien kopiert Dateien in den `test` Ordner im `file-test` Ziel-Bucket.

Wenn der Schritt „Datei kopieren“ nicht der erste Schritt Ihres Workflows ist, können Sie den Speicherort der Datei angeben. Durch Angabe des Dateispeicherorts können Sie entweder die Datei, die im vorherigen Schritt verwendet wurde, oder die Originaldatei, die hochgeladen wurde, kopieren. Sie können diese Funktion verwenden, um mehrere Kopien der Originaldatei zu erstellen und gleichzeitig die Quelldatei für die Dateiarchivierung und Aufbewahrung von Aufzeichnungen intakt zu lassen. Ein Beispiel finden Sie unter Beispiel für einen Arbeitsablauf zum Markieren und Löschen.




## Geben Sie den Bucket und die wichtigsten Details an

Sie müssen den Bucket-Namen und einen Schlüssel für das Ziel des Schritts „Datei kopieren“ angeben. Der Schlüssel kann entweder ein Pfadname oder ein Dateiname sein. Ob der Schlüssel als Pfad- oder Dateiname behandelt wird, hängt davon ab, ob Sie den Schlüssel mit einem Schrägstrich (/) beenden.

Wenn das letzte Zeichen ist/, wird Ihre Datei in den Ordner kopiert, und ihr Name ändert sich nicht. Wenn das letzte Zeichen alphanumerisch ist, wird Ihre hochgeladene Datei in den Schlüsselwert

umbenannt. Wenn in diesem Fall bereits eine Datei mit diesem Namen existiert, hängt das Verhalten von der Einstellung für das Feld Bestehende überschreiben ab.

- Wenn Bestehende überschreiben ausgewählt ist, wird die vorhandene Datei durch die Datei ersetzt, die gerade verarbeitet wird.
- Wenn „Bestehende überschreiben“ nicht ausgewählt ist, passiert nichts und die Workflow-Verarbeitung wird gestoppt.

> **Tip**
>
> Wenn gleichzeitige Schreibvorgänge auf demselben Dateipfad ausgeführt werden, kann dies zu unerwartetem Verhalten beim Überschreiben von Dateien führen.

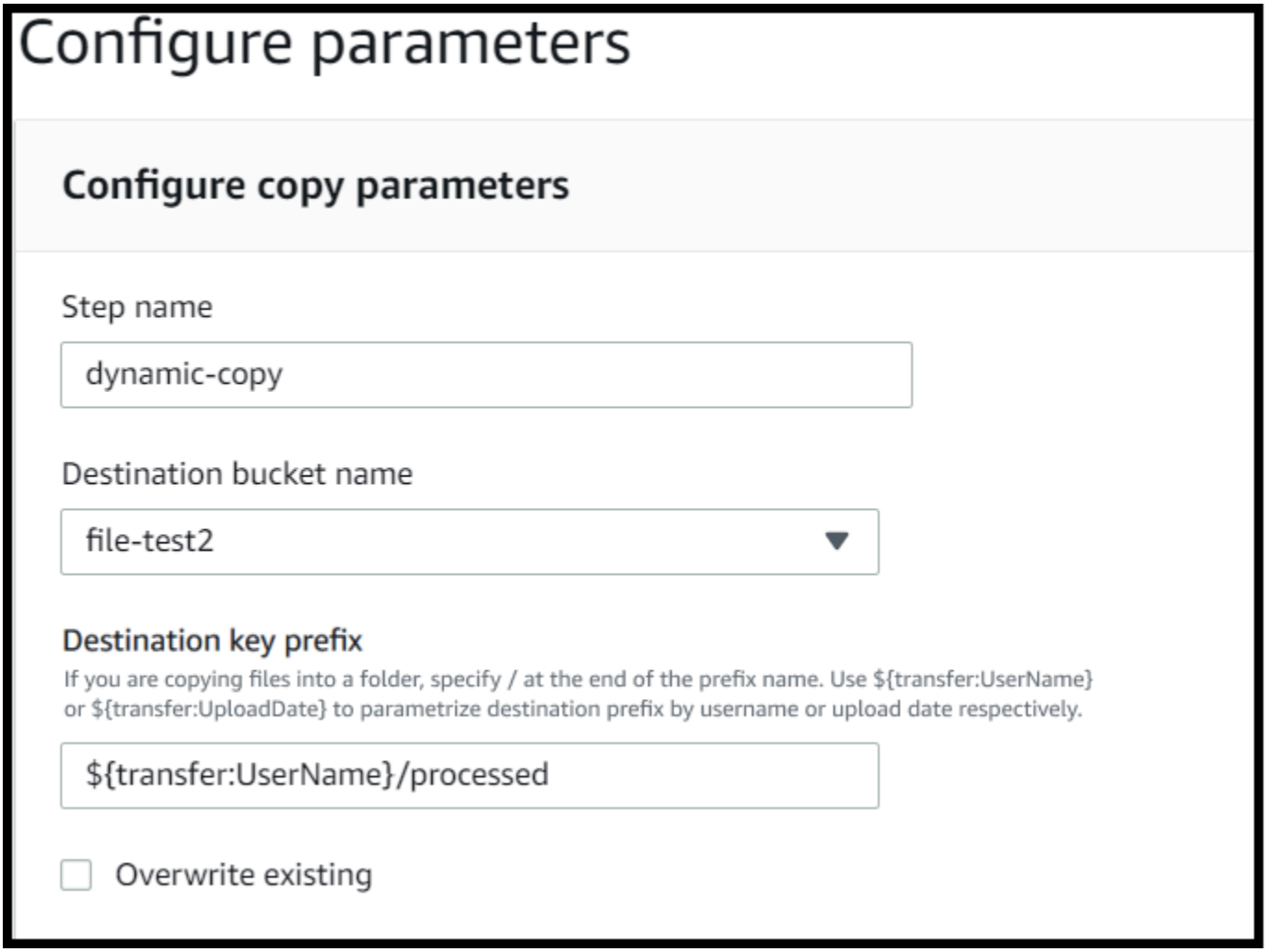
Wenn Ihr Schlüsselwert beispielsweise lautet`test/`, werden Ihre hochgeladenen Dateien in den `test` Ordner kopiert. Wenn Ihr Schlüsselwert ist (und Bestehende überschreiben ausgewählt ist)`test/today`, wird jede Datei, die Sie hochladen, in eine Datei mit dem Namen `today` im `test` Ordner kopiert, und jede nachfolgende Datei überschreibt die vorherige.

> **Note**
>
> Amazon S3 unterstützt Buckets und Objekt. Es gibt keine Hierarchie in Amazon S3. Sie können jedoch Präfixe und Trennzeichen in Objektschlüsselnamen verwenden, um eine Hierarchie zu implizieren und Ihre Daten ähnlich wie in Ordnern zu organisieren.

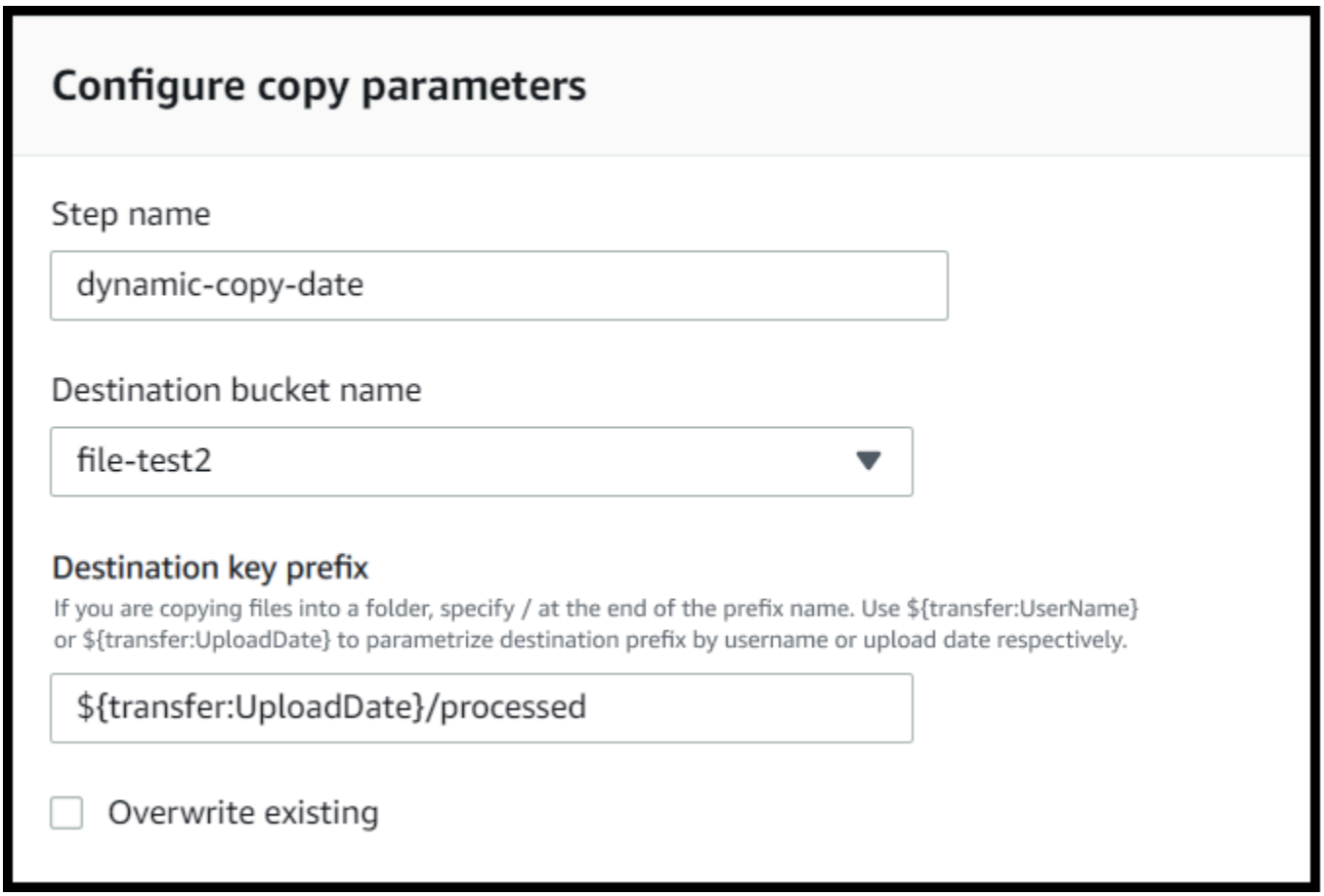
## Verwenden Sie eine benannte Variable in einem Schritt zum Kopieren von Dateien

In einem Schritt zum Kopieren von Dateien können Sie eine Variable verwenden, um Ihre Dateien dynamisch in benutzerspezifische Ordner zu kopieren. Derzeit können Sie `${transfer:UserName}` oder `${transfer:UploadDate}` als Variable verwenden, um Dateien an einen Zielort für den Benutzer zu kopieren, der gerade Dateien hochlädt, oder auf der Grundlage des aktuellen Datums.

Wenn der Benutzer im folgenden Beispiel eine Datei `richard-roe` hochlädt, wird sie in den `file-test2/richard-roe/processed/` Ordner kopiert. Wenn der Benutzer eine Datei `mary-major` hochlädt, wird sie in den `file-test2/mary-major/processed/` Ordner kopiert.




In ähnlicher Weise können Sie die Variable `${transfer:UploadDate}` als Variable verwenden, um Dateien an einen Zielort zu kopieren, der nach dem aktuellen Datum benannt ist. Wenn Sie im folgenden Beispiel das Ziel `${transfer:UploadDate}/processed` auf den 1. Februar 2022 festlegen, werden die hochgeladenen Dateien in den `file-test2/2022-02-01/processed/` Ordner kopiert.




Sie können diese beiden Variablen auch zusammen verwenden und so ihre Funktionalität kombinieren. Beispielsweise:

- Sie könnten das Zielschlüsselpräfix auf festlegen**`folder/${transfer:UserName}/${transfer:UploadDate}/`**, wodurch beispielsweise `folder/marymajor/2023-01-05/` verschachtelte Ordner erstellt würden.
- Sie könnten das Zielschlüsselpräfix auf setzen**`folder/${transfer:UserName}-${transfer:UploadDate}/`**, um beispielsweise die beiden Variablen zu verketten. `folder/marymajor-2023-01-05/`

## IAM-Berechtigungen für den Kopierschritt

Damit ein Kopierschritt erfolgreich ausgeführt werden kann, stellen Sie sicher, dass die Ausführungsrolle für Ihren Workflow die folgenden Berechtigungen enthält.

```
{
            "Sid": "ListBucket",
```

```
            "Effect": "Allow",
            "Action": "s3:ListBucket",
            "Resource": [
                "arn:aws:s3:::destination-bucket-name"
            ]
        },
        {
            "Sid": "HomeDirObjectAccess",
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
                "s3:DeleteObjectVersion",
                "s3:DeleteObject",
                "s3:GetObjectVersion"
            ],
            "Resource": "arn:aws:s3:::destination-bucket-name/*"
        }
```

> **Note**
>
> Die `s3:ListBucket` Berechtigung ist nur erforderlich, wenn Sie Bestehende überschreiben nicht auswählen. Mit dieser Berechtigung wird in Ihrem Bucket überprüft, ob bereits eine Datei mit demselben Namen existiert. Wenn Sie Existierende überschreiben ausgewählt haben, muss der Workflow nicht nach der Datei suchen, sondern kann sie einfach schreiben. Wenn Ihre Amazon S3 S3-Dateien Tags haben, müssen Sie Ihrer IAM-Richtlinie eine oder zwei Berechtigungen hinzufügen.
>
> - `s3:GetObjectTagging`Für eine Amazon S3 S3-Datei hinzufügen, die nicht versioniert ist.
> - `s3:GetObjectVersionTagging`Für eine Amazon S3 S3-Datei hinzufügen, die versioniert ist.

## Datei entschlüsseln

Der AWS Speicher-Blog enthält einen Beitrag, in dem beschrieben wird, wie Dateien mithilfe von Transfer Family Managed Workflows, Verschlüsseln und Entschlüsseln von Dateien mit PGP und einfach entschlüsselt werden können, ohne Code zu schreiben, beschrieben werden. AWS Transfer Family

## Verwenden Sie die PGP-Entschlüsselung in Ihrem Workflow

Transfer Family bietet integrierte Unterstützung für die Pretty Good Privacy (PGP) -Entschlüsselung. Sie können die PGP-Entschlüsselung für Dateien verwenden, die über SFTP, FTPS oder FTP auf Amazon Simple Storage Service (Amazon S3) oder Amazon Elastic File System (Amazon EFS) hochgeladen werden.

Um die PGP-Entschlüsselung verwenden zu können, müssen Sie die privaten PGP-Schlüssel erstellen und speichern, die für die Entschlüsselung Ihrer Dateien verwendet werden. Ihre Benutzer können dann Dateien mit den entsprechenden PGP-Verschlüsselungsschlüsseln verschlüsseln, bevor sie die Dateien auf Ihren Transfer Family Family-Server hochladen. Nachdem Sie die verschlüsselten Dateien erhalten haben, können Sie diese Dateien in Ihrem Workflow entschlüsseln. Ein detailliertes Tutorial finden Sie unter Einen verwalteten Workflow zum Entschlüsseln einer Datei einrichten.

Um die PGP-Entschlüsselung in Ihrem Workflow zu verwenden

1. Identifizieren Sie einen Transfer Family Family-Server, auf dem Ihr Workflow gehostet werden soll, oder erstellen Sie einen neuen. Sie benötigen die Server-ID, bevor Sie Ihre PGP-Schlüssel AWS Secrets Manager mit dem richtigen geheimen Namen speichern können.
2. Speichern Sie Ihren PGP-Schlüssel AWS Secrets Manager unter dem erforderlichen geheimen Namen. Details hierzu finden Sie unter PGP-Schlüssel verwalten. Workflows können anhand des geheimen Namens in Secrets Manager automatisch den richtigen PGP-Schlüssel finden, der für die Entschlüsselung verwendet werden soll.

   > **Note**
   >
   > Wenn Sie Geheimnisse im Secrets Manager speichern, AWS-Konto fallen Gebühren an. Informationen zu Preisen erhalten Sie unter AWS Secrets Manager -Preise.

3. Verschlüsseln Sie eine Datei mit Ihrem PGP-Schlüsselpaar. (Eine Liste der unterstützten Clients finden Sie unterUnterstützte PGP-Clients.) Wenn Sie die Befehlszeile verwenden, führen Sie den folgenden Befehl aus. Um diesen Befehl zu verwenden, ersetzen Sie ihn durch die E-Mail-Adresse, *username@example.com* mit der Sie das PGP-Schlüsselpaar erstellt haben. *testfile.txt*Ersetzen Sie es durch den Namen der Datei, die Sie verschlüsseln möchten.

   ```
   gpg -e -r username@example.com testfile.txt
   ```

4. Laden Sie die verschlüsselte Datei auf Ihren Transfer Family Family-Server hoch.

5. Konfigurieren Sie einen Entschlüsselungsschritt in Ihrem Workflow. Weitere Informationen finden Sie unter Fügen Sie einen Entschlüsselungsschritt hinzu.

## Fügen Sie einen Entschlüsselungsschritt hinzu

Ein Entschlüsselungsschritt entschlüsselt eine verschlüsselte Datei, die als Teil Ihres Workflows auf Amazon S3 oder Amazon EFS hochgeladen wurde. Einzelheiten zur Konfiguration der Entschlüsselung finden Sie unter. Verwenden Sie die PGP-Entschlüsselung in Ihrem Workflow

Wenn Sie Ihren Entschlüsselungsschritt für einen Workflow erstellen, müssen Sie das Ziel für die entschlüsselten Dateien angeben. Sie müssen auch auswählen, ob vorhandene Dateien überschrieben werden sollen, wenn am Zielspeicherort bereits eine Datei vorhanden ist. Mithilfe von Amazon CloudWatch Logs können Sie die Ergebnisse des Entschlüsselungsworkflows überwachen und Prüfprotokolle für jede Datei in Echtzeit abrufen.

Nachdem Sie den Dateityp Entschlüsseln für Ihren Schritt ausgewählt haben, wird die Seite „Parameter konfigurieren“ angezeigt. Geben Sie die Werte für den Abschnitt PGP-Entschlüsselungsparameter konfigurieren ein.

Die verfügbaren Optionen lauten wie folgt:

- Schrittname — Geben Sie einen aussagekräftigen Namen für den Schritt ein.
- Dateispeicherort — Durch Angabe des Dateispeicherorts können Sie entweder die Datei, die im vorherigen Schritt verwendet wurde, oder die Originaldatei, die hochgeladen wurde, entschlüsseln.

  > **Note**
  >
  > Dieser Parameter ist nicht verfügbar, wenn dieser Schritt der erste Schritt des Workflows ist.

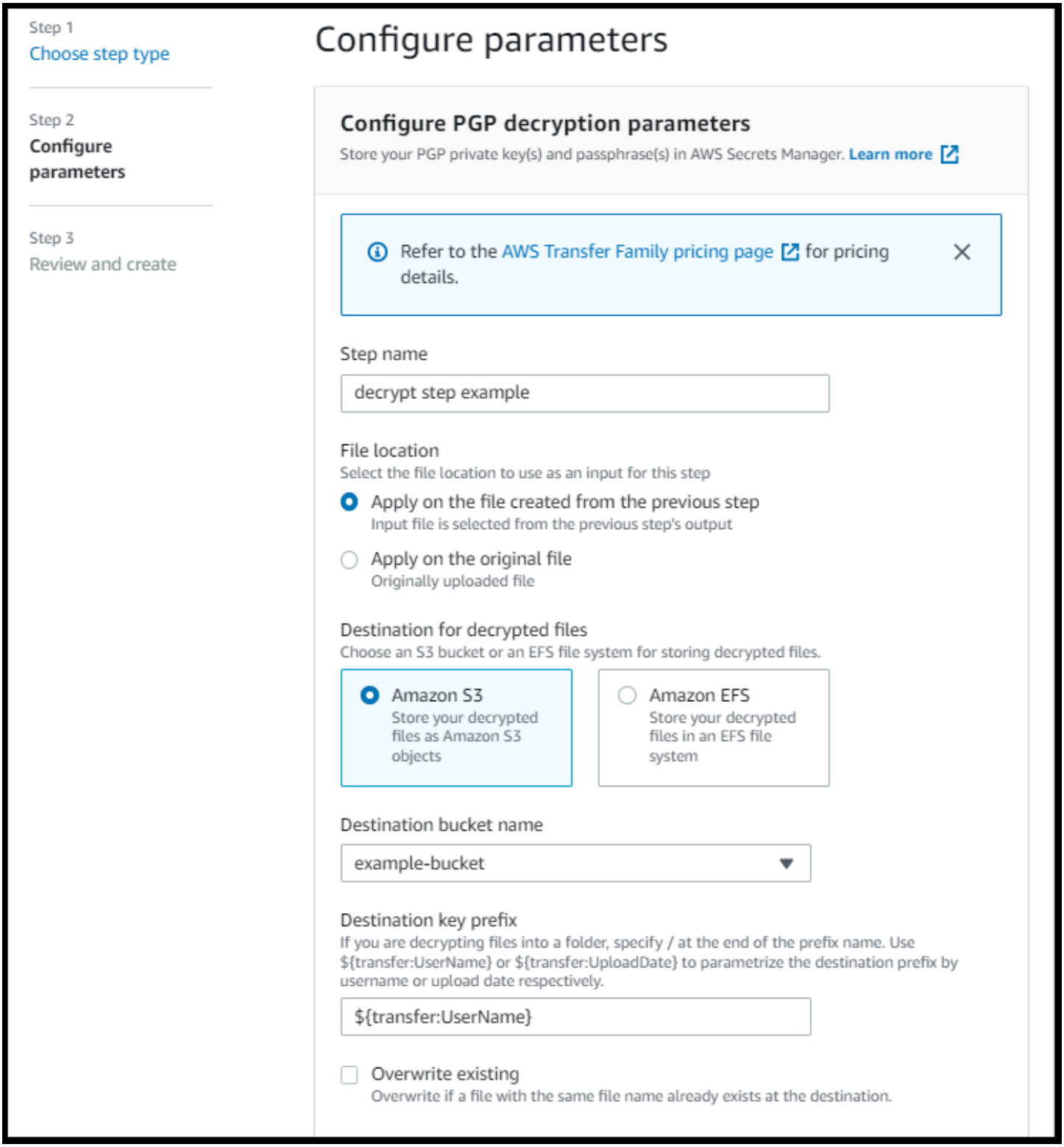
- Ziel für entschlüsselte Dateien — Wählen Sie einen Amazon S3 S3-Bucket oder ein Amazon EFS-Dateisystem als Ziel für die entschlüsselte Datei.
  - Wenn Sie sich für Amazon S3 entscheiden, müssen Sie einen Ziel-Bucket-Namen und ein Zielschlüsselpräfix angeben. Um das Zielschlüsselpräfix nach Benutzername zu parametrisieren, geben Sie **`${transfer:UserName}`** als Zielschlüsselpräfix ein. Um das Zielschlüsselpräfix anhand des Upload-Datums zu parametrisieren, geben **`${Transfer:UploadDate}`** Sie in ähnlicher Weise das Zielschlüsselpräfix ein.
  - Wenn Sie Amazon EFS wählen, müssen Sie ein Zieldateisystem und einen Pfad angeben.

> **Note**
>
> Die Speicheroption, die Sie hier auswählen, muss mit dem Speichersystem übereinstimmen, das vom Transfer Family Family-Server verwendet wird, mit dem dieser Workflow verknüpft ist. Andernfalls erhalten Sie eine Fehlermeldung, wenn Sie versuchen, diesen Workflow auszuführen.

- Bestehende überschreiben — Wenn Sie eine Datei hochladen und am Ziel bereits eine Datei mit demselben Dateinamen vorhanden ist, hängt das Verhalten von der Einstellung für diesen Parameter ab:
  - Wenn „Bestehende überschreiben“ ausgewählt ist, wird die bestehende Datei durch die Datei ersetzt, die gerade verarbeitet wird.
  - Wenn „Bestehende überschreiben“ nicht ausgewählt ist, passiert nichts und die Workflow-Verarbeitung wird gestoppt.

    > **Tip**
    >
    > Wenn gleichzeitige Schreibvorgänge auf demselben Dateipfad ausgeführt werden, kann dies zu unerwartetem Verhalten beim Überschreiben von Dateien führen.

Der folgende Screenshot zeigt ein Beispiel für die Optionen, die Sie für den Schritt „Datei entschlüsseln“ wählen könnten.


Step 1
Choose step type
Step 2
Configure parameters
Step 3
Review and create
Configure parameters
Configure PGP decryption parameters
Store your PGP private key(s) and passphrase(s) in AWS Secrets Manager. Learn more
Refer to the AWS Transfer Family pricing page for pricing details.
Step name
decrypt step example
File location
Select the file location to use as an input for this step
Apply on the file created from the previous step
Input file is selected from the previous step's output
Apply on the original file
Originally uploaded file
Destination for decrypted files
Choose an S3 bucket or an EFS file system for storing decrypted files.
Amazon S3
Store your decrypted files as Amazon S3 objects
Amazon EFS
Store your decrypted files in an EFS file system
Destination bucket name
example-bucket
Destination key prefix
If you are decrypting files into a folder, specify / at the end of the prefix name. Use ${transfer:UserName} or ${transfer:UploadDate} to parametrize the destination prefix by username or upload date respectively.
${transfer:UserName}
Overwrite existing
Overwrite if a file with the same file name already exists at the destination.

## Schritt „IAM-Berechtigungen für die Entschlüsselung“

Damit ein Entschlüsselungsschritt erfolgreich ausgeführt werden kann, stellen Sie sicher, dass die Ausführungsrolle für Ihren Workflow die folgenden Berechtigungen enthält.

```
{
            "Sid": "ListBucket",
            "Effect": "Allow",
            "Action": "s3:ListBucket",
            "Resource": [
                "arn:aws:s3:::destination-bucket-name"
            ]
        },
        {
            "Sid": "HomeDirObjectAccess",
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
                "s3:DeleteObjectVersion",
                "s3:DeleteObject",
                "s3:GetObjectVersion"
            ],
            "Resource": "arn:aws:s3:::destination-bucket-name/*"
        },
        {
            "Sid": "Decrypt",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue",
            ],
            "Resource": "arn:aws:secretsmanager:region:account-id:secret:aws/transfer/
*"
        }
```

> **Note**
>
> Die `s3:ListBucket` Berechtigung ist nur erforderlich, wenn Sie Bestehende überschreiben nicht auswählen. Mit dieser Berechtigung wird in Ihrem Bucket überprüft, ob bereits eine Datei mit demselben Namen existiert. Wenn Sie Existierende überschreiben ausgewählt haben, muss der Workflow nicht nach der Datei suchen, sondern kann sie einfach schreiben.

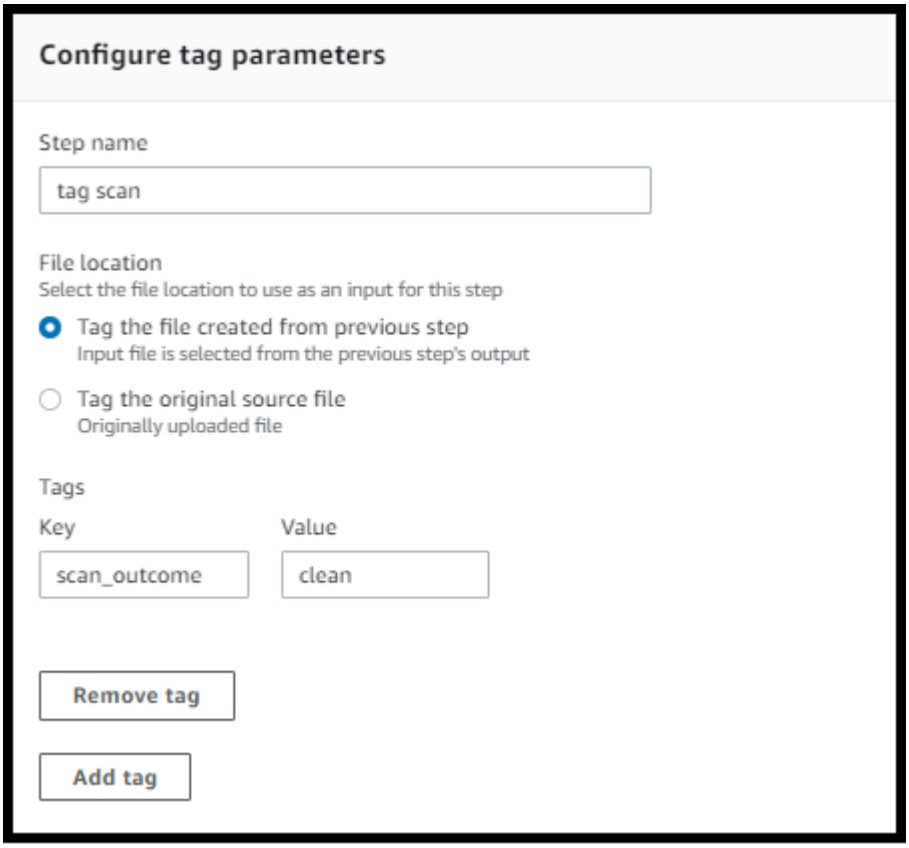
Wenn Ihre Amazon S3 S3-Dateien Tags haben, müssen Sie Ihrer IAM-Richtlinie eine oder zwei Berechtigungen hinzufügen.

- `s3:GetObjectTagging`Für eine Amazon S3 S3-Datei hinzufügen, die nicht versioniert ist.
- `s3:GetObjectVersionTagging`Für eine Amazon S3 S3-Datei hinzufügen, die versioniert ist.

## Datei kennzeichnen

Verwenden Sie einen Tag-Schritt, um eingehende Dateien für die weitere Verarbeitung zu kennzeichnen. Geben Sie den Wert des Tags ein, das Sie den eingehenden Dateien zuweisen möchten. Derzeit wird der Tag-Vorgang nur unterstützt, wenn Sie Amazon S3 für Ihren Transfer Family Family-Serverspeicher verwenden.

Der folgende Beispiel-Tag-Schritt weist `scan_outcome` und `clean` als Tag-Schlüssel bzw. -Wert zu.



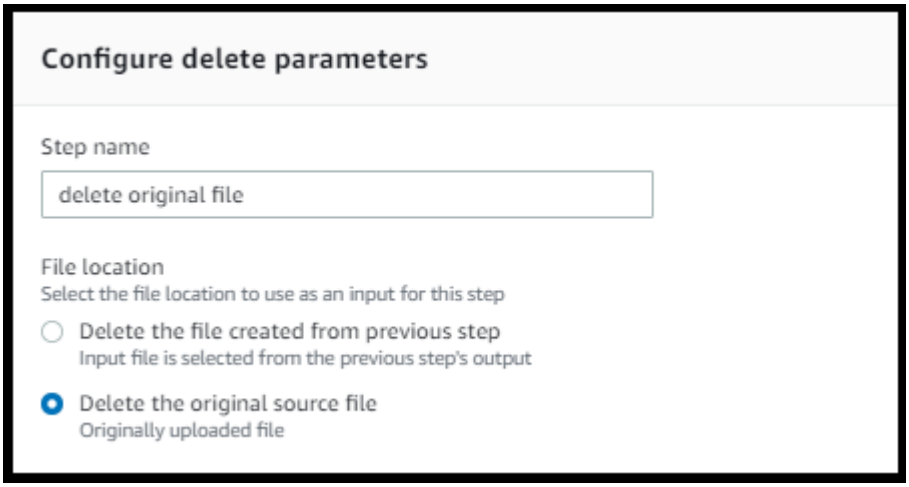

Damit ein Tag-Schritt erfolgreich ausgeführt werden kann, stellen Sie sicher, dass die Ausführungsrolle für Ihren Workflow die folgenden Berechtigungen enthält.

```
{
            "Sid": "Tag",
            "Effect": "Allow",
            "Action": [
                "s3:PutObjectTagging",
                "s3:PutObjectVersionTagging"
            ],
            "Resource": [
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
            ]
}
```

> **Note**
>
> Wenn Ihr Workflow einen Tag-Schritt enthält, der entweder vor einem Kopier- oder Entschlüsselungsschritt ausgeführt wird, müssen Sie Ihrer IAM-Richtlinie eine oder zwei Berechtigungen hinzufügen.
>
> - `s3:GetObjectTagging`Für eine Amazon S3 S3-Datei hinzufügen, die nicht versioniert ist.
> - `s3:GetObjectVersionTagging`Für eine Amazon S3 S3-Datei hinzufügen, die versioniert ist.

## Datei löschen

Um eine verarbeitete Datei aus einem vorherigen Workflow-Schritt oder die ursprünglich hochgeladene Datei zu löschen, verwenden Sie einen Schritt „Datei löschen“.



Damit ein Löschschritt erfolgreich ausgeführt werden kann, stellen Sie sicher, dass die Ausführungsrolle für Ihren Workflow die folgenden Berechtigungen enthält.

```
{
            "Sid": "Delete",
            "Effect": "Allow",
            "Action": [
                "s3:DeleteObjectVersion",
                "s3:DeleteObject"
            ],
            "Resource": "arn:aws:secretsmanager:region:account-ID:secret:aws/transfer/
*"
        }
```

## Benannte Variablen für Workflows

Für Schritte zum Kopieren und Entschlüsseln können Sie eine Variable verwenden, um Aktionen dynamisch auszuführen. AWS Transfer Family Unterstützt derzeit die folgenden benannten Variablen.

- Wird verwendet`${transfer:UserName}`, um Dateien an ein Ziel zu kopieren oder zu entschlüsseln, das auf dem Benutzer basiert, der die Dateien hochlädt.
- Wird verwendet`${transfer:UploadDate}`, um Dateien auf der Grundlage des aktuellen Datums an einen Zielort zu kopieren oder zu entschlüsseln.

## Beispiel für einen Arbeitsablauf zum Markieren und Löschen

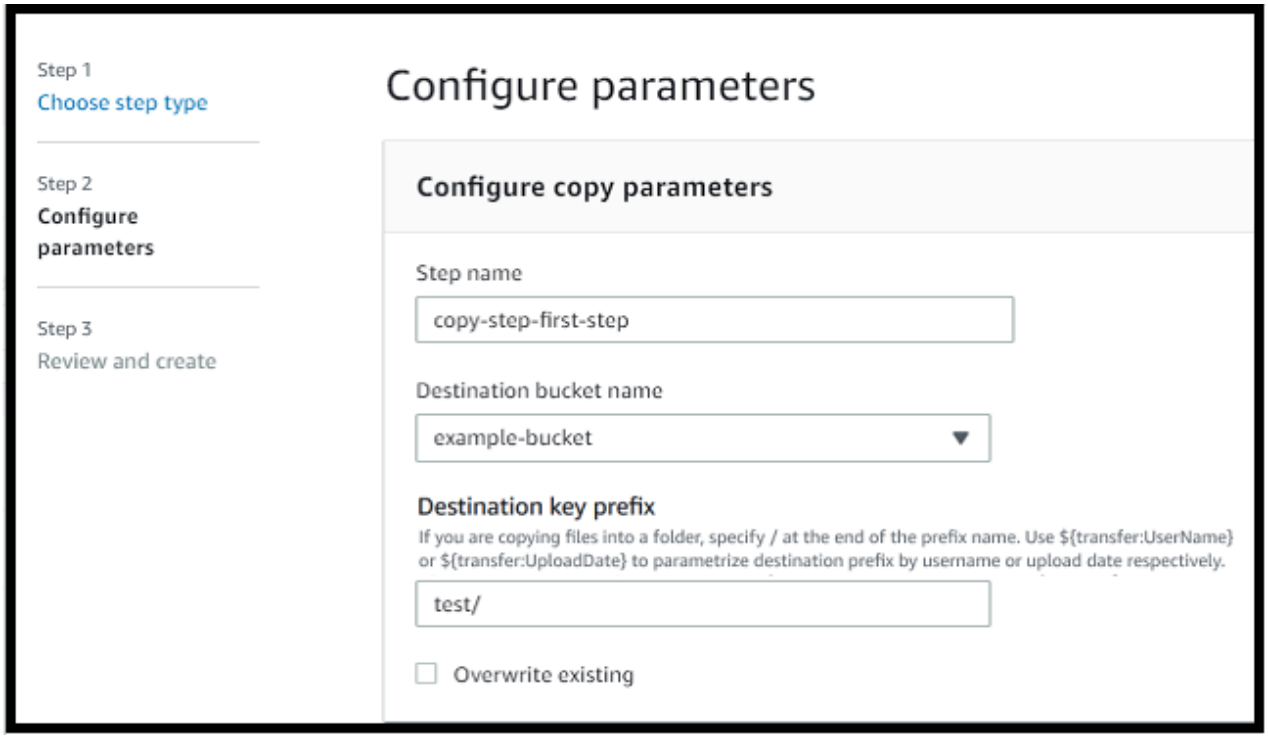
Das folgende Beispiel zeigt einen Workflow, der eingehende Dateien kennzeichnet, die von einer nachgelagerten Anwendung, z. B. einer Datenanalyseplattform, verarbeitet werden müssen. Nach dem Markieren der eingehenden Datei löscht der Workflow dann die ursprünglich hochgeladene Datei, um Speicherkosten zu sparen.

Console

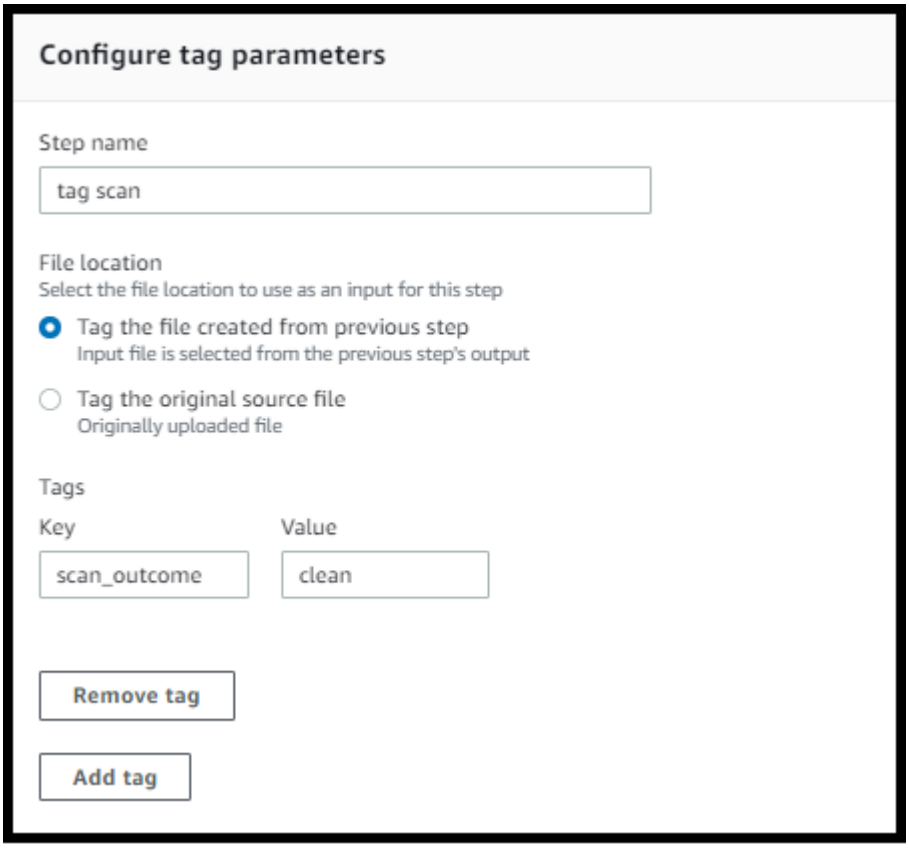
Beispiel für einen Arbeitsablauf zum Markieren und Verschieben

1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Wählen Sie im linken Navigationsbereich Workflows aus.

3. Wählen Sie auf der Seite Workflows die Option Workflow erstellen aus.
4. Geben Sie auf der Seite Workflow erstellen eine Beschreibung ein. Diese Beschreibung wird auf der Seite Workflows angezeigt.
5. Fügen Sie den ersten Schritt hinzu (Kopie).
   a. Wählen Sie im Abschnitt Nominale Schritte die Option Schritt hinzufügen aus.
   b. Wählen Sie „Datei kopieren“ und anschließend „Weiter“.
   c. Geben Sie einen Schrittnamen ein und wählen Sie dann einen Ziel-Bucket und ein key prefix aus.




   d. Wählen Sie Weiter und überprüfen Sie dann die Details für den Schritt.
   e. Wählen Sie Schritt erstellen, um den Schritt hinzuzufügen und fortzufahren.
6. Fügen Sie den zweiten Schritt (Tag) hinzu.
   a. Wählen Sie im Abschnitt Nominale Schritte die Option Schritt hinzufügen aus.
   b. Wählen Sie „Datei taggen“ und anschließend „Weiter“.
   c. Geben Sie einen Schrittnamen ein.
   d. Wählen Sie für Dateispeicherort die Option Datei kennzeichnen, die im vorherigen Schritt erstellt wurde.
   e. Geben Sie einen Schlüssel und einen Wert ein.



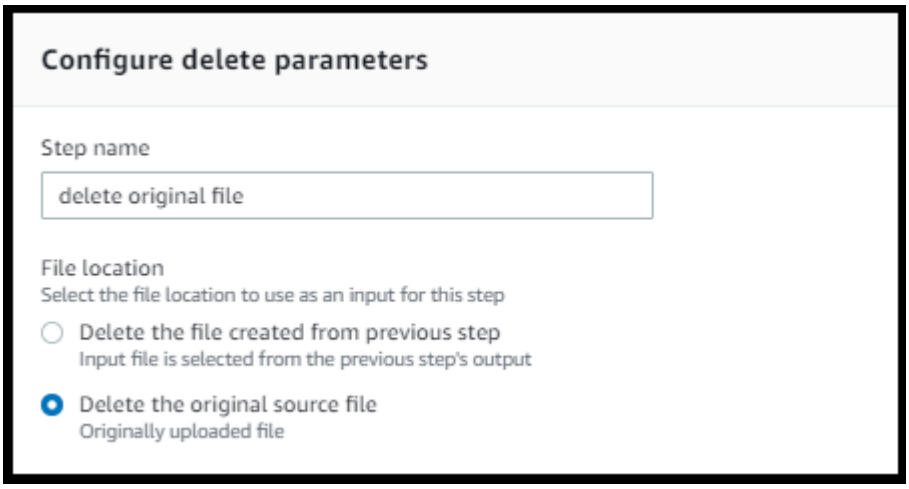

f. Wählen Sie Weiter und überprüfen Sie dann die Details für den Schritt.

g. Wählen Sie Schritt erstellen, um den Schritt hinzuzufügen und fortzufahren.

7. Fügen Sie den dritten Schritt hinzu (Löschen).

a. Wählen Sie im Abschnitt Nominale Schritte die Option Schritt hinzufügen aus.

b. Wählen Sie „Datei löschen“ und anschließend „Weiter“.




c. Geben Sie einen Schrittnamen ein.

d. Wählen Sie unter Dateispeicherort die Option Ursprüngliche Quelldatei löschen aus.

e. Wählen Sie Weiter und überprüfen Sie dann die Details für den Schritt.

f. Wählen Sie Schritt erstellen, um den Schritt hinzuzufügen und fortzufahren.

8. Überprüfen Sie die Workflow-Konfiguration und wählen Sie dann Workflow erstellen aus.

CLI

Beispiel für einen Workflow zum Markieren und Verschieben

1. Speichern Sie den folgenden Code in einer Datei, `tagAndMoveWorkflow.json` z. B. Ersetzen Sie jeden *`user input placeholder`* durch Ihre Informationen.

```
[
   {
       "Type": "COPY",
       "CopyStepDetails": {
          "Name": "CopyStep",
          "DestinationFileLocation": {
             "S3FileLocation": {
                "Bucket": "DOC-EXAMPLE-BUCKET",
                "Key": "test/"
             }
          }
       }
   },
   {
       "Type": "TAG",
       "TagStepDetails": {
          "Name": "TagStep",
          "Tags": [
             {
                "Key": "name",
                "Value": "demo"
             }
          ],
          "SourceFileLocation": "${previous.file}"
       }
   },
   {
      "Type": "DELETE",
      "DeleteStepDetails":{
```

```
        "Name":"DeleteStep",
        "SourceFileLocation": "${original.file}"
      }
  }
]
```

Im ersten Schritt wird die hochgeladene Datei an einen neuen Amazon S3 S3-Speicherort kopiert. Im zweiten Schritt wird der Datei (), die an den neuen Speicherort kopiert wurde, ein Tag (Schlüssel-Wert-Paar`previous.file`) hinzugefügt. Und schließlich löscht der dritte Schritt die Originaldatei (). `original.file`

2. Erstellen Sie einen Workflow aus der gespeicherten Datei. Ersetzen Sie jeden *`user input placeholder`* durch Ihre Informationen.

```
aws transfer create-workflow --description "short-description" --steps
 file://path-to-file --region region-ID
```

Beispielsweise:

```
aws transfer create-workflow --description "copy-tag-delete workflow" --steps
 file://tagAndMoveWorkflow.json --region us-east-1
```

> **Note**
>
> Weitere Informationen zur Verwendung von Dateien zum Laden von Parametern finden Sie unter So laden Sie Parameter aus einer Datei.

3. Aktualisieren Sie einen vorhandenen Server.

> **Note**
>
> In diesem Schritt wird davon ausgegangen, dass Sie bereits über einen Transfer Family Family-Server verfügen und diesem einen Workflow zuordnen möchten. Falls nicht, siehe Konfiguration eines SFTP-, FTPS- oder FTP-Serverendpunkts. Ersetzen Sie jeden *`user input placeholder`* durch Ihre Informationen.

```
aws transfer update-server --server-id server-ID --region region-ID
```

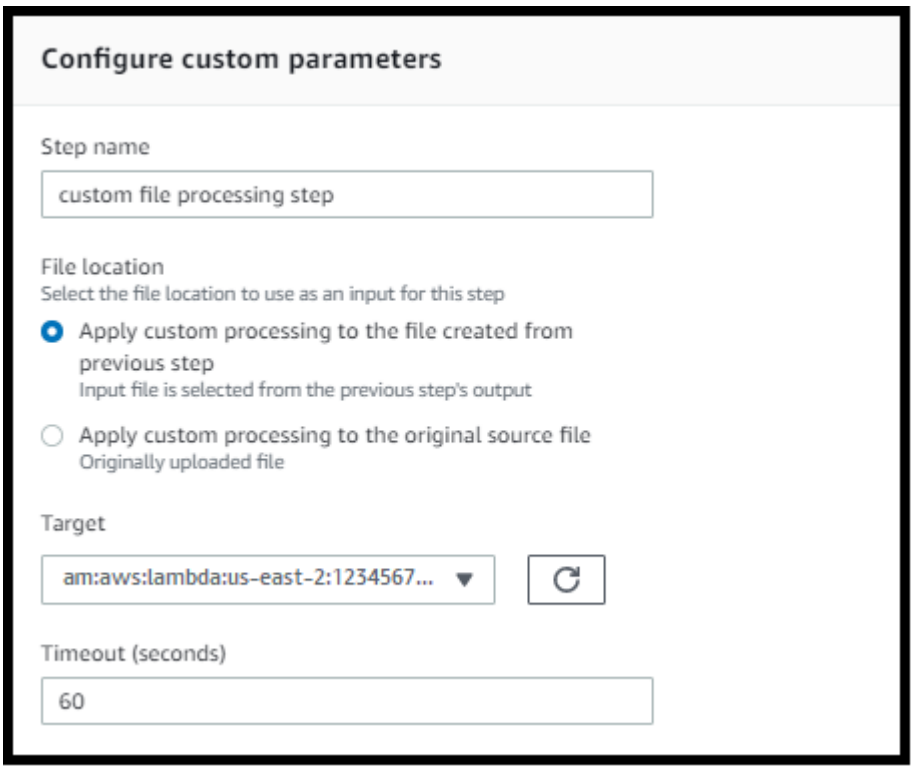
```
  --workflow-details '{"OnUpload":[{ "WorkflowId": "workflow-
ID","ExecutionRole": "execution-role-ARN"}]}'
```

Beispielsweise:

```
aws transfer update-server --server-id s-1234567890abcdef0 --region us-east-2
  --workflow-details '{"OnUpload":[{ "WorkflowId": "w-
abcdef01234567890","ExecutionRole": "arn:aws:iam::111111111111:role/nikki-wolf-
execution-role"}]}'
```

# Verwenden Sie benutzerdefinierte Schritte zur Dateiverarbeitung

Mithilfe eines benutzerdefinierten Dateiverarbeitungsschritts können Sie Ihre eigene Dateiverarbeitungslogik verwenden. AWS Lambda Beim Eintreffen einer Datei ruft ein Transfer Family Family-Server eine Lambda-Funktion auf, die benutzerdefinierte Dateiverarbeitungslogik enthält, z. B. das Verschlüsseln von Dateien, das Scannen nach Malware oder das Überprüfen auf falsche Dateitypen. Im folgenden Beispiel wird die AWS Lambda Zielfunktion verwendet, um die Ausgabedatei aus dem vorherigen Schritt zu verarbeiten.



> **Note**
>
> Eine Beispielfunktion für Lambda finden Sie unter Beispiel für eine Lambda-Funktion für einen benutzerdefinierten Workflow-Schritt. Beispiele für Ereignisse (einschließlich des Speicherorts für Dateien, die an das Lambda übergeben wurden) finden Sie unterBeispiele für Ereignisse, an die AWS Lambda beim Hochladen einer Datei gesendet werden.

Bei einem benutzerdefinierten Workflow-Schritt müssen Sie die Lambda-Funktion so konfigurieren, dass sie den SendWorkflowStepStateAPI-Vorgang aufruft. `SendWorkflowStepState`benachrichtigt die Workflow-Ausführung darüber, dass der Schritt entweder mit einem Erfolgs- oder einem Fehlerstatus abgeschlossen wurde. Der Status der `SendWorkflowStepState` API-Operation ruft einen Exception-Handler-Schritt oder einen nominalen Schritt in der linearen Sequenz auf, basierend auf dem Ergebnis der Lambda-Funktion.

Wenn die Lambda-Funktion ausfällt oder das Zeitlimit überschritten wird, schlägt der Schritt fehl, und das sehen Sie `StepErrored` in Ihren CloudWatch Protokollen. Wenn die Lambda-Funktion Teil des nominalen Schritts ist und die Funktion `SendWorkflowStepState` mit einem Timeout `Status="FAILURE"` oder einem Timeout reagiert, wird der Ablauf mit den Exception-Handler-Schritten fortgesetzt. In diesem Fall führt der Workflow die verbleibenden (falls vorhanden) nominalen Schritte nicht weiter aus. Weitere Details finden Sie unter Ausnahmebehandlung für einen Workflow.

Wenn Sie den `SendWorkflowStepState` API-Vorgang aufrufen, müssen Sie die folgenden Parameter senden:

```
{
    "ExecutionId": "string",
    "Status": "string",
    "Token": "string",
    "WorkflowId": "string"
}
```

Sie können das `ExecutionId`Token, und `WorkflowId` aus dem Eingabeereignis extrahieren, das bei der Ausführung der Lambda-Funktion übergeben wird (Beispiele werden in den folgenden Abschnitten gezeigt). Der `Status` Wert kann entweder oder `SUCCESS` sein. `FAILURE`

Um den `SendWorkflowStepState` API-Vorgang von Ihrer Lambda-Funktion aus aufrufen zu können, müssen Sie eine Version des AWS SDK verwenden, die nach der Einführung von Managed Workflows veröffentlicht wurde.

## Mehrere Lambda-Funktionen nacheinander verwenden

Wenn Sie mehrere benutzerdefinierte Schritte nacheinander verwenden, funktioniert die Option Dateispeicherort anders als wenn Sie nur einen einzigen benutzerdefinierten Schritt verwenden. Transfer Family unterstützt nicht die Rückgabe der mit Lambda verarbeiteten Datei, um sie als Eingabe für den nächsten Schritt zu verwenden. Wenn Sie also mehrere benutzerdefinierte Schritte haben, die alle für die Verwendung dieser `previous.file` Option konfiguriert sind, verwenden sie alle denselben Dateispeicherort (den Speicherort der Eingabedatei für den ersten benutzerdefinierten Schritt).

> **Note**
>
> Die `previous.file` Einstellung funktioniert auch anders, wenn Sie nach einem benutzerdefinierten Schritt einen vordefinierten Schritt (kennzeichnen, kopieren, entschlüsseln oder löschen) haben. Wenn der vordefinierte Schritt so konfiguriert ist, dass er die `previous.file` Einstellung verwendet, verwendet der vordefinierte Schritt dieselbe Eingabedatei wie der benutzerdefinierte Schritt. Die verarbeitete Datei aus dem benutzerdefinierten Schritt wird nicht an den vordefinierten Schritt übergeben.

## Zugreifen auf eine Datei nach der benutzerdefinierten Verarbeitung

Wenn Sie Amazon S3 als Speicher verwenden und Ihr Workflow einen benutzerdefinierten Schritt umfasst, der Aktionen an der ursprünglich hochgeladenen Datei ausführt, können nachfolgende Schritte nicht auf diese verarbeitete Datei zugreifen. Das heißt, jeder Schritt nach dem benutzerdefinierten Schritt kann nicht auf die aktualisierte Datei aus der Ausgabe des benutzerdefinierten Schritts verweisen.

Nehmen wir zum Beispiel an, dass Sie die folgenden drei Schritte in Ihrem Workflow haben.

- Schritt 1 — Laden Sie eine Datei mit dem Namen hoch`example-file.txt`.
- Schritt 2 — Rufen Sie eine Lambda-Funktion auf, die sich `example-file.txt` in irgendeiner Weise ändert.
- Schritt 3 — Versuchen Sie, eine weitere Verarbeitung mit der aktualisierten Version von durchzuführen. `example-file.txt`

Wenn Sie das `sourceFileLocation` für Schritt 3 so konfigurieren`${original.file}`, verwendet Schritt 3 den ursprünglichen Speicherort der Datei aus dem Zeitpunkt, zu dem der Server die Datei in Schritt 1 in den Speicher hochgeladen hat. Wenn Sie `${previous.file}` für Schritt 3 verwenden, wird in Schritt 3 der Speicherort wiederverwendet, den Schritt 2 als Eingabe verwendet hat.

Daher verursacht Schritt 3 einen Fehler. Wenn in Schritt 3 beispielsweise versucht wird, das Update zu kopieren`example-file.txt`, wird die folgende Fehlermeldung angezeigt:

```
{
    "type": "StepErrored",
    "details": {
        "errorType": "NOT_FOUND",
        "errorMessage": "ETag constraint not met (Service: null; Status Code: 412;
 Error Code: null; Request ID: null; S3 Extended Request ID: null; Proxy: null)",
        "stepType": "COPY",
        "stepName": "CopyFile"
    },
```

Dieser Fehler tritt auf, weil der benutzerdefinierte Schritt das Entity-Tag (ETag) für `example-file.txt` so ändert, dass es nicht mit der Originaldatei übereinstimmt.

> **Note**
>
> Dieses Verhalten tritt nicht auf, wenn Sie Amazon EFS verwenden, da Amazon EFS keine Entitäts-Tags zur Identifizierung von Dateien verwendet.

## Beispiele für Ereignisse, an die AWS Lambda beim Hochladen einer Datei gesendet werden

Die folgenden Beispiele zeigen die Ereignisse, an die gesendet werden, AWS Lambda wenn ein Datei-Upload abgeschlossen ist. Ein Beispiel verwendet einen Transfer Family Family-Server, auf dem die Domain mit Amazon S3 konfiguriert ist. Das andere Beispiel verwendet einen Transfer Family Family-Server, auf dem die Domain Amazon EFS verwendet.

Custom step that uses an Amazon S3 domain

```
{
    "token": "MzI0Nzc4ZDktMGRmMi00MjFhLTgxMjUtYWZmZmRmODNkYjc0",
```

```
    "serviceMetadata": {
        "executionDetails": {
            "workflowId": "w-1234567890example",
            "executionId": "abcd1234-aa11-bb22-cc33-abcdef123456"
        },
        "transferDetails": {
            "sessionId": "36688ff5d2deda8c",
            "userName": "myuser",
            "serverId": "s-example1234567890"
        }
    },
    "fileLocation": {
        "domain": "S3",
        "bucket": "DOC-EXAMPLE-BUCKET",
        "key": "path/to/mykey",
        "eTag": "d8e8fca2dc0f896fd7cb4cb0031ba249",
        "versionId": null
    }
}
```

Custom step that uses an Amazon EFS domain

```
{
    "token": "MTg0N2Y3N2UtNWI5Ny00ZmZlLTk5YTgtZTU3YzViYjllNmZm",
    "serviceMetadata": {
        "executionDetails": {
            "workflowId": "w-1234567890example",
            "executionId": "abcd1234-aa11-bb22-cc33-abcdef123456"
        },
        "transferDetails": {
            "sessionId": "36688ff5d2deda8c",
            "userName": "myuser",
            "serverId": "s-example1234567890"
        }
    },
    "fileLocation": {
        "domain": "EFS",
        "fileSystemId": "fs-1234567",
        "path": "/path/to/myfile"
    }
}
```

## Beispiel für eine Lambda-Funktion für einen benutzerdefinierten Workflow-Schritt

Die folgende Lambda-Funktion extrahiert die Informationen zum Ausführungsstatus und ruft dann den SendWorkflowStepStateAPI-Vorgang auf, um den Status für den Schritt an den Workflow zurückzugeben — SUCCESS entweder oder. FAILURE Bevor Ihre Funktion den SendWorkflowStepState API-Vorgang aufruft, können Sie Lambda so konfigurieren, dass eine Aktion auf der Grundlage Ihrer Workflow-Logik ausgeführt wird.

```
import json
import boto3

transfer = boto3.client('transfer')

def lambda_handler(event, context):
    print(json.dumps(event))

    # call the SendWorkflowStepState API to notify the workflow about the step's
 SUCCESS or FAILURE status
    response = transfer.send_workflow_step_state(
        WorkflowId=event['serviceMetadata']['executionDetails']['workflowId'],
        ExecutionId=event['serviceMetadata']['executionDetails']['executionId'],
        Token=event['token'],
        Status='SUCCESS|FAILURE'
    )

    print(json.dumps(response))

    return {
      'statusCode': 200,
      'body': json.dumps(response)
    }
```

## IAM-Berechtigungen für einen benutzerdefinierten Schritt

Damit ein Schritt, der ein Lambda aufruft, erfolgreich sein kann, stellen Sie sicher, dass die Ausführungsrolle für Ihren Workflow die folgenden Berechtigungen enthält.

```
{
            "Sid": "Custom",
            "Effect": "Allow",
```

```
            "Action": [
                "lambda:InvokeFunction"
            ],
            "Resource": [
                "arn:aws:lambda:region:account-id:function:function-name"
            ]
        }
```

# IAM-Richtlinien für Workflows

Wenn Sie einem Server einen Workflow hinzufügen, müssen Sie eine Ausführungsrolle auswählen. Der Server verwendet diese Rolle, wenn er den Workflow ausführt. Wenn die Rolle nicht über die richtigen Berechtigungen verfügt, AWS Transfer Family kann der Workflow nicht ausgeführt werden.

In diesem Abschnitt wird ein möglicher Satz von AWS Identity and Access Management (IAM-) Berechtigungen beschrieben, mit denen Sie einen Workflow ausführen können. Weitere Beispiele werden später in diesem Thema beschrieben.

> **Note**
>
> Wenn Ihre Amazon S3 S3-Dateien Tags haben, müssen Sie Ihrer IAM-Richtlinie eine oder zwei Berechtigungen hinzufügen.
>
> - `s3:GetObjectTagging`Für eine Amazon S3 S3-Datei hinzufügen, die nicht versioniert ist.
> - `s3:GetObjectVersionTagging`Für eine Amazon S3 S3-Datei hinzufügen, die versioniert ist.

Um eine Ausführungsrolle für Ihren Workflow zu erstellen

1. Erstellen Sie eine neue IAM-Rolle und fügen Sie der Rolle die AWS verwaltete Richtlinie `AWSTransferFullAccess` hinzu. Weitere Informationen zum Erstellen einer neuen IAM-Rolle finden Sie unter. the section called “Erstellen Sie eine IAM-Rolle und -Richtlinie”
2. Erstellen Sie eine weitere Richtlinie mit den folgenden Berechtigungen und fügen Sie sie Ihrer Rolle hinzu. Ersetzen Sie jeden *`user input placeholder`* durch Ihre Informationen.

   ```
   {
       "Version": "2012-10-17",
   ```

```
    "Statement": [
        {
            "Sid": "ConsoleAccess",
            "Effect": "Allow",
            "Action": "s3:GetBucketLocation",
            "Resource": "*"
        },
        {
            "Sid": "ListObjectsInBucket",
            "Effect": "Allow",
            "Action": "s3:ListBucket",
            "Resource": [
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET"
            ]
        },
        {
            "Sid": "AllObjectActions",
            "Effect": "Allow",
            "Action": "s3:*Object",
            "Resource": [
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
            ]
        },
        {
            "Sid": "GetObjectVersion",
            "Effect": "Allow",
            "Action": "s3:GetObjectVersion",
            "Resource": [
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
            ]
        },
        {
            "Sid": "Custom",
            "Effect": "Allow",
            "Action": [
                "lambda:InvokeFunction"
            ],
            "Resource": [
                "arn:aws:lambda:region:account-id:function:function-name"
            ]
        },
        {
            "Sid": "Tag",
            "Effect": "Allow",
```

```
            "Action": [
                "s3:PutObjectTagging",
                "s3:PutObjectVersionTagging"
            ],
            "Resource": [
                "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
            ]
        }
    ]
}
```

3. Speichern Sie diese Rolle und geben Sie sie als Ausführungsrolle an, wenn Sie einem Server einen Workflow hinzufügen.

> **Note**
>
> AWS empfiehlt, bei der Erstellung von IAM-Rollen den Zugriff auf Ihre Ressourcen so weit wie möglich für Ihren Workflow einzuschränken.

## Vertrauensbeziehungen im Arbeitsablauf

Rollen für die Workflow-Ausführung erfordern auch eine Vertrauensbeziehung mit`transfer.amazonaws.com`. Informationen zum Einrichten einer Vertrauensbeziehung für AWS Transfer Family finden Sie unterSo stellen Sie eine Vertrauensbeziehung her.

Während Sie Ihr Vertrauensverhältnis aufbauen, können Sie auch Maßnahmen ergreifen, um das Problem des verwirrten Stellvertreters zu vermeiden. Eine Beschreibung dieses Problems sowie Beispiele, wie es vermieden werden kann, finden Sie unterthe section called “Serviceübergreifende Confused-Deputy-Prävention”.

## Beispiel für eine Ausführungsrolle: Entschlüsseln, Kopieren und Markieren

Wenn Sie Workflows haben, die Schritte zum Markieren, Kopieren und Entschlüsseln beinhalten, können Sie die folgende IAM-Richtlinie verwenden. Ersetzen Sie jeden *`user input placeholder`* durch Ihre Informationen.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
```

```
            "Sid": "CopyRead",
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:GetObjectTagging",
                "s3:GetObjectVersionTagging"
            ],
            "Resource": "arn:aws:s3:::source-bucket-name/*"
        },
        {
            "Sid": "CopyWrite",
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:PutObjectTagging"
            ],
            "Resource": "arn:aws:s3:::destination-bucket-name/*"
        },
        {
            "Sid": "CopyList",
            "Effect": "Allow",
            "Action": "s3:ListBucket",
            "Resource": [
                "arn:aws:s3:::source-bucket-name",
                "arn:aws:s3:::destination-bucket-name"
            ]
        },
        {
            "Sid": "Tag",
            "Effect": "Allow",
            "Action": [
                "s3:PutObjectTagging",
                "s3:PutObjectVersionTagging"
            ],
            "Resource": "arn:aws:s3:::destination-bucket-name/*",
            "Condition": {
                "StringEquals": {
                    "s3:RequestObjectTag/Archive": "yes"
                }
            }
        },
        {
            "Sid": "ListBucket",
            "Effect": "Allow",
```

```
            "Action": "s3:ListBucket",
            "Resource": [
                "arn:aws:s3:::destination-bucket-name"
            ]
        },
        {
            "Sid": "HomeDirObjectAccess",
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
                "s3:DeleteObjectVersion",
                "s3:DeleteObject",
                "s3:GetObjectVersion"
            ],
            "Resource": "arn:aws:s3:::destination-bucket-name/*"
        },
        {
            "Sid": "Decrypt",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue"
            ],
            "Resource": "arn:aws:secretsmanager:region:account-ID:secret:aws/transfer/
*"
        }
    ]
}
```

## Beispiel für eine Ausführungsrolle: Funktion ausführen und löschen

In diesem Beispiel haben Sie einen Workflow, der eine AWS Lambda Funktion aufruft. Wenn der Workflow die hochgeladene Datei löscht und über einen Exception-Handler-Schritt verfügt, der auf eine fehlgeschlagene Workflow-Ausführung im vorherigen Schritt reagiert, verwenden Sie die folgende IAM-Richtlinie. Ersetzen Sie jeden *user input placeholder* durch Ihre Informationen.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Delete",
            "Effect": "Allow",
            "Action": [
```

```
                "s3:DeleteObject",
                "s3:DeleteObjectVersion"
            ],
            "Resource": "arn:aws:s3:::bucket-name"
        },
        {
            "Sid": "Custom",
            "Effect": "Allow",
            "Action": [
                "lambda:InvokeFunction"
            ],
            "Resource": [
                "arn:aws:lambda:region:account-id:function:function-name"
            ]
        }
    ]
}
```

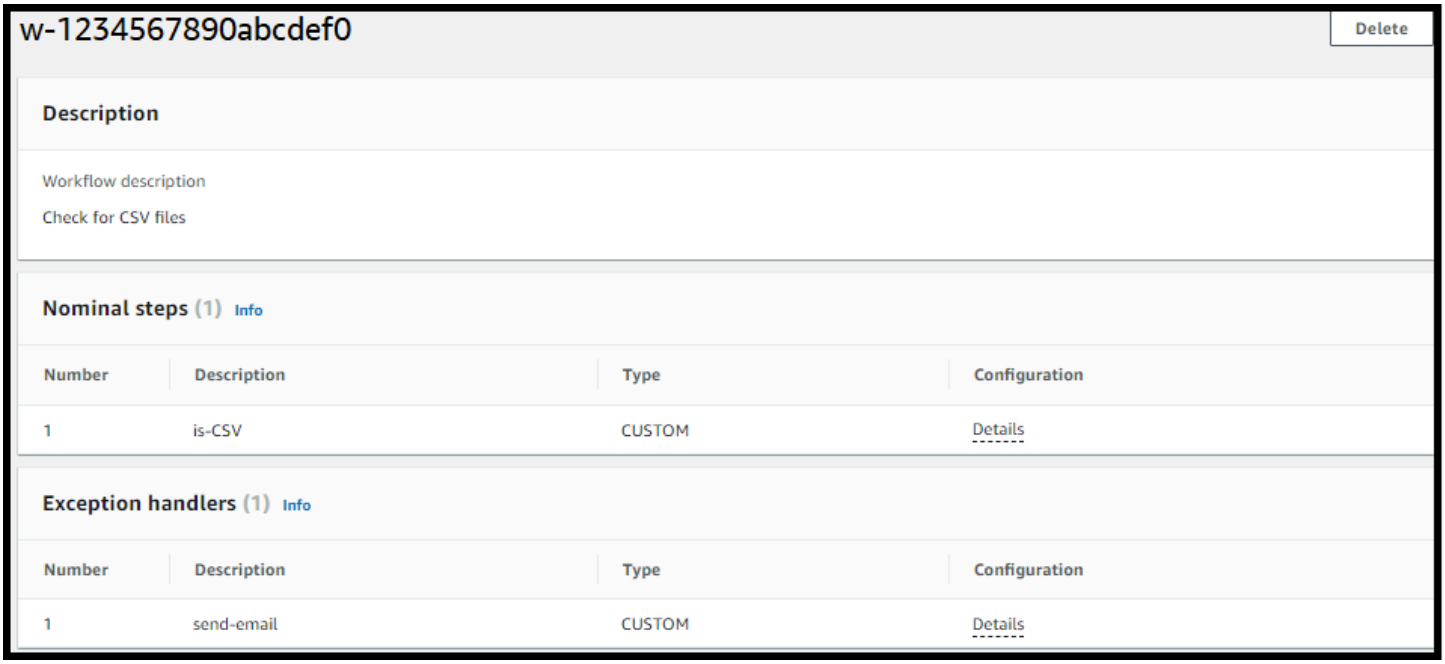
## Ausnahmebehandlung für einen Workflow

Wenn während der Ausführung eines Workflows Fehler auftreten, werden die von Ihnen angegebenen Schritte zur Ausnahmebehandlung ausgeführt. Sie geben die Schritte zur Fehlerbehandlung für einen Workflow auf die gleiche Weise an, wie Sie die nominalen Schritte für den Workflow angeben. Nehmen wir beispielsweise an, Sie haben die benutzerdefinierte Verarbeitung in nominalen Schritten konfiguriert, um eingehende Dateien zu überprüfen. Wenn die Dateiüberprüfung fehlschlägt, kann in einem Schritt zur Ausnahmebehandlung eine E-Mail an den Administrator gesendet werden.

Der folgende Beispiel-Workflow umfasst zwei Schritte:

- Ein nominaler Schritt, der überprüft, ob die hochgeladene Datei im CSV-Format vorliegt
- Ein Schritt zur Ausnahmebehandlung, bei dem eine E-Mail gesendet wird, falls die hochgeladene Datei nicht im CSV-Format vorliegt und der nominelle Schritt fehlschlägt

Um den Schritt zur Ausnahmebehandlung einzuleiten, muss die AWS Lambda Funktion im nominalen Schritt mit antworten. `Status="FAILURE"` Weitere Informationen zur Fehlerbehandlung in Workflows finden Sie unter. the section called “Verwenden Sie benutzerdefinierte Schritte zur Dateiverarbeitung”




# Überwachen Sie die Workflow-Ausführung

Amazon CloudWatch überwacht Ihre AWS Ressourcen und die Anwendungen, die Sie ausführen, AWS Cloud in Echtzeit. Sie können Amazon verwenden, CloudWatch um Metriken zu sammeln und zu verfolgen. Dabei handelt es sich um Variablen, die Sie für Ihre Workflows messen können. Sie können Workflow-Metriken und konsolidierte Protokolle mithilfe von Amazon anzeigen CloudWatch.

## CloudWatch Protokollierung für einen Workflow

CloudWatch bietet eine konsolidierte Prüfung und Protokollierung des Fortschritts und der Ergebnisse von Workflows.

CloudWatch Amazon-Protokolle für Workflows anzeigen

1. Öffnen Sie die CloudWatch Amazon-Konsole unter https://console.aws.amazon.com/cloudwatch/.
2. Wählen Sie im linken Navigationsbereich Logs und anschließend Log groups aus.
3. Wählen Sie auf der Seite Protokollgruppen in der Navigationsleiste die richtige Region für Ihren AWS Transfer Family Server aus.
4. Wählen Sie die Protokollgruppe aus, die Ihrem Server entspricht.

Wenn Ihre Server-ID beispielsweise lautets-1234567890abcdef0, ist Ihre Protokollgruppe/aws/transfer/s-1234567890abcdef0.

5. Auf der Seite mit den Protokollgruppendetails für Ihren Server werden die neuesten Protokollstreams angezeigt. Es gibt zwei Protokollstreams für den Benutzer, den Sie untersuchen:

   - Einer für jede Secure Shell (SSH) File Transfer Protocol (SFTP) -Sitzung.
   - Eine für den Workflow, der für Ihren Server ausgeführt wird. Das Format für den Protokollstream für den Workflow ist*username*.*workflowID*.*uniqueStreamSuffix*.

   Wenn Ihr Benutzer beispielsweise istmary-major, haben Sie die folgenden Protokollstreams:

   ```
   mary-major-east.1234567890abcdef0
   mary.w-abcdef01234567890.021345abcdef6789
   ```

   > **Note**
   >
   > Die in diesem Beispiel aufgeführten 16-stelligen alphanumerischen Identifikatoren sind fiktiv. Die Werte, die Sie bei Amazon sehen, CloudWatch sind unterschiedlich.

Auf der Seite Ereignisse protokollieren für mary-major-usa-east.1234567890abcdef0 werden die Details für jede Benutzersitzung angezeigt, und der mary.w-abcdef01234567890.021345abcdef6789 Protokollstream enthält die Details für den Workflow.

Im Folgenden finden Sie ein Beispiel für einen Protokollstream fürmary.w-abcdef01234567890.021345abcdef6789, der auf einem Workflow (w-abcdef01234567890) basiert, der einen Kopierschritt enthält.

```
{
    "type": "ExecutionStarted",
    "details": {
        "input": {
            "initialFileLocation": {
                "bucket": "DOC-EXAMPLE-BUCKET",
                "key": "mary/workflowSteps2.json",
                "versionId": "version-id",
                "etag": "etag-id"
```

```
            }
        }
    },
    "workflowId":"w-abcdef01234567890",
    "executionId":"execution-id",
    "transferDetails": {
        "serverId":"s-server-id",
        "username":"mary",
        "sessionId":"session-id"
    }
},
{
    "type":"StepStarted",
    "details": {
        "input": {
            "fileLocation": {
                "backingStore":"S3",
                "bucket":"DOC-EXAMPLE-BUCKET",
                "key":"mary/workflowSteps2.json",
                "versionId":"version-id",
                "etag":"etag-id"
            }
        },
        "stepType":"COPY",
        "stepName":"copyToShared"
    },
    "workflowId":"w-abcdef01234567890",
    "executionId":"execution-id",
    "transferDetails": {
        "serverId":"s-server-id",
        "username":"mary",
        "sessionId":"session-id"
    }
},
{
    "type":"StepCompleted",
    "details":{
        "output":{},
        "stepType":"COPY",
        "stepName":"copyToShared"
    },
    "workflowId":"w-abcdef01234567890",
    "executionId":"execution-id",
    "transferDetails":{
```

```
        "serverId":"server-id",
        "username":"mary",
        "sessionId":"session-id"
    }
},
{
    "type":"ExecutionCompleted",
    "details": {},
    "workflowId":"w-abcdef01234567890",
    "executionId":"execution-id",
    "transferDetails":{
        "serverId":"s-server-id",
        "username":"mary",
        "sessionId":"session-id"
    }
}
```

## CloudWatch Metriken für Workflows

AWS Transfer Family bietet mehrere Metriken für Workflows. Sie können Messwerte darüber anzeigen, wie viele Workflow-Ausführungen in der letzten Minute gestartet, erfolgreich abgeschlossen und fehlgeschlagen sind. Alle CloudWatch Metriken für Transfer Family werden unter beschrieben CloudWatch Metriken für Transfer Family verwenden.

# Erstellen Sie einen Workflow aus einer Vorlage

Sie können einen AWS CloudFormation Stapel bereitstellen, der aus einer Vorlage einen Workflow und einen Server erstellt. Dieses Verfahren enthält ein Beispiel, mit dem Sie schnell einen Workflow bereitstellen können.

Um einen AWS CloudFormation Stack zu erstellen, der einen AWS Transfer Family Workflow und einen Server erstellt

1. Öffnen Sie die AWS CloudFormation Konsole unter https://console.aws.amazon.com/cloudformation.
2. Speichern Sie den folgenden Code in einer Datei.

   YAML

   ```
   AWSTemplateFormatVersion: 2010-09-09
   Resources:
   ```

```
SFTPServer:
  Type: 'AWS::Transfer::Server'
  Properties:
    WorkflowDetails:
      OnUpload:
        - ExecutionRole: workflow-execution-role-arn
          WorkflowId: !GetAtt
            - TransferWorkflow
            - WorkflowId
TransferWorkflow:
  Type: AWS::Transfer::Workflow
  Properties:
    Description: Transfer Family Workflows Blog
    Steps:
      - Type: COPY
        CopyStepDetails:
          Name: copyToUserKey
          DestinationFileLocation:
            S3FileLocation:
              Bucket: archived-records
              Key: ${transfer:UserName}/
          OverwriteExisting: 'TRUE'
      - Type: TAG
        TagStepDetails:
          Name: tagFileForArchive
          Tags:
            - Key: Archive
              Value: yes
      - Type: CUSTOM
        CustomStepDetails:
          Name: transferExtract
          Target: arn:aws:lambda:region:account-id:function:function-name
          TimeoutSeconds: 60
      - Type: DELETE
        DeleteStepDetails:
          Name: DeleteInputFile
          SourceFileLocation: '${original.file}'
    Tags:
      - Key: Name
        Value: TransferFamilyWorkflows
```

JSON

```
{
    "AWSTemplateFormatVersion": "2010-09-09",
    "Resources": {
        "SFTPServer": {
            "Type": "AWS::Transfer::Server",
            "Properties": {
                "WorkflowDetails": {
                    "OnUpload": [
                        {
                            "ExecutionRole": "workflow-execution-role-arn",
                            "WorkflowId": {
                                "Fn::GetAtt": [
                                    "TransferWorkflow",
                                    "WorkflowId"
                                ]
                            }
                        }
                    ]
                }
            }
        },
        "TransferWorkflow": {
            "Type": "AWS::Transfer::Workflow",
            "Properties": {
                "Description": "Transfer Family Workflows Blog",
                "Steps": [
                    {
                        "Type": "COPY",
                        "CopyStepDetails": {
                            "Name": "copyToUserKey",
                            "DestinationFileLocation": {
                                "S3FileLocation": {
                                    "Bucket": "archived-records",
                                    "Key": "${transfer:UserName}/"
                                }
                            },
                            "OverwriteExisting": "TRUE"
                        }
                    },
                    {
                        "Type": "TAG",
```

```
                        "TagStepDetails": {
                            "Name": "tagFileForArchive",
                            "Tags": [
                                {
                                    "Key": "Archive",
                                    "Value": "yes"
                                }
                            ]
                        }
                    },
                    {
                        "Type": "CUSTOM",
                        "CustomStepDetails": {
                            "Name": "transferExtract",
                            "Target": "arn:aws:lambda:region:account-
id:function:function-name",
                            "TimeoutSeconds": 60
                        }
                    },
                    {
                        "Type": "DELETE",
                        "DeleteStepDetails": {
                            "Name": "DeleteInputFile",
                            "SourceFileLocation": "${original.file}"
                        }
                    }
                ],
                "Tags": [
                    {
                        "Key": "Name",
                        "Value": "TransferFamilyWorkflows"
                    }
                ]
            }
        }
    }
}
```

3. Ersetzen Sie die folgenden Elemente durch Ihre tatsächlichen Werte.

   - Ersetzen Sie *workflow-execution-role-arn* mit dem ARN für eine tatsächliche Workflow-Ausführungsrolle. Beispiel: `arn:aws:transfer:us-east-2:111122223333:workflow/w-1234567890abcdef0`

- `arn:aws:lambda:region:account-id:function:function-name`Ersetzen Sie es durch den ARN für Ihre Lambda-Funktion. z. B. `arn:aws:lambda:us-east-2:123456789012:function:example-lambda-idp`.

4. Folgen Sie den Anweisungen zum Bereitstellen eines AWS CloudFormation Stacks anhand einer vorhandenen Vorlage unter Auswahl einer Stack-Vorlage im AWS CloudFormation Benutzerhandbuch.

Nachdem der Stack bereitgestellt wurde, können Sie Details dazu auf der Registerkarte Ausgaben in der CloudFormation Konsole einsehen. Die Vorlage erstellt einen neuen AWS Transfer Family SFTP-Server, der vom Service verwaltete Benutzer verwendet, sowie einen neuen Workflow und ordnet den Workflow dem neuen Server zu.

# Einen Workflow von einem Transfer Family Family-Server entfernen

Wenn Sie einem Transfer Family Family-Server einen Workflow zugeordnet haben und diese Zuordnung nun entfernen möchten, können Sie dies mithilfe der Konsole oder programmgesteuert tun.

Console

So entfernen Sie einen Workflow von einem Transfer Family Family-Server

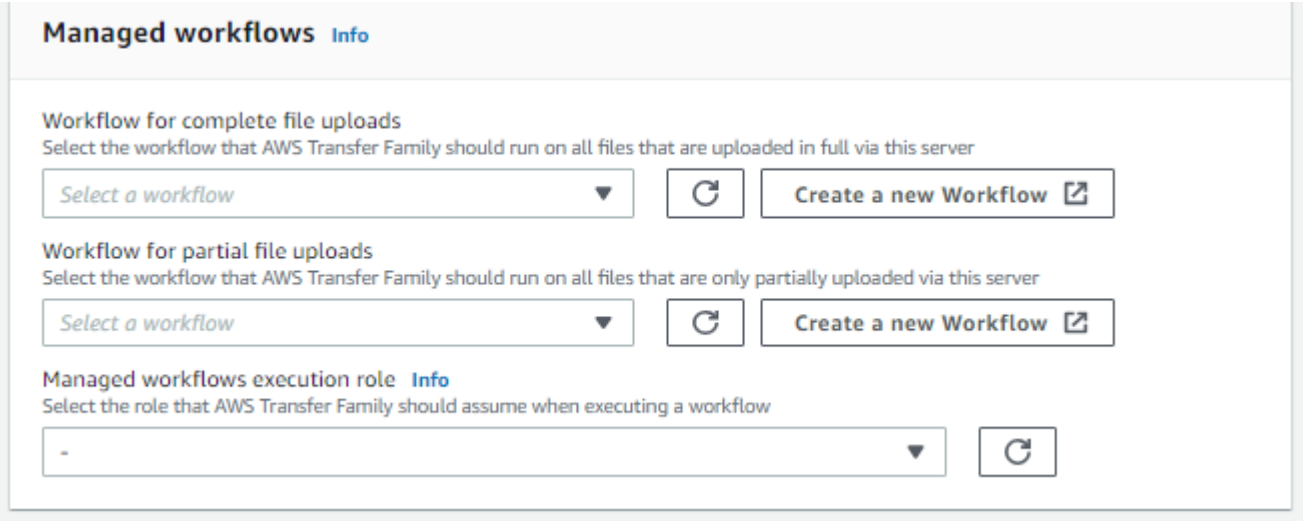
1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Wählen Sie im linken Navigationsbereich Server aus.
3. Wählen Sie den Bezeichner für den Server in der Spalte Server-ID aus.
4. Scrollen Sie auf der Detailseite für den Server nach unten zum Abschnitt Zusätzliche Details und wählen Sie dann Bearbeiten aus.
5. Löschen Sie auf der Seite Zusätzliche Details bearbeiten im Abschnitt Verwaltete Workflows die Informationen für alle Einstellungen:

   - Wählen Sie den Bindestrich (-) aus der Liste der Workflows für den Workflow für vollständige Datei-Uploads aus.
   - Falls nicht bereits deaktiviert, wählen Sie den Bindestrich (-) aus der Liste der Workflows für den Workflow für unvollständige Datei-Uploads aus.

- Wählen Sie den Bindestrich (-) aus der Rollenliste für die Ausführungsrolle Verwaltete Workflows aus.

Wenn Sie den Gedankenstrich nicht sehen, scrollen Sie nach oben, bis Sie ihn sehen, da er der erste Wert in jedem Menü ist.

Der Bildschirm sollte wie folgt aussehen.




6. Scrollen Sie nach unten und wählen Sie Speichern, um Ihre Änderungen zu speichern.

CLI

Sie verwenden den Aufruf `update-server` (oder `UpdateServer` für API) und geben leere Argumente für die `OnPartialUpload` Parameter `OnUpload` und an.

Führen Sie von der AWS CLI aus den folgenden Befehl aus:

```
aws transfer update-server --server-id your-server-id --workflow-details
 '{"OnPartialUpload":[],"OnUpload":[]}'
```

*`your-server-id`*Ersetzen Sie es durch die ID für Ihren Server. Wenn Ihre Server-ID beispielsweise`s-01234567890abcdef`, lautet der Befehl wie folgt:

```
aws transfer update-server --server-id s-01234567890abcdef --workflow-details
 '{"OnPartialUpload":[],"OnUpload":[]}'
```

# Einschränkungen und Einschränkungen verwalteter Workflows

## Einschränkungen

Die folgenden Einschränkungen gelten derzeit für Workflows zur Verarbeitung nach dem Upload für AWS Transfer Family.

- Konto- und regionsübergreifende AWS Lambda Funktionen werden nicht unterstützt. Sie können jedoch kontenübergreifend kopieren, sofern Ihre AWS Identity and Access Management (IAM-) Richtlinien korrekt konfiguriert sind.
- Für alle Workflow-Schritte müssen sich alle Amazon S3 S3-Buckets, auf die der Workflow zugreift, in derselben Region wie der Workflow selbst befinden.
- Für einen Entschlüsselungsschritt muss das Entschlüsselungsziel mit der Quelle für Region und Backing-Store übereinstimmen (wenn die zu entschlüsselnde Datei beispielsweise in Amazon S3 gespeichert ist, muss sich das angegebene Ziel auch in Amazon S3 befinden).
- Nur asynchrone benutzerdefinierte Schritte werden unterstützt.
- Die Timeouts für benutzerdefinierte Schritte sind ungefähre Angaben. Das heißt, das Timeout kann etwas länger dauern als angegeben. Darüber hinaus ist der Workflow von der Lambda-Funktion abhängig. Wenn die Funktion während der Ausführung verzögert wird, ist sich der Workflow der Verzögerung daher nicht bewusst.
- Wenn Sie Ihr Drosselungslimit überschreiten, fügt Transfer Family der Warteschlange keine Workflow-Operationen hinzu.
- Workflows werden für Dateien mit einer Größe von 0 nicht initiiert. Dateien mit einer Größe von mehr als 0 initiieren den zugehörigen Workflow.

## Einschränkungen

Darüber hinaus gelten die folgenden Funktionseinschränkungen für Workflows für Transfer Family:

- Die Anzahl der Workflows pro Region und Konto ist auf 10 begrenzt.
- Das maximale Timeout für benutzerdefinierte Schritte beträgt 30 Minuten.
- Die maximale Anzahl von Schritten in einem Workflow beträgt 8.
- Die maximale Anzahl von Tags pro Workflow beträgt 50.
- Die maximale Anzahl gleichzeitiger Ausführungen, die einen Entschlüsselungsschritt enthalten, beträgt 250 pro Workflow.

- Sie können maximal 3 private PGP-Schlüssel pro Transfer Family Family-Server pro Benutzer speichern.
- Die maximale Größe für eine entschlüsselte Datei beträgt 10 GB.
- Wir drosseln die neue Ausführungsrate mithilfe eines Token-Bucket-Systems mit einer Burst-Kapazität von 100 und einer Nachfüllrate von 1.
- Jedes Mal, wenn Sie einen Workflow von einem Server entfernen und ihn durch einen neuen ersetzen oder die Serverkonfiguration aktualisieren (was sich auf die Ausführungsrolle eines Workflows auswirkt), müssen Sie etwa 10 Minuten warten, bevor Sie den neuen Workflow ausführen. Der Transfer Family Family-Server speichert die Workflow-Details im Cache, und es dauert 10 Minuten, bis der Server seinen Cache aktualisiert hat.

  Darüber hinaus müssen Sie sich von allen aktiven SFTP-Sitzungen abmelden und sich nach Ablauf der 10-minütigen Wartezeit wieder anmelden, um die Änderungen zu sehen.

# Server verwalten

In diesem Abschnitt finden Sie Informationen zum Anzeigen einer Liste Ihrer Server, zum Anzeigen Ihrer Serverdetails, zum Bearbeiten Ihrer Serverdetails und zum Ändern des Hostschlüssels für Ihren SFTP-fähigen Server.

Themen


- Eine Liste von Servern anzeigen
- Löschen Sie einen Server
- SFTP-, FTPS- und FTP-Serverdetails anzeigen
- AS2-Serverdetails anzeigen
- Serverdetails bearbeiten
- Hostschlüssel für Ihren SFTP-fähigen Server verwalten
- Überwachung der Nutzung in der Konsole


## Eine Liste von Servern anzeigen

Auf der AWS Transfer Family Konsole finden Sie eine Liste all Ihrer Server, die sich in der von Ihnen ausgewählten AWS Region befinden.

Um eine Liste Ihrer Server zu finden, die in einer AWS Region existieren

- Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.

  Wenn Sie einen oder mehrere Server in der aktuellen AWS Region haben, öffnet sich die Konsole und zeigt eine Liste Ihrer Server an. Wenn Sie keine Serverliste sehen, stellen Sie sicher, dass Sie sich in der richtigen Region befinden. Sie können auch im Navigationsbereich Servers (Server) auswählen.

  Weitere Informationen zum Anzeigen Ihrer Serverdetails finden Sie unterSFTP-, FTPS- und FTP-Serverdetails anzeigen.

# Löschen Sie einen Server

In diesem Verfahren wird erklärt, wie Sie einen Transfer Family Family-Server mithilfe der AWS Transfer Family Konsole oder löschen AWS CLI.

> **Important**
>
> Ihnen wird jedes Protokoll, das für den Zugriff auf Ihren Endpunkt aktiviert ist, in Rechnung gestellt, bis Sie den Server löschen.

> **Warning**
>
> Das Löschen eines Servers hat zur Folge, dass alle seine Benutzer gelöscht werden. Daten in dem Bucket, auf den über den Server zugegriffen wurde, werden nicht gelöscht und sind weiterhin für AWS Benutzer zugänglich, die über Berechtigungen für diese Amazon S3 S3-Buckets verfügen.

Console

Um einen Server mithilfe der Konsole zu löschen

1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Wählen Sie im linken Navigationsbereich Server aus.
3. Aktivieren Sie das Kontrollkästchen des Servers, den Sie löschen möchten.
4. Klicken Sie bei  Actions auf Delete.
5. Geben Sie im daraufhin angezeigten Bestätigungsdialogfeld das Wort ein**`delete`**, und klicken Sie dann auf Löschen, um zu bestätigen, dass Sie den Server löschen möchten.

Der Server wird von der Seite Server gelöscht und es wird Ihnen nichts mehr in Rechnung gestellt.

AWS CLI

So löschen Sie einen Server mit der CLI

1. (Optional) Führen Sie den folgenden Befehl aus, um die Details für den Server anzuzeigen, den Sie dauerhaft löschen möchten.

   ```
   aws transfer describe-server --server-id your-server-id
   ```

   Dieser `describe-server` Befehl gibt alle Details für Ihren Server zurück.

2. Führen Sie den folgenden Befehl aus, um den Server zu löschen.

   ```
   aws transfer delete-server --server-id your-server-id
   ```

Bei Erfolg löscht der Befehl den Server und gibt keine Informationen zurück.

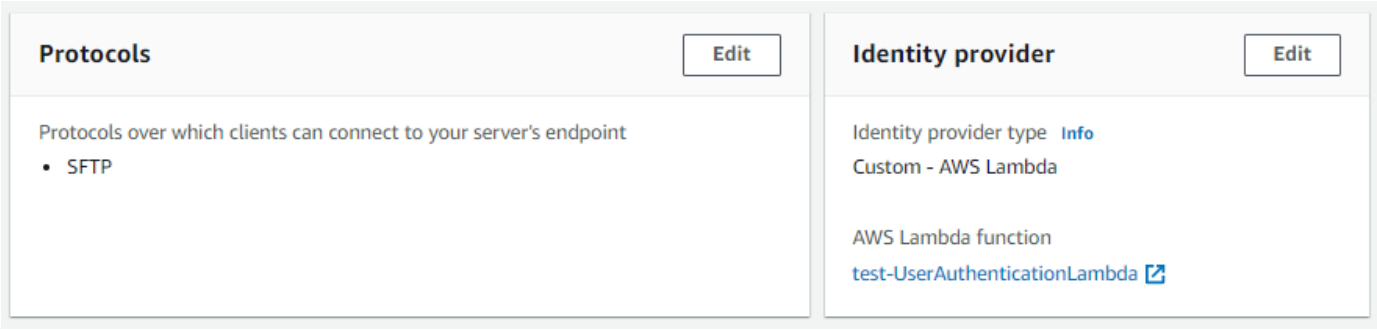
# SFTP-, FTPS- und FTP-Serverdetails anzeigen

Sie finden eine Liste mit Details und Eigenschaften für einen einzelnen AWS Transfer Family Server. Zu den Servereigenschaften gehören Protokolle, Identitätsanbieter, Status, Endpunkttyp, benutzerdefinierter Hostname, Endpunkt, Benutzer, Protokollierungsrolle, Serverhostschlüssel und Tags.

Um Serverdetails anzuzeigen

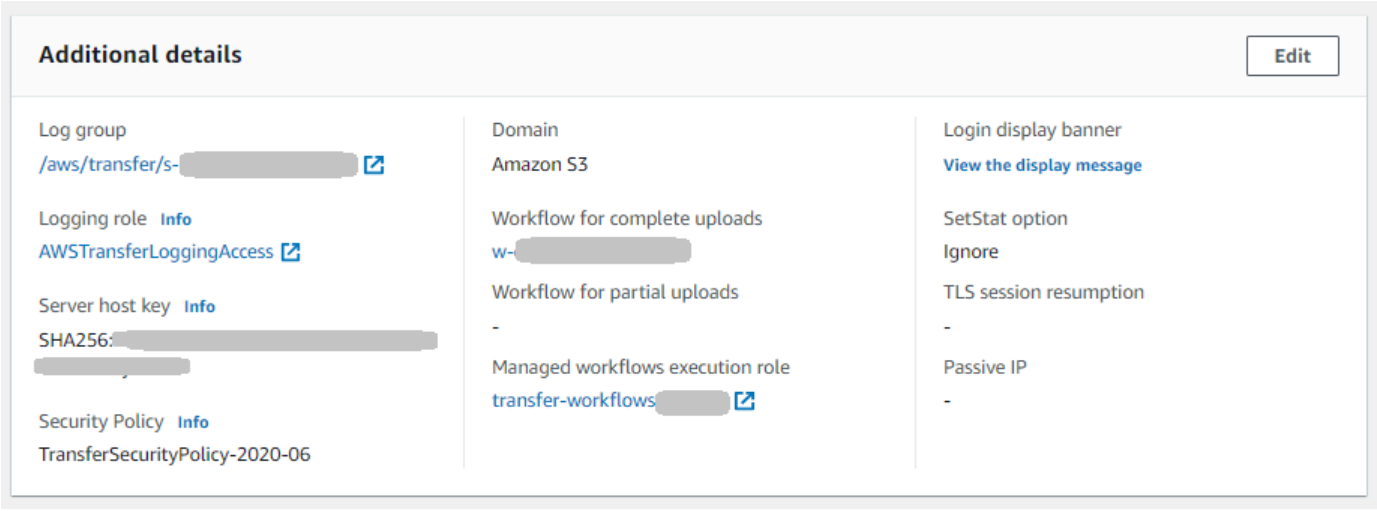
1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Klicken Sie im Navigationsbereich auf Servers (Server).
3. Wählen Sie den Bezeichner in der Spalte Server-ID aus, um die Seite mit den Serverdetails aufzurufen (siehe unten).

   Sie können die Eigenschaften des Servers auf dieser Seite ändern, indem Sie Bearbeiten wählen. Weitere Informationen zum Bearbeiten von Serverdetails finden Sie unterServerdetails bearbeiten. Die Detailseite für AS2-Server unterscheidet sich geringfügig. Informationen zu AS2-Servern finden Sie unter. AS2-Serverdetails anzeigen



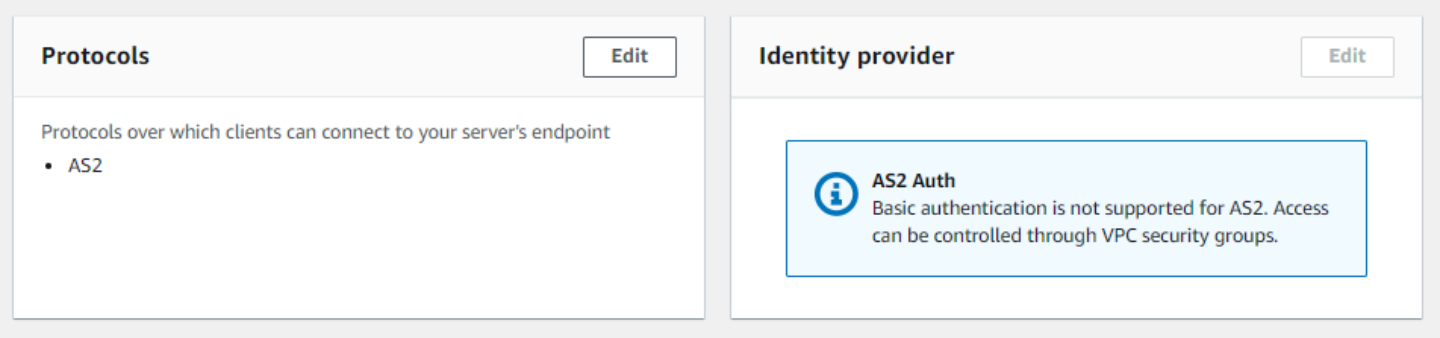

**Note**

Die Werte für die Beschreibung des Server-Host-Schlüssels und das Importdatum sind seit September 2022 neu. Diese Werte wurden eingeführt, um die Funktion für mehrere Hostschlüssel zu unterstützen. Diese Funktion erforderte die Migration aller einzelnen Hostschlüssel, die vor der Einführung mehrerer Hostschlüssel verwendet wurden.
Der Wert Importdatum für einen migrierten Server-Hostschlüssel ist auf das Datum der letzten Änderung für den Server festgelegt. Das heißt, das Datum, das Sie für Ihren migrierten Hostschlüssel sehen, entspricht dem Datum, an dem Sie den Server vor der Migration des Server-Host-Schlüssels zuletzt geändert haben.
Der einzige Schlüssel, der migriert wurde, ist Ihr ältester oder einziger Server-Hostschlüssel. Alle zusätzlichen Schlüssel haben ihr aktuelles Datum, ab dem Sie sie importiert haben. Darüber hinaus enthält der migrierte Schlüssel eine Beschreibung, anhand derer er leicht als migriert identifiziert werden kann.
Die Migration fand zwischen dem 2. und 13. September statt. Das tatsächliche Migrationsdatum innerhalb dieses Bereichs hängt von der Region Ihres Servers ab.

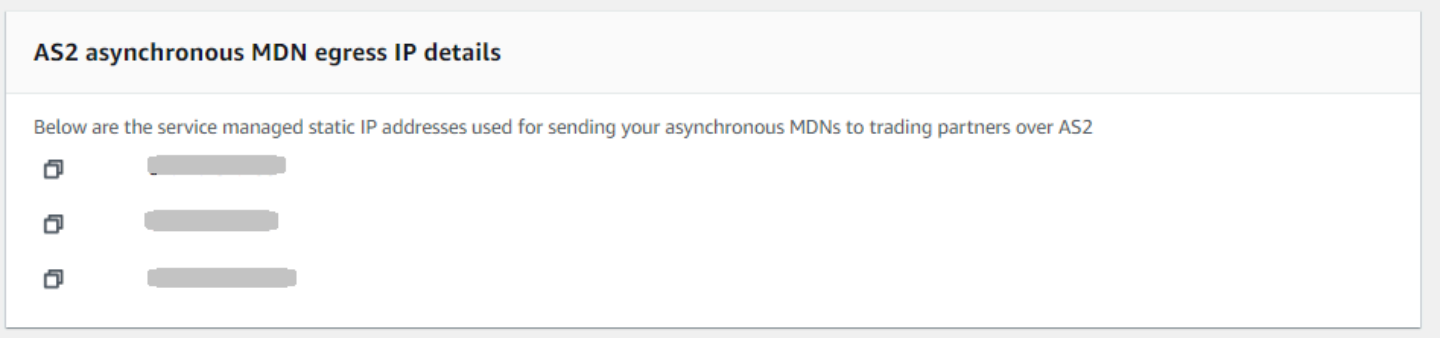



# AS2-Serverdetails anzeigen

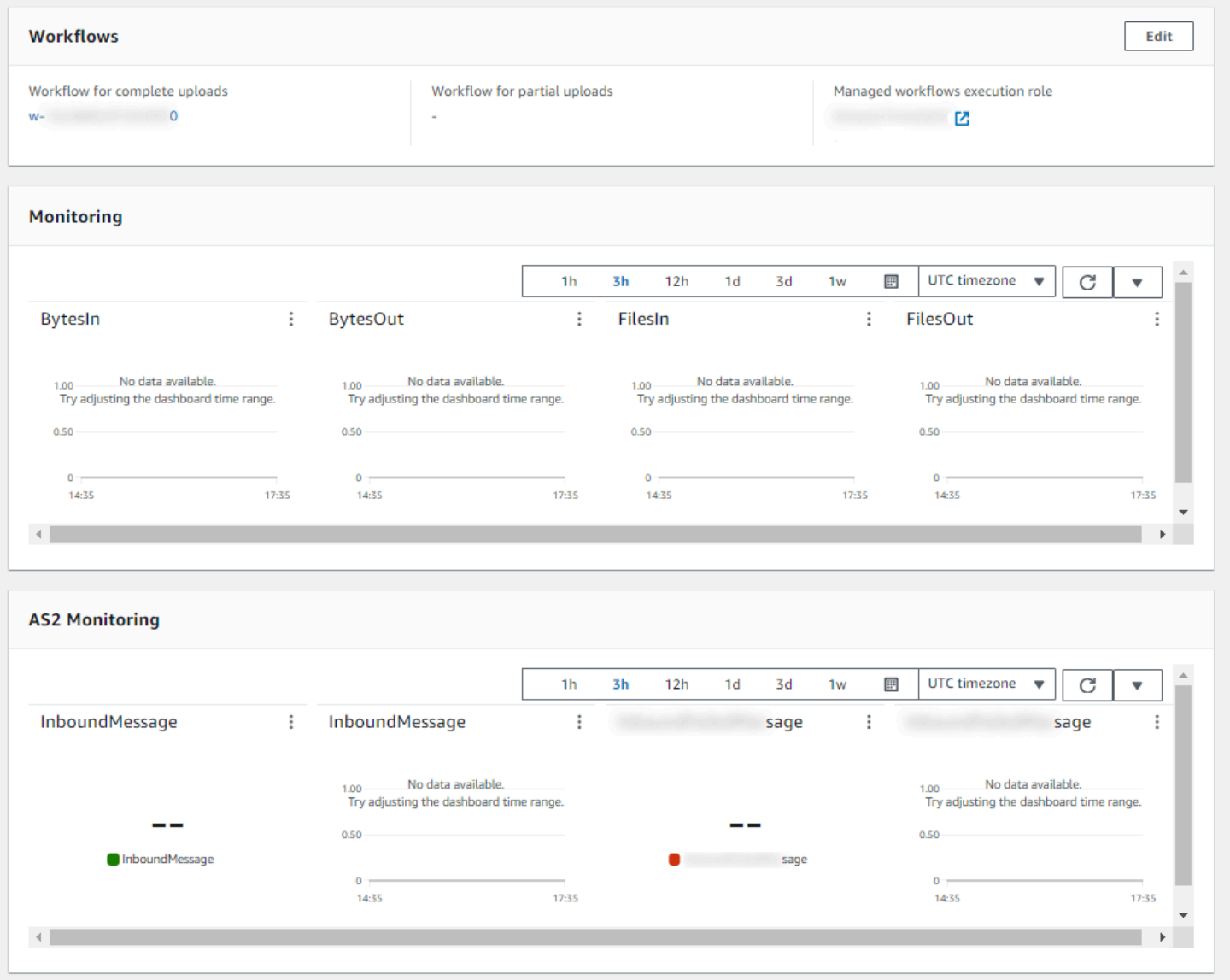
Sie finden eine Liste mit Details und Eigenschaften für einen einzelnen AWS Transfer Family Server. Zu den Servereigenschaften gehören Protokolle, Status und mehr. Bei AS2-Servern können Sie auch die asynchronen MDN-Ausgangs-IP-Adressen von AS2 anzeigen.




Jedem AS2-Server werden drei statische IP-Adressen zugewiesen. Verwenden Sie diese IP-Adressen, um asynchrone mDNS über AS2 an Ihre Handelspartner zu senden.



Der untere Teil der Seite mit den AS2-Serverdetails enthält Details zu allen angehängten Workflows sowie Überwachungs- und Tagging-Informationen.




# Serverdetails bearbeiten

Nachdem Sie einen AWS Transfer Family Server erstellt haben, können Sie die Serverkonfiguration bearbeiten.

Themen

- Bearbeiten Sie die Dateiübertragungsprotokolle
- Bearbeiten Sie die benutzerdefinierten Identity-Provider-Parameter
- Bearbeiten Sie den Serverendpunkt

- Bearbeiten Sie Ihre Logging-Konfiguration
- Bearbeiten Sie die Sicherheitsrichtlinie
- Ändern Sie den verwalteten Workflow für Ihren Server
- Ändern Sie die Display-Banner für Ihren Server
- Schalten Sie Ihren Server online oder offline

Um die Konfiguration eines Servers zu bearbeiten

1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Wählen Sie im linken Navigationsbereich Server aus.
3. Wählen Sie den Bezeichner in der Spalte Server-ID aus, um die Seite mit den Serverdetails aufzurufen (siehe unten).

   Sie können die Eigenschaften des Servers auf dieser Seite ändern, indem Sie Bearbeiten wählen:

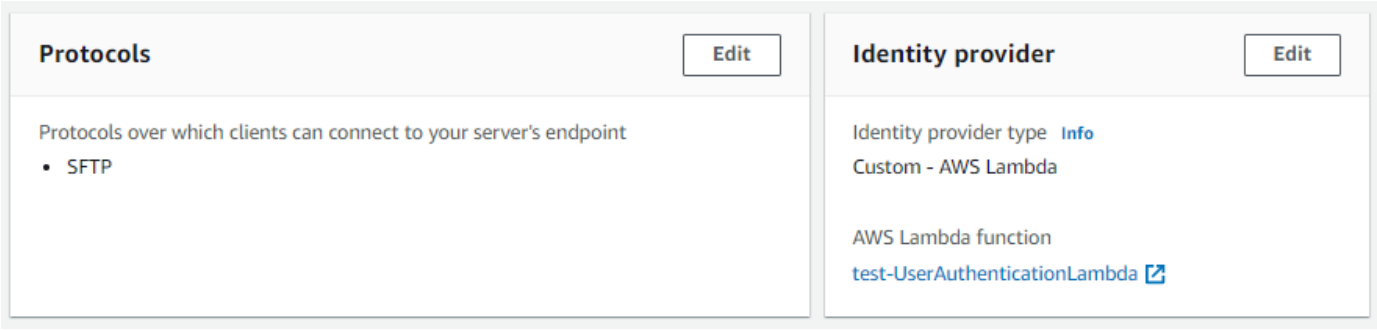
   - Informationen zum Ändern der Protokolle finden Sie unterBearbeiten Sie die Dateiübertragungsprotokolle.
   - Beachten Sie beim Identitätsanbieter, dass Sie den Identitätsanbietertyp eines Servers nicht ändern können, nachdem Sie den Server erstellt haben. Um den Identitätsanbieter zu ändern, müssen Sie den Server löschen und dann mit dem gewünschten Identitätsanbieter neu erstellen.

     > **Note**
     >
     > Wenn Ihr Server einen benutzerdefinierten Identitätsanbieter verwendet, können Sie einige Eigenschaften bearbeiten. Details hierzu finden Sie unter Bearbeiten Sie die benutzerdefinierten Identity-Provider-Parameter.

   - Informationen zum Ändern des Endpunkttyps oder des benutzerdefinierten Hostnamens finden Sie unterBearbeiten Sie den Serverendpunkt.
   - Um eine Vereinbarung hinzuzufügen, müssen Sie zunächst AS2 als Protokoll zu Ihrem Server hinzufügen. Details hierzu finden Sie unter Bearbeiten Sie die Dateiübertragungsprotokolle.
   - Informationen zur Verwaltung von Hostschlüsseln für Ihren Server finden Sie unterHostschlüssel für Ihren SFTP-fähigen Server verwalten.
   - Unter Zusätzliche Details können Sie die folgenden Informationen bearbeiten:

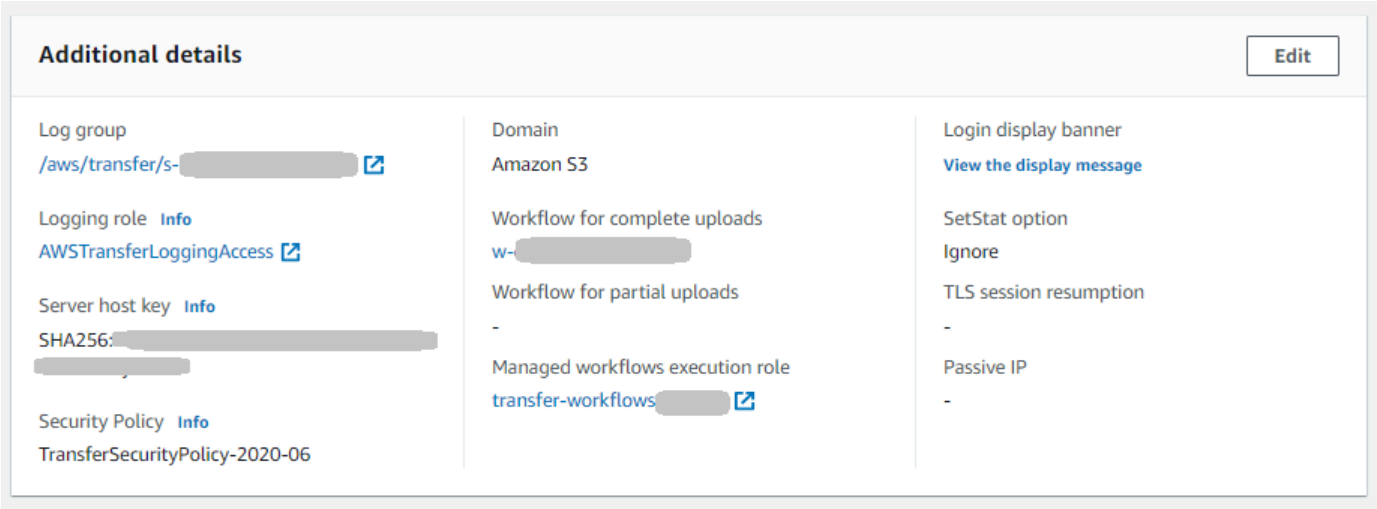
- Informationen zum Ändern der Protokollierungsrolle finden Sie unterBearbeiten Sie Ihre Logging-Konfiguration.
- Informationen zum Ändern der Sicherheitsrichtlinie finden Sie unterBearbeiten Sie die Sicherheitsrichtlinie.
- Informationen zum Ändern des Server-Host-Schlüssels finden Sie unterHostschlüssel für Ihren SFTP-fähigen Server verwalten.
- Informationen zum Ändern des verwalteten Workflows für Ihren Server finden Sie unterÄndern Sie den verwalteten Workflow für Ihren Server.
- Informationen zum Bearbeiten der Display-Banner für Ihren Server finden Sie unterÄndern Sie die Display-Banner für Ihren Server.

- Unter Zusätzliche Konfiguration können Sie die folgenden Informationen bearbeiten:
  - SetStat Option: Aktivieren Sie diese Option, um den Fehler zu ignorieren, der generiert wird, wenn ein Client versucht, eine Datei `SETSTAT` zu verwenden, die Sie in einen Amazon S3 S3-Bucket hochladen. Weitere Informationen finden Sie in der `SetStatOption` Dokumentation im ProtocolDetailsThema.
  - Wiederaufnahme der TLS-Sitzung: bietet einen Mechanismus zur Wiederaufnahme oder gemeinsamen Nutzung eines ausgehandelten geheimen Schlüssels zwischen der Kontroll- und Datenverbindung für eine FTPS-Sitzung. Weitere Informationen finden Sie in der `TlsSessionResumptionMode` Dokumentation zum Thema. ProtocolDetails
  - Passive IP: steht für den passiven Modus für FTP- und FTPS-Protokolle. Geben Sie eine einzelne IPv4-Adresse ein, z. B. die öffentliche IP-Adresse einer Firewall, eines Routers oder eines Load Balancers. Weitere Informationen finden Sie in der `PassiveIp` Dokumentation zum ProtocolDetailsThema.
- Informationen zum Starten oder Stoppen Ihres Servers finden Sie unterSchalten Sie Ihren Server online oder offline.
- Informationen zum Löschen eines Servers finden Sie unterLöschen Sie einen Server.
- Informationen zum Bearbeiten der Eigenschaften eines Benutzers finden Sie unterVerwaltung der Zugriffskontrollen.




**Note**

Die Werte für die Beschreibung des Server-Host-Schlüssels und das Importdatum sind seit September 2022 neu. Diese Werte wurden eingeführt, um die Funktion für mehrere Hostschlüssel zu unterstützen. Diese Funktion erforderte die Migration aller einzelnen Hostschlüssel, die vor der Einführung mehrerer Hostschlüssel verwendet wurden.
Der Wert Importdatum für einen migrierten Server-Hostschlüssel ist auf das Datum der letzten Änderung für den Server festgelegt. Das heißt, das Datum, das Sie für Ihren migrierten Hostschlüssel sehen, entspricht dem Datum, an dem Sie den Server vor der Migration des Server-Host-Schlüssels zuletzt geändert haben.
Der einzige Schlüssel, der migriert wurde, ist Ihr ältester oder einziger Server-Hostschlüssel. Alle zusätzlichen Schlüssel haben ihr aktuelles Datum, ab dem Sie sie importiert haben. Darüber hinaus enthält der migrierte Schlüssel eine Beschreibung, anhand derer er leicht als migriert identifiziert werden kann.
Die Migration fand zwischen dem 2. und 13. September statt. Das tatsächliche Migrationsdatum innerhalb dieses Bereichs hängt von der Region Ihres Servers ab.




# Bearbeiten Sie die Dateiübertragungsprotokolle

Auf der AWS Transfer Family Konsole können Sie das Dateiübertragungsprotokoll bearbeiten. Das Dateiübertragungsprotokoll verbindet den Client mit dem Endpunkt Ihres Servers.

Um die Protokolle zu bearbeiten

1. Wählen Sie auf der Seite mit den Serverdetails neben Protokolle die Option Bearbeiten aus.
2. Aktivieren oder deaktivieren Sie auf der Seite Protokolle das oder die Kontrollkästchen für das Protokoll, um die folgenden Dateiübertragungsprotokolle hinzuzufügen oder zu entfernen:

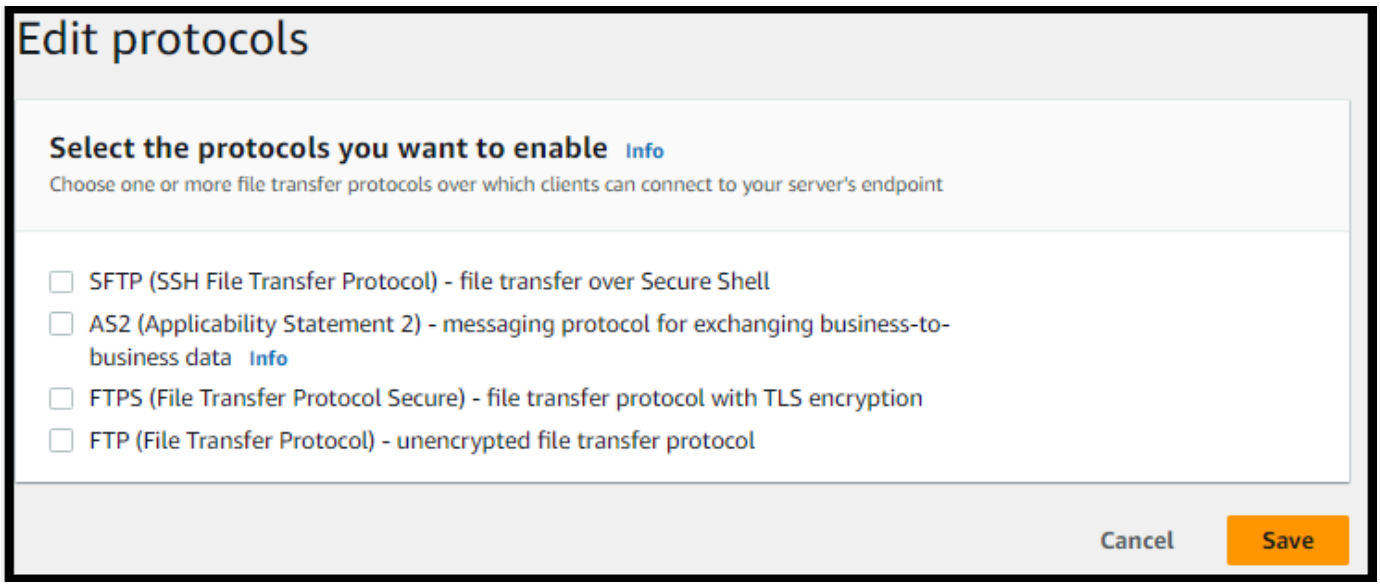
   - Secure Shell (SSH) File Transfer Protocol (SFTP) — Dateiübertragung über SSH

     Weitere Informationen zu SFTP finden Sie unter. Erstellen Sie einen SFTP-fähigen Server
   - File Transfer Protocol Secure (FTPS) — Dateiübertragung mit TLS-Verschlüsselung

     Weitere Informationen zu FTP finden Sie unterErstellen Sie einen FTPS-fähigen Server.
   - File Transfer Protocol (FTP) — unverschlüsselte Dateiübertragung

     Weitere Informationen zu FTPS finden Sie unter. Erstellen Sie einen FTP-fähigen Server

> **Note**
>
> Wenn Sie einen vorhandenen Server haben, der nur für SFTP aktiviert ist, und Sie FTPS und FTP hinzufügen möchten, müssen Sie sicherstellen, dass Sie über die richtigen Einstellungen für Identitätsanbieter und Endpunkttyp verfügen, die mit FTPS und FTP kompatibel sind.




Wenn Sie FTPS auswählen, müssen Sie ein in AWS Certificate Manager (ACM) gespeichertes Zertifikat auswählen, das zur Identifizierung Ihres Servers verwendet wird, wenn Clients über FTPS eine Verbindung zu ihm herstellen.

Informationen zum Anfordern eines neuen öffentlichen Zertifikats finden Sie unter Anfordern eines öffentlichen Zertifikats im AWS Certificate Manager Benutzerhandbuch.

Informationen zum Importieren eines vorhandenen Zertifikats in ACM finden Sie unter Zertifikate in ACM importieren im AWS Certificate Manager Benutzerhandbuch.

Informationen zum Anfordern eines privaten Zertifikats für die Verwendung von FTPS über private IP-Adressen finden Sie unter Anfordern eines privaten Zertifikats im AWS Certificate Manager Benutzerhandbuch.

Zertifikate mit den folgenden kryptografischen Algorithmen und Schlüsselgrößen werden unterstützt:

- 2048-Bit-RSA (RSA_2048)
- 4096-Bit-RSA (RSA_4096)
- Elliptic Prime Curve 256-Bit (EC_prime256v1)
- Elliptic Prime Curve 384-Bit (EC_secp384r1)
- Elliptic Prime Curve 521-Bit (EC_secp521r1)

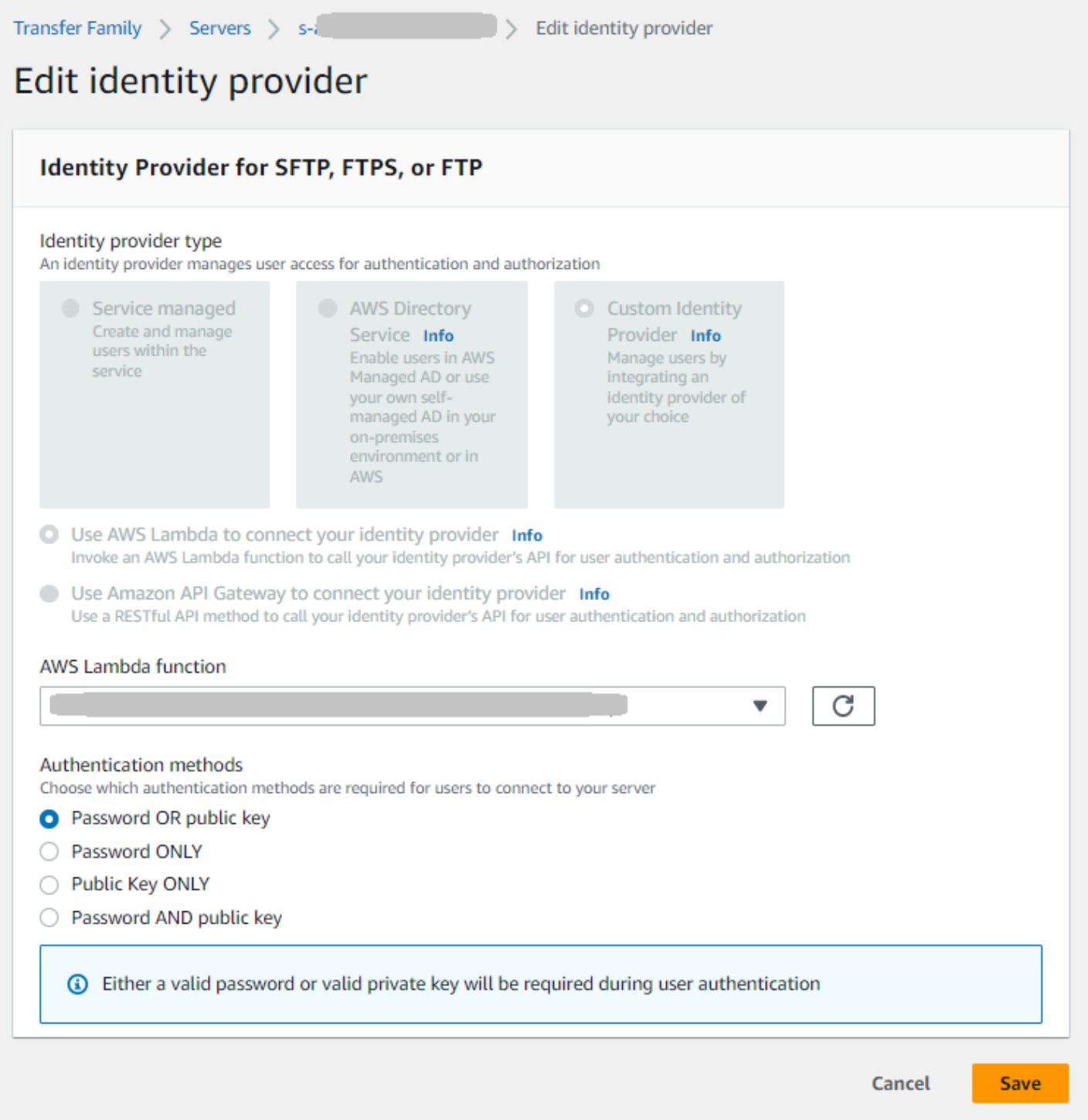
> **Note**
>
> Das Zertifikat muss ein gültiges SSL/TLS X.509 Version 3-Zertifikat mit angegebenem FQDN oder IP-Adresse sein und Informationen über den Aussteller enthalten.

3. Wählen Sie Speichern. Sie kehren zur Seite mit den Serverdetails zurück.

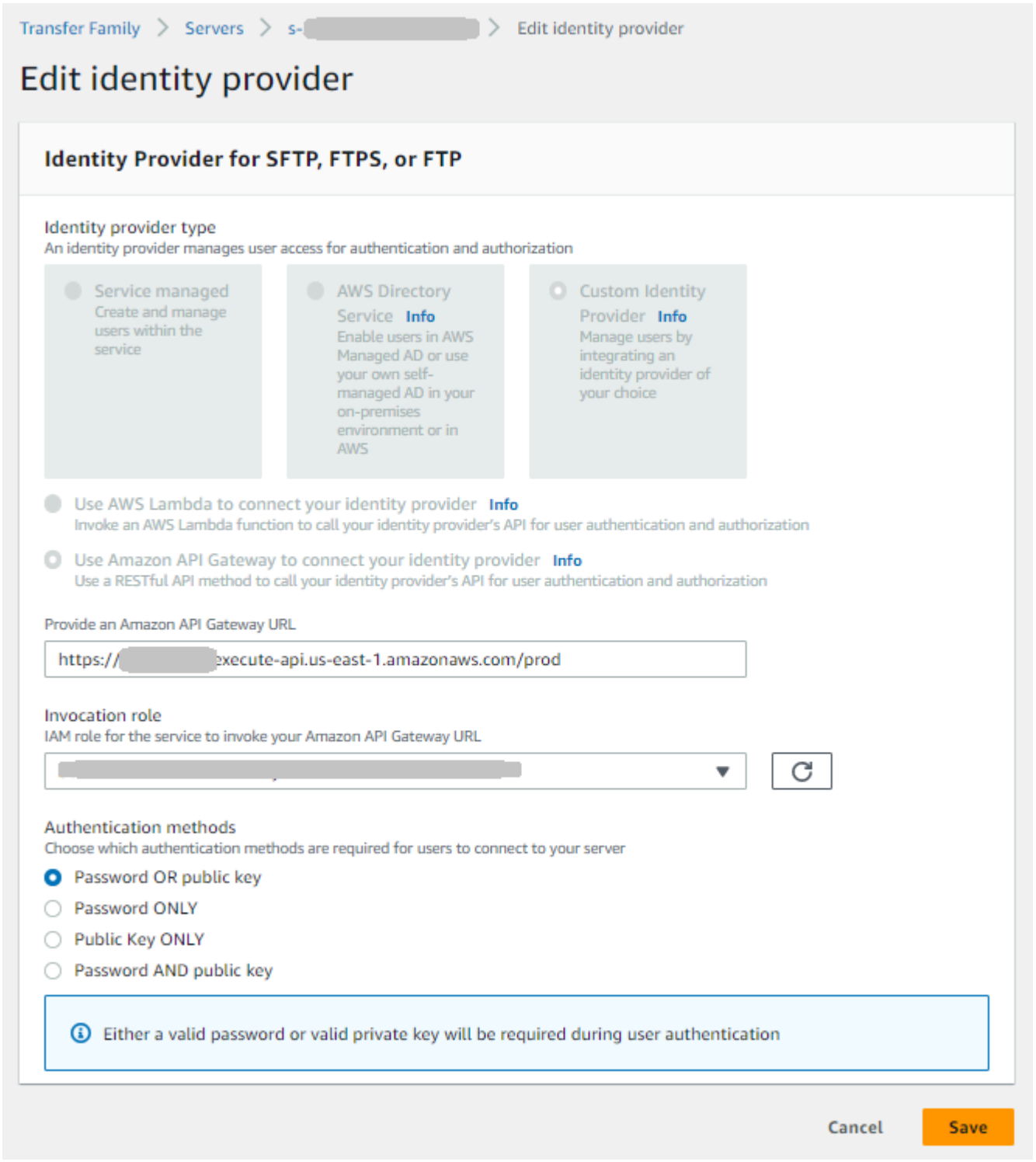
## Bearbeiten Sie die benutzerdefinierten Identity-Provider-Parameter

In der AWS Transfer Family Konsole können Sie für benutzerdefinierte Identitätsanbieter einige Einstellungen ändern, je nachdem, ob Sie eine Lambda-Funktion oder ein API Gateway verwenden. In beiden Fällen können Sie Ihre Authentifizierungsmethode bearbeiten, wenn Ihr Server das SFTP-Protokoll verwendet.

- Wenn Sie ein Lambda als Identitätsanbieter verwenden, können Sie die zugrunde liegende Lambda-Funktion ändern.




- Wenn Sie ein API Gateway als Identitätsanbieter verwenden, können Sie die Gateway-URL oder die Aufrufrolle oder beides aktualisieren.


Transfer Family > Servers > s- > Edit identity provider
Edit identity provider
Identity Provider for SFTP, FTPS, or FTP
Identity provider type
An identity provider manages user access for authentication and authorization
Service managed
Create and manage users within the service
AWS Directory Service Info
Enable users in AWS Managed AD or use your own self-managed AD in your on-premises environment or in AWS
Custom Identity Provider Info
Manage users by integrating an identity provider of your choice
Use AWS Lambda to connect your identity provider Info
Invoke an AWS Lambda function to call your identity provider's API for user authentication and authorization
Use Amazon API Gateway to connect your identity provider Info
Use a RESTful API method to call your identity provider's API for user authentication and authorization
Provide an Amazon API Gateway URL
https:// execute-api.us-east-1.amazonaws.com/prod
Invocation role
IAM role for the service to invoke your Amazon API Gateway URL
Authentication methods
Choose which authentication methods are required for users to connect to your server
Password OR public key
Password ONLY
Public Key ONLY
Password AND public key
Either a valid password or valid private key will be required during user authentication
Cancel
Save

# Bearbeiten Sie den Serverendpunkt

Auf der AWS Transfer Family Konsole können Sie den Serverendpunkttyp und den benutzerdefinierten Hostnamen ändern. Darüber hinaus können Sie für VPC-Endpoints die Informationen zur Verfügbarkeitszone bearbeiten.

Um die Details des Serverendpunkts zu bearbeiten

1. Wählen Sie auf der Seite mit den Serverdetails neben Endpunktdetails die Option Bearbeiten aus.
2. Bevor Sie den Endpunkttyp bearbeiten können, müssen Sie zuerst den Server beenden. Anschließend können Sie auf der Seite Endpunktkonfiguration bearbeiten für Endpunkttyp einen der folgenden Werte wählen:

   - Öffentlich — Mit dieser Option ist Ihr Server über das Internet zugänglich.
   - VPC — Diese Option macht Ihren Server in Ihrer Virtual Private Cloud (VPC) zugänglich. Hinweise zu VPC finden Sie unterErstellen Sie einen Server in einer virtuellen privaten Cloud.
3. Wählen Sie für Benutzerdefinierter Hostname eine der folgenden Optionen:

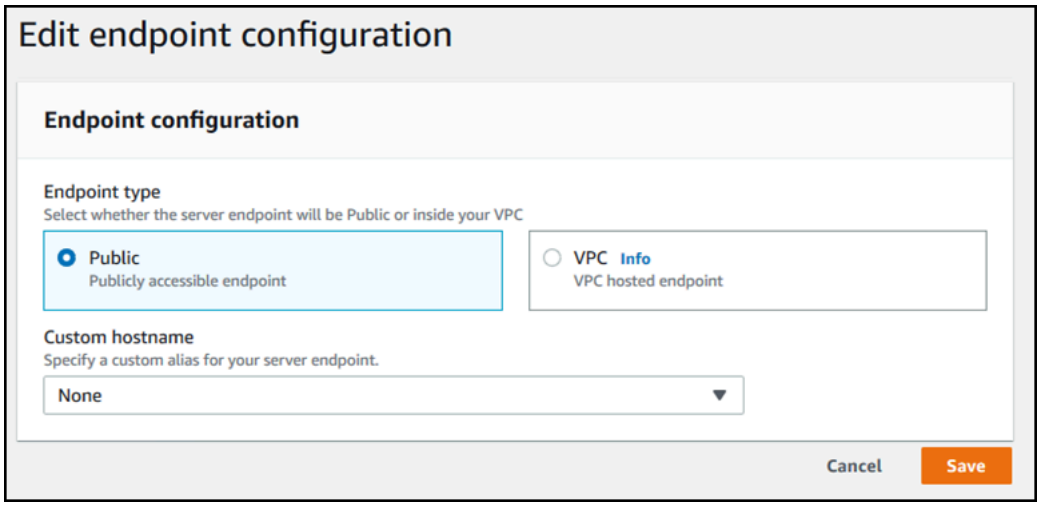
   - Keine — Wenn Sie keine benutzerdefinierte Domain verwenden möchten, wählen Sie Keine.

     Sie erhalten einen Server-Hostnamen, der von bereitgestellt wird AWS Transfer Family. Der Server-Host-Name hat das Format *`serverId`*`.server.transfer.`*`regionId`*`.amazonaws.com`.
   - Amazon Route 53 DNS-Alias — Um einen DNS-Alias zu verwenden, der in Route 53 automatisch für Sie erstellt wurde, wählen Sie diese Option.
   - Anderes DNS — Um einen Hostnamen, den Sie bereits besitzen, in einem externen DNS-Dienst zu verwenden, wählen Sie Anderes DNS.

   Wenn Sie Amazon Route 53 DNS-Alias oder Anderes DNS wählen, geben Sie die Methode zur Namensauflösung an, die dem Endpunkt Ihres Servers zugeordnet werden soll.

   Beispiel: Die benutzerdefinierte Domäne heißt `sftp.inbox.example.com`. Ein benutzerdefinierter Host-Name verwendet einen von Ihnen bereitgestellten DNS-Namen, der von einem DNS-Service aufgelöst werden kann. Sie können Route 53 als Ihren DNS-Resolver verwenden oder Ihren eigenen DNS-Dienstanbieter verwenden. Informationen darüber, wie

Route 53 AWS Transfer Family verwendet wird, um Traffic von Ihrer benutzerdefinierten Domain zum Serverendpunkt weiterzuleiten, finden Sie unterMit benutzerdefinierten Hostnamen arbeiten.




4. Für VPC-Endpoints können Sie die Informationen im Bereich Availability Zones ändern.
5. Wählen Sie Speichern. Sie kehren zur Seite mit den Serverdetails zurück.

## Bearbeiten Sie Ihre Logging-Konfiguration

Auf der AWS Transfer Family Konsole können Sie Ihre Protokollierungskonfiguration ändern.

> **Note**
>
> Wenn Transfer Family beim Erstellen eines Servers eine CloudWatch Protokollierungs-IAM-Rolle für Sie erstellt hat, wird die IAM-Rolle aufgerufen. `AWSTransferLoggingAccess` Sie können es für alle Ihre Transfer Family Family-Server verwenden.

Um Ihre Logging-Konfiguration zu bearbeiten

1. Wählen Sie auf der Seite mit den Serverdetails neben Zusätzliche Details die Option Bearbeiten aus.
2. Wählen Sie je nach Konfiguration zwischen einer Logging-Rolle, einer strukturierten JSON-Protokollierung oder beidem. Weitere Informationen finden Sie unter Die Protokollierung für einen Server aktualisieren.

# Bearbeiten Sie die Sicherheitsrichtlinie

In diesem Verfahren wird erklärt, wie Sie die Sicherheitsrichtlinie eines Transfer Family Family-Servers mithilfe der AWS Transfer Family Konsole oder ändern AWS CLI.

> **Note**
>
> Wenn Ihr Endpunkt FIPS-fähig ist, können Sie die FIPS-Sicherheitsrichtlinie nicht in eine Nicht-FIPS-Sicherheitsrichtlinie ändern.

## Console

Um die Sicherheitsrichtlinie mithilfe der Konsole zu bearbeiten

1. Wählen Sie auf der Seite mit den Serverdetails neben Zusätzliche Details die Option Bearbeiten aus.
2. Wählen Sie im Abschnitt Optionen für kryptografische Algorithmen eine Sicherheitsrichtlinie aus, die die kryptografischen Algorithmen enthält, die für die Verwendung durch Ihren Server aktiviert sind.

   Weitere Informationen zu Sicherheitsrichtlinien finden Sie unter Sicherheitsrichtlinien für AWS Transfer Family Server.
3. Wählen Sie Speichern.

   Sie kehren zur Seite mit den Serverdetails zurück, auf der Sie die aktualisierte Sicherheitsrichtlinie sehen können.

## AWS CLI

So bearbeiten Sie die Sicherheitsrichtlinie mit der CLI

1. Führen Sie den folgenden Befehl aus, um die aktuelle Sicherheitsrichtlinie anzuzeigen, die an Ihren Server angehängt ist.

   ```
   aws transfer describe-server --server-id your-server-id
   ```

Dieser `describe-server` Befehl gibt alle Details für Ihren Server zurück, einschließlich der folgenden Zeile:

```
"SecurityPolicyName": "TransferSecurityPolicy-2018-11"
```

In diesem Fall lautet die Sicherheitsrichtlinie für den Server`TransferSecurityPolicy-2018-11`.

2. Stellen Sie sicher, dass Sie den genauen Namen der Sicherheitsrichtlinie für den Befehl angeben. Führen Sie beispielsweise den folgenden Befehl aus, um den Server auf zu aktualisieren`TransferSecurityPolicy-2023-05`.

```
aws transfer update-server --server-id your-server-id --security-policy-name
 "TransferSecurityPolicy-2023-05"
```

> **Note**
>
> Die Namen der verfügbaren Sicherheitsrichtlinien sind unter aufgeführtSicherheitsrichtlinien für AWS Transfer Family Server.

Bei Erfolg gibt der Befehl den folgenden Code zurück und aktualisiert die Sicherheitsrichtlinie Ihres Servers.

```
{
    "ServerId": "your-server-id"
}
```

## Ändern Sie den verwalteten Workflow für Ihren Server

Auf der AWS Transfer Family Konsole können Sie den verwalteten Workflow ändern, der dem Server zugeordnet ist.

Um den verwalteten Workflow zu ändern

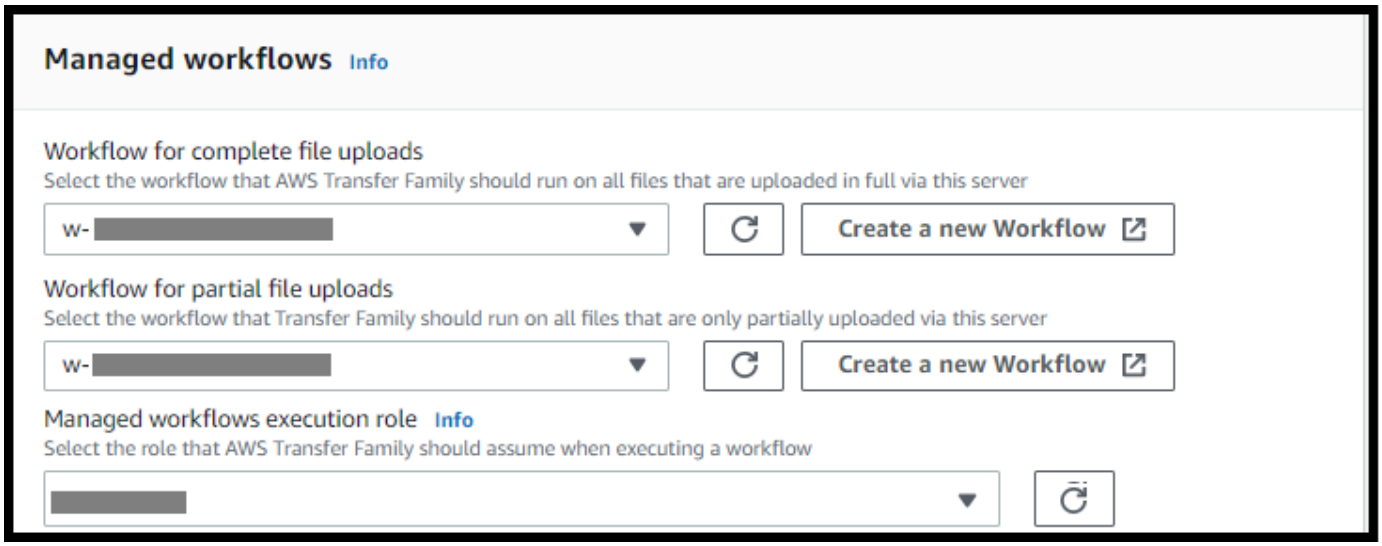
1. Wählen Sie auf der Seite mit den Serverdetails neben Zusätzliche Details die Option Bearbeiten aus.

2. Wählen Sie auf der Seite Zusätzliche Details bearbeiten im Abschnitt Verwaltete Workflows einen Workflow aus, der für alle Uploads ausgeführt werden soll.

> **Note**
>
> Wenn Sie noch keinen Workflow haben, wählen Sie Neuen Workflow erstellen aus, um einen zu erstellen.

   a. Wählen Sie die zu verwendende Workflow-ID aus.

   b. Wählen Sie eine Ausführungsrolle aus. Diese Rolle nimmt Transfer Family bei der Ausführung der Workflow-Schritte ein. Weitere Informationen finden Sie unter IAM-Richtlinien für Workflows. Wählen Sie Save (Speichern) aus.




3. Wählen Sie Speichern. Sie kehren zur Seite mit den Serverdetails zurück.

## Ändern Sie die Display-Banner für Ihren Server

Auf der AWS Transfer Family Konsole können Sie die dem Server zugewiesenen Display-Banner ändern.

Um die Display-Banner zu ändern

1. Wählen Sie auf der Seite mit den Serverdetails neben Zusätzliche Details die Option Bearbeiten aus.

2. Geben Sie auf der Seite Zusätzliche Details bearbeiten im Abschnitt Display-Banner den Text für die verfügbaren Display-Banner ein.
3. Wählen Sie Speichern. Sie kehren zur Seite mit den Serverdetails zurück.

## Schalten Sie Ihren Server online oder offline

Auf der AWS Transfer Family Konsole können Sie Ihren Server online oder offline schalten.

Um Ihren Server online zu schalten

1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Klicken Sie im Navigationsbereich auf Servers (Server).
3. Aktivieren Sie das Kontrollkästchen des Servers, der offline ist.
4. Wählen Sie für Actions (Aktionen) die Option Start.

Es kann einige Minuten dauern, bis ein Server von offline auf online umgestellt wird.

> **Note**
>
> Wenn Sie einen Server anhalten, um ihn offline zu schalten, fallen derzeit immer noch Servicegebühren für diesen Server an. Um zusätzliche serverbasierte Gebühren zu vermeiden, löschen Sie diesen Server.

Um Ihren Server offline zu schalten

1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Klicken Sie im Navigationsbereich auf Servers (Server).
3. Aktivieren Sie das Kontrollkästchen des Servers, der online ist.
4. Wählen Sie für Actions (Aktionen) die Option Stop (Stopp).

Während ein Server gestartet oder heruntergefahren wird, sind Server nicht für Dateioperationen verfügbar. Die Konsole zeigt den Start- oder Stopp-Status nicht an.

Wenn Sie den Fehler finden `START_FAILED` oder`STOP_FAILED`, wenden Sie sich an uns, AWS Support um Ihre Probleme zu lösen.

# Hostschlüssel für Ihren SFTP-fähigen Server verwalten

> **Important**
>
> Wenn Sie nicht planen, bestehende Benutzer von einem vorhandenen SFTP-fähigen Server auf einen neuen SFTP-fähigen Server zu migrieren, ignorieren Sie diesen Abschnitt. Das versehentliche Ändern des Host-Schlüssels eines Servers kann zu Unterbrechungen führen. Je nachdem, wie Ihr SFTP-Client konfiguriert ist, kann er sofort fehlschlagen, mit der Meldung, dass kein vertrauenswürdiger Hostschlüssel vorhanden ist, oder es werden drohende Eingabeaufforderungen angezeigt. Wenn es Skripts für die Automatisierung von Verbindungen gibt, würden diese höchstwahrscheinlich ebenfalls fehlschlagen.

AWS Transfer Family Stellt standardmäßig einen Hostschlüssel für Ihren SFTP-fähigen Server bereit. Sie können den Host-Standardschlüssel durch den Host-Schlüssel eines anderen Servers ersetzen. Tun Sie dies nur, wenn Sie beabsichtigen, bestehende Benutzer von einem vorhandenen SFTP-fähigen Server auf Ihren neuen SFTP-fähigen Server zu verschieben.

Um zu verhindern, dass Ihre Benutzer erneut aufgefordert werden, die Authentizität Ihres SFTP-fähigen Servers zu überprüfen, importieren Sie den Hostschlüssel für Ihren lokalen Server auf den SFTP-fähigen Server. Auf diese Weise wird auch verhindert, dass Ihre Benutzer vor einem möglichen Angriff gewarnt werden. man-in-the-middle

Als zusätzliche Sicherheitsmaßnahme können Sie die Hostschlüssel auch regelmäßig wechseln.

> **Note**
>
> In der Transfer Family Family-Konsole können Sie zwar Serverhostschlüssel für alle Server angeben und hinzufügen, diese Schlüssel sind jedoch nur für Server nützlich, die das SFTP-Protokoll verwenden.

Themen


- Fügen Sie einen zusätzlichen Server-Host-Schlüssel hinzu
- Löscht einen Server-Hostschlüssel
- Rotiert die Server-Hostschlüssel

- Zusätzliche Schlüsselinformationen zum Server-Host

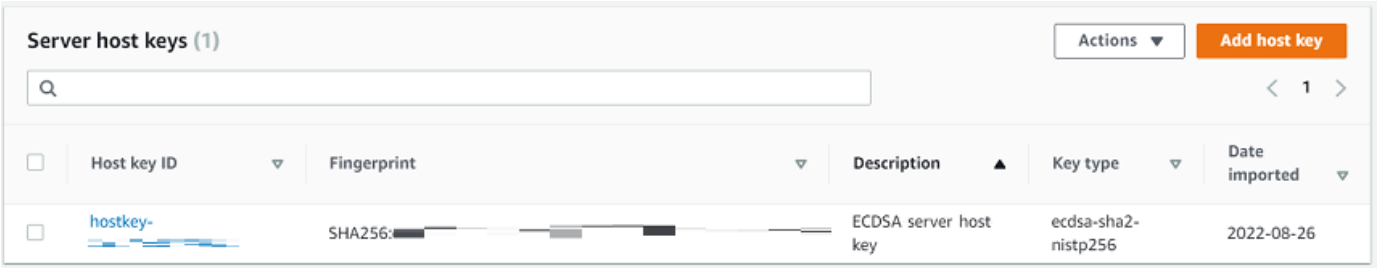
## Fügen Sie einen zusätzlichen Server-Host-Schlüssel hinzu

Auf der AWS Transfer Family Konsole können Sie zusätzliche Server-Hostschlüssel hinzufügen. Das Hinzufügen zusätzlicher Hostschlüssel in unterschiedlichen Formaten kann nützlich sein, um einen Server zu identifizieren, wenn Clients eine Verbindung zu ihm herstellen, und um Ihr Sicherheitsprofil zu verbessern. Wenn Ihr Originalschlüssel beispielsweise ein RSA-Schlüssel ist, können Sie einen zusätzlichen ECDSA-Schlüssel hinzufügen.

> **Note**
>
> Der SFTP-Client stellt eine Verbindung mit dem ersten öffentlichen Schlüssel her, über den er verfügt und der einem der aktiven Serverschlüssel entsprechen kann.

Um einen zusätzlichen Server-Host-Schlüssel hinzuzufügen

1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Wählen Sie im linken Navigationsbereich Server und dann einen Server aus, der das SFTP-Protokoll verwendet.
3. Scrollen Sie auf der Seite mit den Serverdetails nach unten zum Abschnitt Server-Hostschlüssel.




4. Wählen Sie Hostschlüssel hinzufügen.

   Die Seite Server-Host-Schlüssel hinzufügen wird angezeigt.
5. Geben Sie im Abschnitt Server-Hostschlüssel einen privaten RSA-, ECDSA- oder ED25519-Schlüssel ein, der zur Identifizierung Ihres Servers verwendet wird, wenn Clients über den SFTP-fähigen Server eine Verbindung zu ihm herstellen.

> **Note**
>
> Achten Sie beim Erstellen eines Server-Hostschlüssels darauf, dass Sie Folgendes angeben (keine Passphrase). `-N ""` Einzelheiten SSH-Schlüssel auf macOS, Linux oder Unix erstellen zum Generieren von Schlüsselpaaren finden Sie unter.

6. (Optional) Fügen Sie eine Beschreibung hinzu, um zwischen mehreren Server-Hostschlüsseln zu unterscheiden. Sie können Ihrem Schlüssel auch Tags hinzufügen.
7. Wählen Sie Schlüssel hinzufügen. Sie kehren zur Seite mit den Serverdetails zurück.

Um einen Hostschlüssel mithilfe von AWS Command Line Interface (AWS CLI) hinzuzufügen, verwenden Sie die the section called “ImportHostKey” API-Operation und geben Sie den neuen Hostschlüssel an. Wenn Sie einen neuen SFTP-fähigen Server erstellen, geben Sie Ihren Hostschlüssel als Parameter in der the section called “CreateServer” API-Operation an. Sie können den auch verwenden AWS CLI , um die Beschreibung für einen vorhandenen Hostschlüssel zu aktualisieren.

Der folgende `import-host-key` AWS CLI Beispielbefehl importiert einen Hostschlüssel für den angegebenen SFTP-fähigen Server.

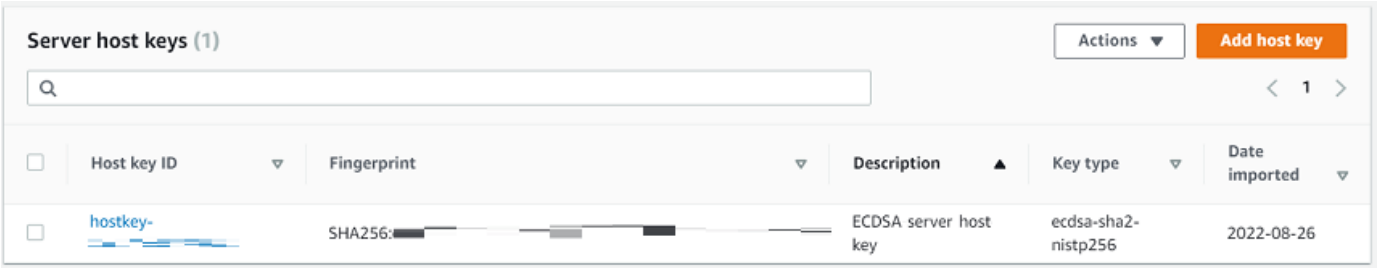
```
aws transfer import-host-key --description key-description --server-id your-server-id
 --host-key-body file://my-host-key
```

## Löscht einen Server-Hostschlüssel

Auf der AWS Transfer Family Konsole können Sie einen Server-Hostschlüssel löschen.

Um einen Server-Hostschlüssel zu löschen

1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Wählen Sie im linken Navigationsbereich Server und dann einen Server aus, der das SFTP-Protokoll verwendet.
3. Scrollen Sie auf der Seite mit den Serverdetails nach unten zum Abschnitt Server-Hostschlüssel.




4. Wählen Sie im Abschnitt Server-Hostschlüssel einen Schlüssel aus, und klicken Sie dann unter Aktionen auf Löschen.
5. Geben Sie im daraufhin angezeigten Bestätigungsdialogfeld das Wort ein**`delete`**, und klicken Sie dann auf Löschen, um zu bestätigen, dass Sie den Hostschlüssel löschen möchten.

Der Hostschlüssel wird von der Serverseite gelöscht.

Um den Hostschlüssel mithilfe von zu löschen AWS CLI, verwenden Sie den the section called "DeleteHostKey" API-Vorgang und geben Sie die Server-ID und die Hostschlüssel-ID ein.

Der folgende `delete-host-key` AWS CLI Beispielbefehl löscht einen Hostschlüssel für den angegebenen SFTP-fähigen Server.

```
aws transfer delete-host-key --server-id your-server-id --host-key-id your-host-key-id
```

## Rotiert die Server-Hostschlüssel

In regelmäßigen Abständen können Sie Ihren Server-Host-Schlüssel rotieren.

### Wie der Client einen Server-Host-Schlüssel auswählt

Die Art und Weise, wie Transfer Family den anzuwendenden Serverschlüssel auswählt, hängt von den Bedingungen für den SFTP-Client ab, wie hier erklärt. Es wird davon ausgegangen, dass es einen älteren und einen neueren Schlüssel gibt.

- Ein SFTP-Client hat zuvor keinen öffentlichen Hostschlüssel für den Server. Wenn der Client zum ersten Mal eine Verbindung zum Server herstellt, tritt einer der folgenden Fälle ein:
  - Der Client schlägt die Verbindung fehl, wenn er dafür konfiguriert ist.
  - Oder der Client wählt den ersten Schlüssel aus, der den möglichen verfügbaren Algorithmen entspricht, und fragt den Benutzer, ob dieser Schlüssel vertrauenswürdig ist. Wenn ja, aktualisiert der Client die `known_hosts` Datei (oder die lokale Konfigurationsdatei oder

Ressource, die der Client zum Aufzeichnen von Vertrauensentscheidungen verwendet) automatisch und gibt diesen Schlüssel ein.

- Ein SFTP-Client hat einen älteren Schlüssel in seiner `known_hosts` Datei. Der Client bevorzugt es, diesen Schlüssel zu verwenden, auch wenn ein neuerer Schlüssel existiert, entweder für den Algorithmus dieses Schlüssels oder für einen anderen Algorithmus. Das liegt daran, dass der Client dem Schlüssel, der sich in seiner `known_hosts` Datei befindet, ein höheres Maß an Vertrauen entgegenbringt.
- Ein SFTP-Client hat den neuen Schlüssel (in einem der verfügbaren Algorithmen) in seiner `known_hosts` Schlüsseldatei. Der Client ignoriert ältere Schlüssel, weil sie nicht vertrauenswürdig sind, und verwendet den neuen Schlüssel.
- Ein SFTP-Client hat beide Schlüssel in seiner `known_hosts` Datei. Der Client wählt den ersten Schlüssel anhand des Index aus, der mit der vom Server angebotenen Liste der verfügbaren Schlüssel übereinstimmt.

Transfer Family bevorzugt es, dass der SFTP-Client alle Schlüssel in seiner `known_hosts` Datei hat, da dies die größte Flexibilität bei der Verbindung zu einem Transfer Family Family-Server ermöglicht. Die Schlüsselrotation basiert auf der Tatsache, dass mehrere Einträge in der `known_hosts` Datei für denselben Transfer Family Family-Server vorhanden sein können.

## Rotieren Sie das Verfahren für den Server-Host-Schlüssel

Nehmen wir als Beispiel an, dass Sie Ihrem Transfer Family Family-Server den folgenden Satz von Server-Hostschlüsseln hinzugefügt haben.

Server-Host-Schlüssel

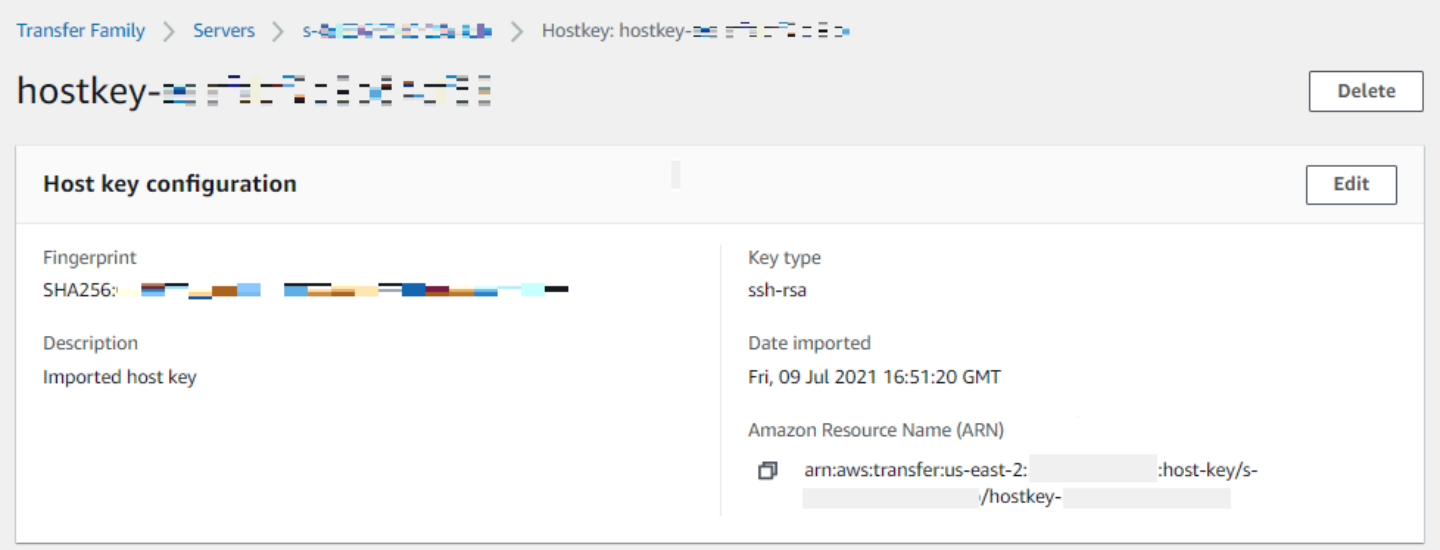
| Typ des Host-Schlüssels | Datum, das dem Server hinzugefügt wurde |
|---|---|
| RSA | 01. April 2020 |
| ECDSA | 1. Februar 2020 |
| ED25519 | 1. Dezember 2019 |
| RSA | 1. Oktober 2019 |
| ECDSA | 1. Juni 2019 |
| ED25519 | 1. März 2019 |

Um den Server-Host-Schlüssel zu rotieren

1. Fügen Sie einen neuen Server-Hostschlüssel hinzu. Dieses Verfahren wird unter beschriebenFügen Sie einen zusätzlichen Server-Host-Schlüssel hinzu.
2. Löschen Sie einen oder mehrere Hostschlüssel desselben Typs, die Sie zuvor hinzugefügt haben. Dieses Verfahren wird unter beschriebenLöscht einen Server-Hostschlüssel.
3. Alle Tasten sind sichtbar und können je nach dem zuvor unter beschriebenen Verhalten aktiv seinWie der Client einen Server-Host-Schlüssel auswählt.

## Zusätzliche Schlüsselinformationen zum Server-Host

Sie können einen Hostschlüssel auswählen, um Details für diesen Schlüssel anzuzeigen.




Sie können einen Hostschlüssel löschen oder seine Beschreibung im Aktionsmenü auf dem Bildschirm mit den Serverdetails bearbeiten. Wählen Sie den Host-Schlüssel und dann die entsprechende Aktion aus dem Menü aus.

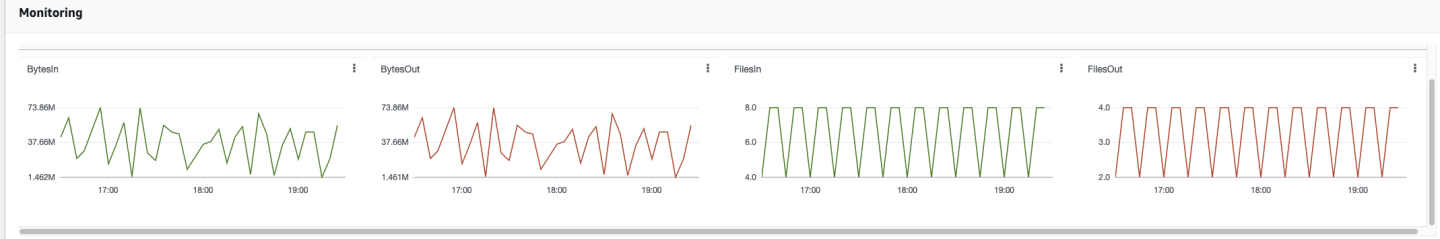
# Überwachung der Nutzung in der Konsole

Informationen zu den Messwerten Ihres Servers finden Sie auf der Seite mit den Serverdetails. Auf diese Weise haben Sie einen zentralen Ort, an dem Sie Ihre Workloads bei Dateiübertragungen überwachen können. Mithilfe eines zentralen Dashboards können Sie verfolgen, wie viele Dateien Sie mit Ihren Partnern ausgetauscht haben, und deren Nutzung genau verfolgen. Details hierzu finden Sie unter SFTP-, FTPS- und FTP-Serverdetails anzeigen. In der folgenden Tabelle werden die für Transfer Family verfügbaren Metriken beschrieben.

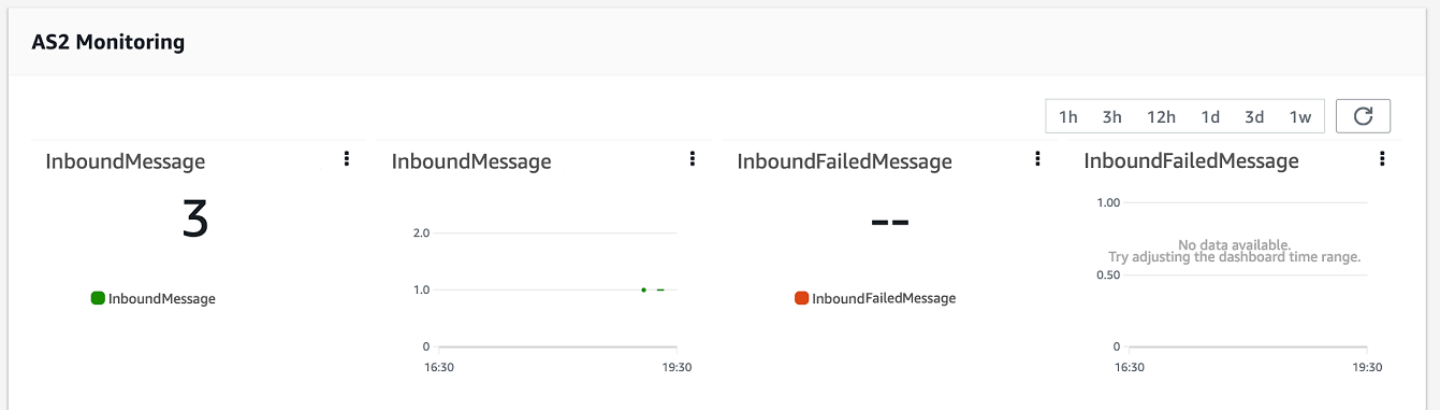
| Namespace | Metrik | Beschreibung |
|---|---|---|
| `AWS/Transfer` | `BytesIn` | Die Gesamtzahl der auf den Server übertragenen Byte.<br><br>Einheiten: Anzahl<br><br>Dauer: 5 Minuten |
| | `BytesOut` | Die Gesamtzahl der vom Server übertragenen Byte.<br><br>Einheit: Anzahl<br><br>Dauer: 5 Minuten |
| | `FilesIn` | Die Gesamtzahl der auf den Server übertragenen Dateien.<br><br>Bei Servern, die das AS2-Protokoll verwenden, stellt diese Metrik die Anzahl der empfangenen Nachrichten dar.<br><br>Einheiten: Anzahl<br><br>Dauer: 5 Minuten |
| | `FilesOut` | Die Gesamtzahl der vom Server übertragenen Dateien.<br><br>Einheiten: Anzahl<br><br>Dauer: 5 Minuten |

| Namespace | Metrik | Beschreibung |
| --- | --- | --- |
| | `InboundMessage` | Die Gesamtzahl der erfolgreich von einem Handelspartner empfangenen AS2-Nachrichten.<br><br>Einheiten: Anzahl<br><br>Dauer: 5 Minuten |
| | `InboundFailedMessage` | Die Gesamtzahl der AS2-Nachrichten, die von einem Handelspartner erfolglos empfangen wurden. Das heißt, ein Handelspartner hat eine Nachricht gesendet, aber der Transfer Family Family-Server konnte sie nicht erfolgreich verarbeiten.<br><br>Einheiten: Anzahl<br><br>Dauer: 5 Minuten |
| | `OnUploadExecutionsStarted` | Die Gesamtzahl der auf dem Server gestarteten Workflow-Ausführungen.<br><br>Einheiten: Anzahl<br><br>Zeitraum: 1 Minute |
| | `OnUploadExecutionsSuccess` | Die Gesamtzahl der erfolgreichen Workflow-Ausführungen auf dem Server.<br><br>Einheiten: Anzahl<br><br>Zeitraum: 1 Minute |
| | `OnUploadExecutionsFailed` | Die Gesamtzahl der erfolglosen Workflow-Ausführungen auf dem Server.<br><br>Einheiten: Anzahl<br><br>Zeitraum: 1 Minute |

Der Abschnitt Überwachung enthält vier einzelne Grafiken. Diese Diagramme zeigen die eingehenden und ausgehenden Byte, die eingehenden Dateien und die ausgehenden Dateien.




Für Server, auf denen das AS2-Protokoll aktiviert ist, befindet sich unter den Überwachungsinformationen ein Abschnitt AS2-Überwachung. Dieser Abschnitt enthält Angaben zur Anzahl der erfolgreichen und fehlgeschlagenen eingehenden Nachrichten.




Um das ausgewählte Diagramm in einem eigenen Fenster zu öffnen, wählen Sie das Erweiterungssymbol (  ).

Sie können auch auf das vertikale Ellipsensymbol () eines Diagramms klicken, um ein Dropdownmenü mit den folgenden Elementen zu öffnen:

- Vergrößern — Öffnet das ausgewählte Diagramm in einem eigenen Fenster.
- Aktualisieren — Lädt das Diagramm mit den neuesten Daten neu.
- In Metriken anzeigen — Öffnet die entsprechenden Metrikdetails in Amazon CloudWatch.
- Protokolle anzeigen — Öffnet die entsprechende Protokollgruppe in CloudWatch.

# Verwaltung der Zugriffskontrollen

Sie können den Zugriff eines Benutzers auf AWS Transfer Family Ressourcen mithilfe einer AWS Identity and Access Management (IAM-) Richtlinie steuern. Eine IAM-Richtlinie ist eine Aussage, in der Regel im JSON-Format, die eine bestimmte Zugriffsebene auf eine Ressource ermöglicht. Sie verwenden eine IAM-Richtlinie, um zu definieren, welche Dateioperationen Sie Ihren Benutzern gestatten möchten und welche nicht. Sie können auch eine IAM-Richtlinie verwenden, um zu definieren, auf welche Amazon S3 S3-Buckets Sie Ihren Benutzern Zugriff gewähren möchten. Um diese Richtlinien für Benutzer festzulegen, erstellen Sie eine IAM-Rolle, der AWS Transfer Family die IAM-Richtlinie und die Vertrauensbeziehung zugeordnet sind.

Jedem Benutzer wird eine IAM-Rolle zugewiesen. Der verwendete IAM-Rollentyp wird als Servicerolle bezeichnet. AWS Transfer Family Wenn sich ein Benutzer bei Ihrem Server anmeldet, AWS Transfer Family nimmt er die dem Benutzer zugeordnete IAM-Rolle an. Weitere Informationen zum Erstellen einer IAM-Rolle, die einem Benutzer Zugriff auf einen Amazon S3 S3-Bucket gewährt, finden Sie unter Creating a role to delegate permissions to an AWS service im IAM-Benutzerhandbuch.

Sie können nur Schreibzugriff auf Amazon S3 S3-Objekte gewähren, indem Sie bestimmte Berechtigungen innerhalb einer IAM-Richtlinie verwenden. Details hierzu finden Sie unter Gewähren Sie die Möglichkeit, nur Dateien zu schreiben und aufzulisten.

Der AWS Storage-Blog enthält einen Beitrag, in dem detailliert beschrieben wird, wie der Zugriff mit den geringsten Rechten eingerichtet wird. Einzelheiten finden Sie unter Implementieren des Zugriffs mit den geringsten Rechten in einem AWS Transfer Family Workflow.

> **Note**
>
> Wenn Ihr Amazon S3 S3-Bucket mit AWS Key Management Service (AWS KMS) verschlüsselt ist, müssen Sie in Ihrer Richtlinie zusätzliche Berechtigungen angeben. Details hierzu finden Sie unter Datenverschlüsselung in Amazon S3. Darüber hinaus finden Sie weitere Informationen zu Sitzungsrichtlinien im IAM-Benutzerhandbuch.

Themen


- Lese- und Schreibzugriff auf einen Amazon S3 S3-Bucket zulassen
- Sitzungsrichtlinie für einen Amazon S3 S3-Bucket erstellen

- Benutzer daran hindern, mkdir in einem S3-Bucket zu laufen

## Lese- und Schreibzugriff auf einen Amazon S3 S3-Bucket zulassen

In diesem Abschnitt wird beschrieben, wie Sie eine IAM-Richtlinie erstellen, die Lese- und Schreibzugriff auf einen bestimmten Amazon S3 S3-Bucket ermöglicht. Wenn Sie Ihrem Benutzer eine IAM-Rolle mit dieser IAM-Richtlinie zuweisen, erhält dieser Benutzer Lese-/Schreibzugriff auf den angegebenen Amazon S3 S3-Bucket.

Die folgende Richtlinie bietet programmatischen Lese-, Schreib- und Tagging-Zugriff auf einen Amazon S3 S3-Bucket. Die `PutObjectACL` Anweisungen `GetObjectACL` und sind nur erforderlich, wenn Sie den kontoübergreifenden Zugriff aktivieren müssen. Das heißt, Ihr Transfer Family Family-Server muss auf einen Bucket in einem anderen Konto zugreifen.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid":"ReadWriteS3",
      "Action": [
            "s3:ListBucket"
                ],
      "Effect": "Allow",
      "Resource": ["arn:aws:s3:::DOC-EXAMPLE-BUCKET"]
    },
    {
      "Effect": "Allow",
      "Action": [
        "s3:PutObject",
        "s3:GetObject",
        "s3:GetObjectTagging",
        "s3:DeleteObject",
        "s3:DeleteObjectVersion",
        "s3:GetObjectVersion",
        "s3:GetObjectVersionTagging",
        "s3:GetObjectACL",
        "s3:PutObjectACL"
      ],
      "Resource": ["arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"]
    }
  ]
```

```
}
```

Die Aktion `ListBucket` setzt die Berechtigung für den Bucket selbst voraus. Die Aktionen `PUT`, `GET` und `DELETE` setzen Objektberechtigungen voraus. Da es sich um unterschiedliche Ressourcen handelt, werden sie mit unterschiedlichen Amazon-Ressourcennamen (ARNs) angegeben.

Um den Zugriff Ihrer Benutzer nur auf das home Präfix des angegebenen Amazon S3 S3-Buckets weiter einzuschränken, sieheSitzungsrichtlinie für einen Amazon S3 S3-Bucket erstellen.

## Sitzungsrichtlinie für einen Amazon S3 S3-Bucket erstellen

Eine Sitzungsrichtlinie ist eine AWS Identity and Access Management (IAM) -Richtlinie, die Benutzer auf bestimmte Bereiche eines Amazon S3 S3-Buckets beschränkt. Dies geschieht durch Evaluierung des Zugriffs in Echtzeit.

> **Note**
>
> Sitzungsrichtlinien werden nur mit Amazon S3 verwendet. Für Amazon EFS verwenden Sie POSIX-Dateiberechtigungen, um den Zugriff einzuschränken.

Sie können eine Sitzungsrichtlinie verwenden, wenn Sie einer Benutzergruppe denselben Zugriff auf einen bestimmten Teil Ihres Amazon S3 S3-Buckets gewähren müssen. Eine Gruppe von Benutzern soll beispielsweise nur auf das home-Verzeichnis zugreifen können. Diese Benutzergruppe teilt sich dieselbe IAM-Rolle.

> **Note**
>
> Die maximale Länge einer Sitzungsrichtlinie beträgt 2048 Zeichen. Weitere Informationen finden Sie unter dem Anforderungsparameter Policy für die `CreateUser` Aktion in der API-Referenz.

Verwenden Sie die folgenden Richtlinienvariablen in Ihrer IAM-Richtlinie, um eine Sitzungsrichtlinie zu erstellen:

- `${transfer:HomeBucket}`

- `${transfer:HomeDirectory}`
- `${transfer:HomeFolder}`
- `${transfer:UserName}`

> **Important**
>
> Sie können die oben genannten Variablen nicht in verwalteten Richtlinien verwenden. Sie können sie auch nicht als Richtlinienvariablen in einer IAM-Rollendefinition verwenden. Sie erstellen diese Variablen in einer IAM-Richtlinie und geben sie direkt bei der Einrichtung Ihres Benutzers an. Außerdem können Sie die `${aws:Username}` Variable in dieser Sitzungsrichtlinie nicht verwenden. Diese Variable bezieht sich auf einen IAM-Benutzernamen und nicht auf den von AWS Transfer Family erforderlichen Benutzernamen.

Der folgende Code zeigt ein Beispiel für eine Sitzungsrichtlinie.

```
{
  "Version": "2012-10-17",
  "Statement": [
      {
          "Sid": "AllowListingOfUserFolder",
          "Action": [
              "s3:ListBucket"
          ],
          "Effect": "Allow",
          "Resource": [
              "arn:aws:s3:::${transfer:HomeBucket}"
          ],
          "Condition": {
              "StringLike": {
                  "s3:prefix": [
                      "${transfer:HomeFolder}/*",
                      "${transfer:HomeFolder}"
                  ]
              }
          }
      },
      {
          "Sid": "HomeDirObjectAccess",
          "Effect": "Allow",
```

```
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
                "s3:DeleteObjectVersion",
                "s3:DeleteObject",
                "s3:GetObjectVersion",
                "s3:GetObjectACL",
                "s3:PutObjectACL"
            ],
            "Resource": "arn:aws:s3:::${transfer:HomeDirectory}/*"
        }
    ]
}
```

> **Note**
>
> Im obigen Richtlinienbeispiel wird davon ausgegangen, dass die Basisverzeichnisse der Benutzer so eingestellt sind, dass sie einen abschließenden Schrägstrich enthalten, um anzuzeigen, dass es sich um ein Verzeichnis handelt. Wenn Sie dagegen das eines Benutzers `HomeDirectory` ohne den abschließenden Schrägstrich angeben, sollten Sie es in Ihre Richtlinie aufnehmen.

Beachten Sie in der vorherigen Beispielrichtlinie die Verwendung der `transfer:HomeFolder` `transfer:HomeDirectory` Richtlinienparameter`transfer:HomeBucket`, und. Diese Parameter werden für den festgelegt`HomeDirectory`, der für den Benutzer konfiguriert ist, wie unter HomeDirectoryund beschriebenImplementierung Ihrer API-Gateway-Methode. Diese Parameter haben die folgenden Definitionen:

- Der `transfer:HomeBucket` Parameter wird durch die erste Komponente von ersetzt`HomeDirectory`.
- Der `transfer:HomeFolder` Parameter wird durch die verbleibenden Teile des `HomeDirectory` Parameters ersetzt.
- Für den `transfer:HomeDirectory` Parameter wurde der führende Schrägstrich (`/`) entfernt, sodass er als Teil eines S3-Amazon-Ressourcennamens (ARN) in einer `Resource` Anweisung verwendet werden kann.

> **Note**
>
> Wenn Sie logische Verzeichnisse verwenden, also die Verzeichnisse des `homeDirectoryType` Benutzers, werden `LOGICAL` diese Richtlinienparameter (`HomeBucketHomeDirectory`, und`HomeFolder`) nicht unterstützt.

Nehmen wir beispielsweise an, dass der `HomeDirectory` Parameter, der für den Transfer Family Family-Benutzer konfiguriert ist, lautet`/home/bob/amazon/stuff/`.

- `transfer:HomeBucket`ist auf eingestellt`/home`.
- `transfer:HomeFolder`ist auf gesetzt`/bob/amazon/stuff/`.
- `transfer:HomeDirectory`wirdhome`/bob/amazon/stuff/`.

Die erste `"Sid"` ermöglicht es dem Benutzer, alle Verzeichnisse aufzulisten, beginnend mit`/home/bob/amazon/stuff/`.

Die zweite `"Sid"` schränkt den `get` Zugriff des Benutzers `put` auf denselben Pfad ein,`/home/bob/amazon/stuff/`.

Wenn die oben genannte Richtlinie gilt, kann ein Benutzer, wenn er sich anmeldet, nur auf Objekte in seinem Home-Verzeichnis zugreifen. AWS Transfer Family Ersetzt diese Variablen beim Verbindungsaufbau durch die entsprechenden Werte für den Benutzer. Dies erleichtert die Verwendung von Richtliniendokumenten für mehrere Benutzer. Dieser Ansatz reduziert den Aufwand für die IAM-Rollen- und Richtlinienverwaltung für die Verwaltung des Zugriffs Ihrer Benutzer auf Ihren Amazon S3 S3-Bucket.

Sie können auch eine Sitzungsrichtlinie verwenden, um den Zugriff für jeden Ihrer Benutzer an Ihre Geschäftsanforderungen anzupassen. Weitere Informationen finden Sie unter Berechtigungen für AssumeRole, AssumeRoleWith SAML und AssumeRoleWithWebIdentity im IAM-Benutzerhandbuch.

> **Note**
>
> AWS Transfer Family speichert den Richtlinien-JSON anstelle des Amazon-Ressourcennamens (ARN) der Richtlinie. Wenn Sie also die Richtlinie in der IAM-Konsole ändern, müssen Sie zur AWS Transfer Family Konsole zurückkehren und Ihre Benutzer über die neuesten Richtlinieninhalte informieren. Sie können den Benutzer auf der Registerkarte Richtlinieninformationen im Abschnitt Benutzerkonfiguration aktualisieren.

Wenn Sie den verwenden AWS CLI, können Sie den folgenden Befehl verwenden, um die Richtlinie zu aktualisieren.

```
aws transfer update-user --server-id server --user-name user --policy \
   "$(aws iam get-policy-version --policy-arn policy --version-id version --
output json)"
```

## Benutzer daran hindern, **`mkdir`** in einem S3-Bucket zu laufen

Sie können die Fähigkeit von Benutzern einschränken, ein Verzeichnis in einem Amazon S3 S3-Bucket zu erstellen. Dazu erstellen Sie eine IAM-Richtlinie, die die `s3:PutObject` Aktion zulässt, sie aber auch verweigert, wenn der Schlüssel mit einem „/“ (Schrägstrich) endet. Die folgende Beispielrichtlinie ermöglicht es Benutzern, Dateien in einen Amazon S3 S3-Bucket hochzuladen, verweigert jedoch den mkdir Befehl im Amazon S3 S3-Bucket.

```
{
   "Sid":"DenyMkdir",
   "Action":[
      "s3:PutObject"
   ],
   "Effect":"Deny",
   "Resource":[
      "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*/",
      "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*/*"
    ]
}
```

**Note**

Die zweite Ressourcenzeile macht es Benutzern unmöglich, Unterordner zu erstellen, indem sie einen Befehl wie ausführen. `put my-file DOC-EXAMPLE-BUCKET/new-folder/my-file`

# Protokollierung für AWS Transfer Family

AWS Transfer Family lässt sich sowohl in AWS CloudTrail als auch in Amazon integrieren CloudWatch CloudTrail und CloudWatch dient verschiedenen, aber ergänzenden Zwecken:

- CloudTrail ist ein -AWSService, der eine Aufzeichnung der in Ihrem durchgeführten Aktionen erstelltAWS-Konto. Es überwacht und zeichnet API-Aufrufe kontinuierlich auf Aktivitäten wie Konsolenanmeldungen, AWS Command Line Interface Befehle und SDK/API-Aufrufe auf. Auf diese Weise können Sie ein Protokoll darüber führen, wer welche Maßnahmen wann und von wo aus ergriffen hat. CloudTrail hilft bei der Prüfung, dem Zugriffsmanagement und der Einhaltung gesetzlicher Vorschriften, indem es einen Verlauf aller Aktivitäten in Ihrer -AWSUmgebung bereitstellt. Weitere Informationen finden Sie im AWS CloudTrail -Benutzerhandbuch.
- CloudWatch ist ein Überwachungsservice für AWS Ressourcen und Anwendungen. Es erfasst Metriken und Protokolle, um einen Einblick in die Ressourcenauslastung, die Anwendungsleistung und den allgemeinen Systemzustand zu gewähren. CloudWatch helps bei Betriebsaufgaben wie der Behebung von Problemen, dem Einstellen von Alarmen und Auto Scaling. Weitere Informationen finden Sie im Amazon- CloudWatch Benutzerhandbuch.

Themen


- AWS CloudTrail -Protokollierung für AWS Transfer Family
- Amazon CloudWatch loggt sich ein für AWS Transfer Family


## AWS CloudTrail -Protokollierung für AWS Transfer Family

AWS Transfer Family ist in integriert, einem ServiceAWS CloudTrail, der die Aktionen eines Benutzers, einer Rolle oder eines -AWSServices in aufzeichnetAWS Transfer Family. CloudTrail erfasst alle API-Aufrufe für AWS Transfer Family als Ereignisse. Zu den erfassten Aufrufen gehören Aufrufe von der AWS Transfer Family-Konsole und Code-Aufrufe der AWS Transfer Family-API-Operationen.

Ein Trail ist eine Konfiguration, die die Bereitstellung von Ereignissen als Protokolldateien an einen von Ihnen angegebenen Amazon 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, das Datum und die Uhrzeit der

Aktion, Anforderungsparameter usw. CloudTrail Protokolldateien sind kein geordnetes Stacktrace der öffentlichen API-Aufrufe und erscheinen daher nicht in einer bestimmten Reihenfolge.

Zur kontinuierlichen Aufzeichnung von Ereignissen in Ihrem AWS-Konto, einschließlich Ereignissen für AWS Transfer Family, erstellen Sie einen Trail. Ein Trail ermöglicht CloudTrail die Bereitstellung von Protokolldateien an einen Amazon S3-Bucket. Wenn Sie einen Trail in der Konsole anlegen, gilt dieser für alle AWS-Regionen. Der Trail protokolliert Ereignisse aus allen Regionen in der AWS-Partition und stellt die Protokolldateien in dem von Ihnen angegebenen Amazon-S3-Bucket bereit. Darüber hinaus können Sie andere -AWSServices konfigurieren, um die in den CloudTrail Protokollen erfassten Ereignisdaten weiter zu analysieren und entsprechend zu agieren. Weitere Informationen finden Sie hier:

- Übersicht zum Erstellen eines Trails
- CloudTrail Von unterstützte Services und Integrationen
- Konfigurieren von Amazon SNS-Benachrichtigungen für CloudTrail
- Empfangen von CloudTrail Protokolldateien aus mehreren Regionen und Empfangen von CloudTrail Protokolldateien aus mehreren Konten

Alle -AWS Transfer FamilyAktionen werden von protokolliert CloudTrail und sind in der dokumentiertActionsAPI reference. Aufrufe der `StopServer` Aktionen `CreateServer`, `ListUsers` und erzeugen beispielsweise Einträge in den CloudTrail Protokolldateien.

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 Anforderung mit Root- oder AWS Identity and Access Management-Benutzeranmeldeinformationen ausgeführt wurde.
- Gibt an, ob die Anforderung mit temporären Sicherheitsanmeldeinformationen für eine Rolle oder einen Verbundbenutzer gesendet wurde.
- Ob die Anforderung aus einem anderen AWS-Service gesendet wurde

Weitere Informationen finden Sie unter CloudTrail -Element userIdentity.

Wenn Sie einen Trail erstellen, können Sie die kontinuierliche Bereitstellung von CloudTrail Ereignissen an einen Amazon S3-Bucket aktivieren, einschließlich Ereignissen für AWS Transfer Family. Wenn Sie keinen Trail konfigurieren, können Sie trotzdem die neuesten Ereignisse in der CloudTrail Konsole unter Ereignisverlauf anzeigen.

Anhand der von CloudTrailgesammelten Informationen können Sie die an gestellte AnfrageAWS Transfer Family, die IP-Adresse, von der die Anfrage gestellt wurde, den Initiator der Anfrage, den Zeitpunkt der Anfrage und zusätzliche Details bestimmen.

Weitere Informationen zu CloudTrailfinden Sie im AWS CloudTrail -Benutzerhandbuch.

Themen


- Aktivieren der AWS CloudTrail Protokollierung
- Beispielprotokolleintrag zum Erstellen eines Servers


## Aktivieren der AWS CloudTrail Protokollierung

Sie können AWS Transfer Family-API-Aufrufe mit AWS CloudTrail überwachen. Durch die Überwachung der API-Aufrufe erhalten Sie nützliche Informationen zu Sicherheit und Betrieb. Wenn Sie die Protokollierung auf Amazon S3-Objektebene aktiviert haben, `RoleSessionName` ist im Feld Anforderer als enthalten`[AWS:Role Unique Identifier]/username.sessionid@server-id`. Weitere Informationen zu eindeutigen Kennungen von AWS Identity and Access Management (IAM)-Rollen finden Sie unter Eindeutige Kennungen im AWS Identity and Access Management - Benutzerhandbuch.

> **Important**
>
> Die maximale Länge des `RoleSessionName` beträgt 64 Zeichen. Wenn länger `RoleSessionName` ist, `server-id` wird die gekürzt.

## Beispielprotokolleintrag zum Erstellen eines Servers

Das folgende Beispiel zeigt einen CloudTrail Protokolleintrag (im JSON-Format), der die `CreateServer` Aktion demonstriert.

```
{
    "eventVersion": "1.09",
    "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AAAA4FFF5HHHHH6NNWWWW:user1",
        "arn": "arn:aws:sts::123456789102:assumed-role/Admin/user1",
        "accountId": "123456789102",
        "accessKeyId": "AAAA52C2WWWWW3BB4Z",
```

```
        "sessionContext": {
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2018-12-18T20:03:57Z"
            },
            "sessionIssuer": {
                "type": "Role",
                "principalId": "AAAA4FFF5HHHHH6NNWWWW",
                "arn": "arn:aws:iam::123456789102:role/Admin",
                "accountId": "123456789102",
                "userName": "Admin"
            }
        }
    },
    "eventTime": "2024-02-05T19:18:53Z",
    "eventSource": "transfer.amazonaws.com",
    "eventName": "CreateServer",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "11.22.1.2",
    "userAgent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
 like Gecko) Chrome/121.0.0.0 Safari/537.36",
    "requestParameters": {
        "domain": "S3",
        "hostKey": "HIDDEN_DUE_TO_SECURITY_REASONS",
        "protocols": [
            "SFTP"
        ],
        "protocolDetails": {
            "passiveIp": "AUTO",
            "tlsSessionResumptionMode": "ENFORCED",
            "setStatOption": "DEFAULT"
        },
        "securityPolicyName": "TransferSecurityPolicy-2020-06",
        "s3StorageOptions": {
            "directoryListingOptimization": "ENABLED"
        }
    },
    "responseElements": {
        "serverId": "s-1234abcd5678efghi"
    },
    "requestID": "6fe7e9b1-72fc-45b0-a7f9-5840268aeadf",
    "eventID": "4781364f-7c1e-464e-9598-52d06aa9e63a",
    "readOnly": false,
    "eventType": "AwsApiCall",
```

```
    "managementEvent": true,
    "recipientAccountId": "123456789102",
    "eventCategory": "Management",
    "tlsDetails": {
        "tlsVersion": "TLSv1.3",
        "cipherSuite": "TLS_AES_128_GCM_SHA256",
        "clientProvidedHostHeader": "transfer.us-east-1.amazonaws.com"
    },
    "sessionCredentialFromConsole": "true"
}
```

# Amazon CloudWatch loggt sich ein für AWS Transfer Family

Amazon CloudWatch überwacht Ihre AWS Transfer Family Ressourcen und die Anwendungen, auf denen Sie laufen, AWS in Echtzeit. Sie können CloudWatch damit Metriken sammeln und verfolgen. Dabei handelt es sich um Variablen, die Sie für Ihre Ressourcen und Anwendungen messen können.

Auf der CloudWatch Startseite werden automatisch Metriken zu Transfer Family und allen anderen von Ihnen verwendeten AWS Diensten angezeigt. Sie können zusätzlich benutzerdefinierte Dashboards erstellen, um Metriken über Ihre benutzerdefinierten Anwendungen anzuzeigen und benutzerdefinierte Sammlungen von Metriken Ihrer Wahl anzuzeigen.

Sie können Alarme erstellen, die Metriken überwachen und Benachrichtigungen senden oder automatisch Änderungen an den Ressourcen vornehmen, die Sie überwachen, wenn ein Schwellenwert überschritten wird. Sie können beispielsweise die Dateien überwachen, die auf einen Transfer Family Family-Server übertragen werden, und anhand dieser Daten feststellen, ob Sie zusätzliche Server bereitstellen müssen, um die erhöhte Last zu bewältigen. Sie können diese Daten auch verwenden, um nicht ausgelastete Instanzen zu stoppen oder zu löschen, um Geld zu sparen.

## Arten der CloudWatch Protokollierung für Transfer Family

Transfer Family bietet zwei Möglichkeiten, Ereignisse zu protokollieren CloudWatch:

- Strukturierte JSON-Protokollierung
- Protokollierung über eine Logging-Rolle

Für Transfer Family Family-Server können Sie den Protokollierungsmechanismus wählen, den Sie bevorzugen. Für Konnektoren und Workflows werden nur Protokollierungsrollen unterstützt.

### Strukturierte JSON-Protokollierung

Für die Protokollierung von Serverereignissen empfehlen wir die Verwendung der strukturierten JSON-Protokollierung. Dies bietet ein umfassenderes Protokollierungsformat, das das Abfragen von CloudWatch Protokollen ermöglicht. Für diese Art der Protokollierung muss die IAM-Richtlinie für den Benutzer, der den Server erstellt (oder die Protokollierungskonfiguration des Servers bearbeitet), die folgenden Berechtigungen enthalten:

- `logs:CreateLogDelivery`
- `logs:DeleteLogDelivery`
- `logs:DescribeLogGroups`
- `logs:DescribeResourcePolicies`
- `logs:GetLogDelivery`
- `logs:ListLogDeliveries`
- `logs:PutResourcePolicy`
- `logs:UpdateLogDelivery`

Es folgt eine Beispielrichtlinie .

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogDelivery",
                "logs:GetLogDelivery",
                "logs:UpdateLogDelivery",
                "logs:DeleteLogDelivery",
                "logs:ListLogDeliveries",
                "logs:PutResourcePolicy",
                "logs:DescribeResourcePolicies",
                "logs:DescribeLogGroups"
            ],
            "Resource": "arn:aws:logs:region-id:AWS-Konto:log-group:/aws/transfer/*"
        }
    ]
```

```
}
```

Einzelheiten zur Einrichtung der strukturierten JSON-Protokollierung finden Sie unter. Protokollierung für Server erstellen, aktualisieren und anzeigen

Rolle „Protokollierung“

Um Ereignisse für einen verwalteten Workflow, der an einen Server angehängt ist, sowie für Connectors zu protokollieren, müssen Sie eine Protokollierungsrolle angeben. Um den Zugriff festzulegen, erstellen Sie eine ressourcenbasierte IAM-Richtlinie und eine IAM-Rolle, die diese Zugriffsinformationen bereitstellt. Im Folgenden finden Sie ein Beispiel für eine Richtlinie, die Serverereignisse AWS-Konto protokollieren kann.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogStream",
                "logs:DescribeLogStreams",
                "logs:CreateLogGroup",
                "logs:PutLogEvents"
            ],
            "Resource": "arn:aws:logs:*:*:log-group:/aws/transfer/*"
        }
    ]
}
```

Einzelheiten zur Konfiguration einer Protokollierungsrolle zum Protokollieren von Workflow-Ereignissen finden Sie unterVerwaltung der Protokollierung für Workflows.

Themen

- Protokollierung für Server erstellen, aktualisieren und anzeigen
- Verwaltung der Protokollierung für Workflows
- Konfigurieren Sie die CloudWatch Protokollierungsrolle
- Protokollstreams von Transfer Family anzeigen
- CloudWatch Amazon-Alarme erstellen

- Protokollierung Amazon S3 S3-API-Aufrufen in S3-Zugriffsprotokollen
- Beispiele zur Begrenzung des Problems mit verwirrtem Stellvertreter
- CloudWatch -Protokollstruktur für Transfer Family
- Beispiel für CloudWatch Protokolleinträge
- CloudWatch Metriken für Transfer Family verwenden
- Wird AWS-Benutzerbenachrichtigungen mit verwendet AWS Transfer Family
- Verwenden von Abfragen zum Filtern von Protokolleinträgen

## Protokollierung für Server erstellen, aktualisieren und anzeigen

Für alle AWS Transfer Family Server können Sie zwischen zwei Optionen für die Protokollierung wählen: `LoggingRole` (wird für die Protokollierung von Workflows verwendet, die an den Server angehängt sind) oder`StructuredLogDestinations`. `StructuredLogDestinations` bietet folgende Vorteile:

- Empfangen Sie Protokolle in einem strukturierten JSON-Format.
- Fragen Sie Ihre Logs mit Amazon CloudWatch Logs Insights ab, das automatisch Felder im JSON-Format erkennt.
- Durch die gemeinsame Nutzung von Protokollgruppen für mehrere AWS Transfer Family Ressourcen können Sie Protokollstreams von mehreren Servern zu einer einzigen Protokollgruppe zusammenfassen, was die Verwaltung Ihrer Überwachungskonfigurationen und Einstellungen für die Aufbewahrung von Protokollen erleichtert.
- Erstellen Sie aggregierte Metriken und Visualisierungen, die zu Dashboards hinzugefügt werden können. CloudWatch
- Verfolgen Sie Nutzungs- und Leistungsdaten, indem Sie mithilfe von Protokollgruppen konsolidierte Protokollmetriken, Visualisierungen und Dashboards erstellen.

Die Optionen für `LoggingRole` oder `StructuredLogDestinations` werden separat konfiguriert und gesteuert. Sie können für jeden Server eine oder beide Protokollierungsmethoden einrichten oder Ihren Server so konfigurieren, dass keinerlei Protokollierung erfolgt (obwohl dies nicht empfohlen wird).

Wenn Sie mit der Transfer Family Family-Konsole einen neuen Server erstellen, ist die Protokollierung standardmäßig aktiviert. Nachdem Sie den Server erstellt haben, können Sie

den `UpdateServer` API-Aufruf verwenden, um Ihre Protokollierungskonfiguration zu ändern. Einzelheiten finden Sie unter StructuredLogZiele.

Derzeit müssen Sie für Workflows eine Protokollierungsrolle angeben, wenn Sie die Protokollierung aktivieren möchten:

- Wenn Sie einen Workflow mit einem Server verknüpfen, indem Sie entweder den API-Aufruf `CreateServer` oder den `UpdateServer` API-Aufruf verwenden, erstellt das System nicht automatisch eine Protokollierungsrolle. Wenn Sie Ihre Workflow-Ereignisse protokollieren möchten, müssen Sie dem Server explizit eine Protokollierungsrolle zuweisen.
- Wenn Sie mit der Transfer Family Family-Konsole einen Server erstellen und einen Workflow anhängen, werden Protokolle an eine Protokollgruppe gesendet, deren Name die Server-ID enthält. Das Format ist `/aws/transfer/`*`server-id`* beispielsweise`/aws/transfer/s-1111aaaa2222bbbb3`. Die Serverprotokolle können an dieselbe oder eine andere Protokollgruppe gesendet werden.

Überlegungen zur Protokollierung beim Erstellen und Bearbeiten von Servern in der Konsole

- Neue Server, die über die Konsole erstellt wurden, unterstützen nur die strukturierte JSON-Protokollierung, es sei denn, ein Workflow ist an den Server angehängt.
- Keine Protokollierung ist keine Option für neue Server, die Sie in der Konsole erstellen.
- Bestehende Server können die strukturierte JSON-Protokollierung jederzeit über die Konsole aktivieren.
- Durch die Aktivierung der strukturierten JSON-Protokollierung über die Konsole wird die bestehende Protokollierungsmethode deaktiviert, sodass Kunden nicht doppelt belastet werden. Die Ausnahme ist, wenn ein Workflow an den Server angehängt ist.
- Wenn Sie die strukturierte JSON-Protokollierung aktivieren, können Sie sie später nicht über die Konsole deaktivieren.
- Wenn Sie die strukturierte JSON-Protokollierung aktivieren, können Sie das Ziel der Protokollgruppe jederzeit über die Konsole ändern.
- Wenn Sie die strukturierte JSON-Protokollierung aktivieren, können Sie die Protokollierungsrolle nicht über die Konsole bearbeiten, wenn Sie beide Protokollierungstypen über die API aktiviert haben. Die Ausnahme ist, wenn an Ihren Server ein Workflow angehängt ist. Die Rolle „Protokollierung“ wird jedoch weiterhin unter Zusätzliche Details angezeigt.

Überlegungen zur Protokollierung beim Erstellen und Bearbeiten von Servern mithilfe der API oder des SDK

- Wenn Sie über die API einen neuen Server erstellen, können Sie eine oder beide Arten der Protokollierung konfigurieren oder keine Protokollierung auswählen.
- Für bestehende Server können Sie die strukturierte JSON-Protokollierung jederzeit aktivieren und deaktivieren.
- Sie können die Protokollgruppe jederzeit über die API ändern.
- Sie können die Protokollierungsrolle jederzeit über die API ändern.

Um die strukturierte Protokollierung zu aktivieren, müssen Sie bei einem Konto mit den folgenden Berechtigungen angemeldet sein

- `logs:CreateLogDelivery`
- `logs:DeleteLogDelivery`
- `logs:DescribeLogGroups`
- `logs:DescribeResourcePolicies`
- `logs:GetLogDelivery`
- `logs:ListLogDeliveries`
- `logs:PutResourcePolicy`
- `logs:UpdateLogDelivery`

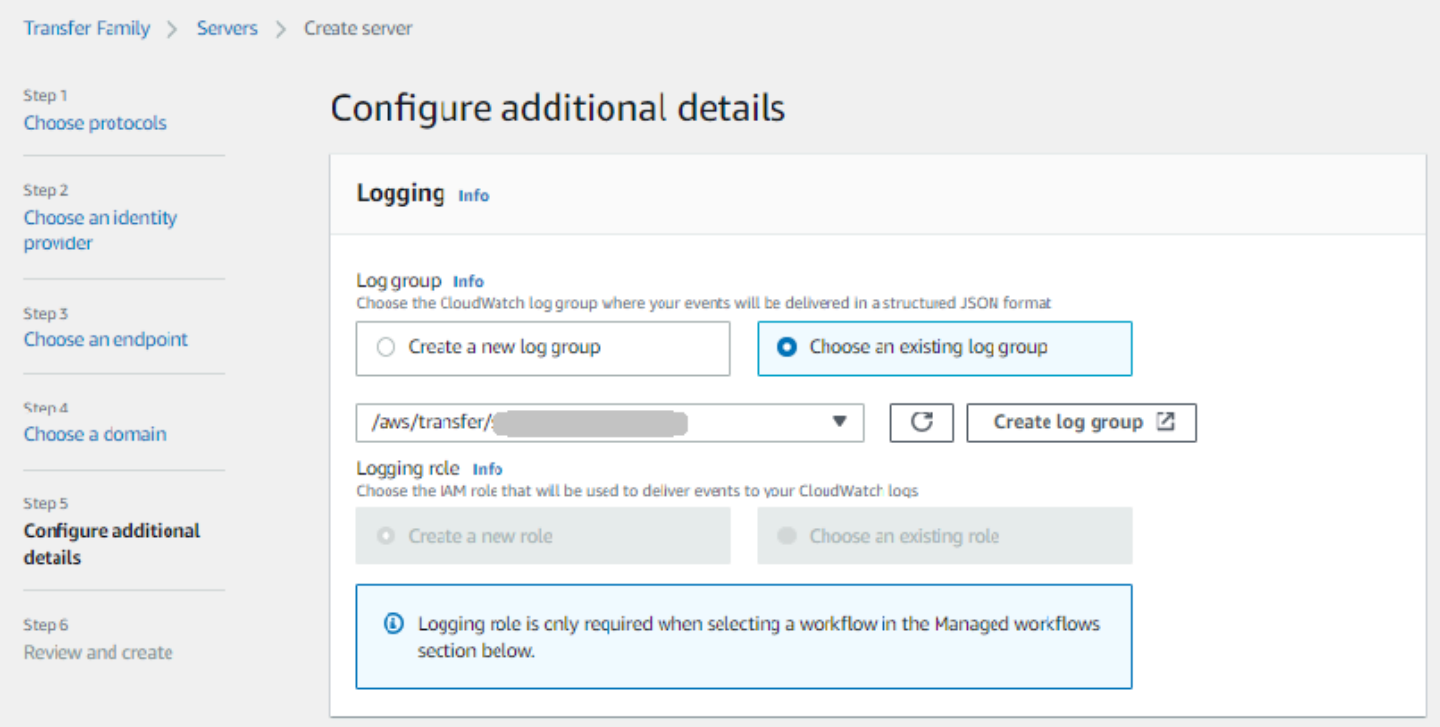
Eine Beispielrichtlinie ist im Abschnitt verfügbarKonfigurieren Sie die CloudWatch Protokollierungsrolle.

Themen


- Protokollierung für Server erstellen
- Die Protokollierung für einen Server aktualisieren
- Serverkonfiguration anzeigen


## Protokollierung für Server erstellen

Wenn Sie einen neuen Server erstellen, können Sie auf der Seite Zusätzliche Details konfigurieren eine vorhandene Protokollgruppe angeben oder eine neue erstellen.




Wenn Sie „Protokollgruppe erstellen“ wählen, wird in der CloudWatch Konsole (https://console.aws.amazon.com/cloudwatch/) die Seite „Protokollgruppe erstellen“ geöffnet. Einzelheiten finden Sie unter Protokollgruppe erstellen in CloudWatch Logs.

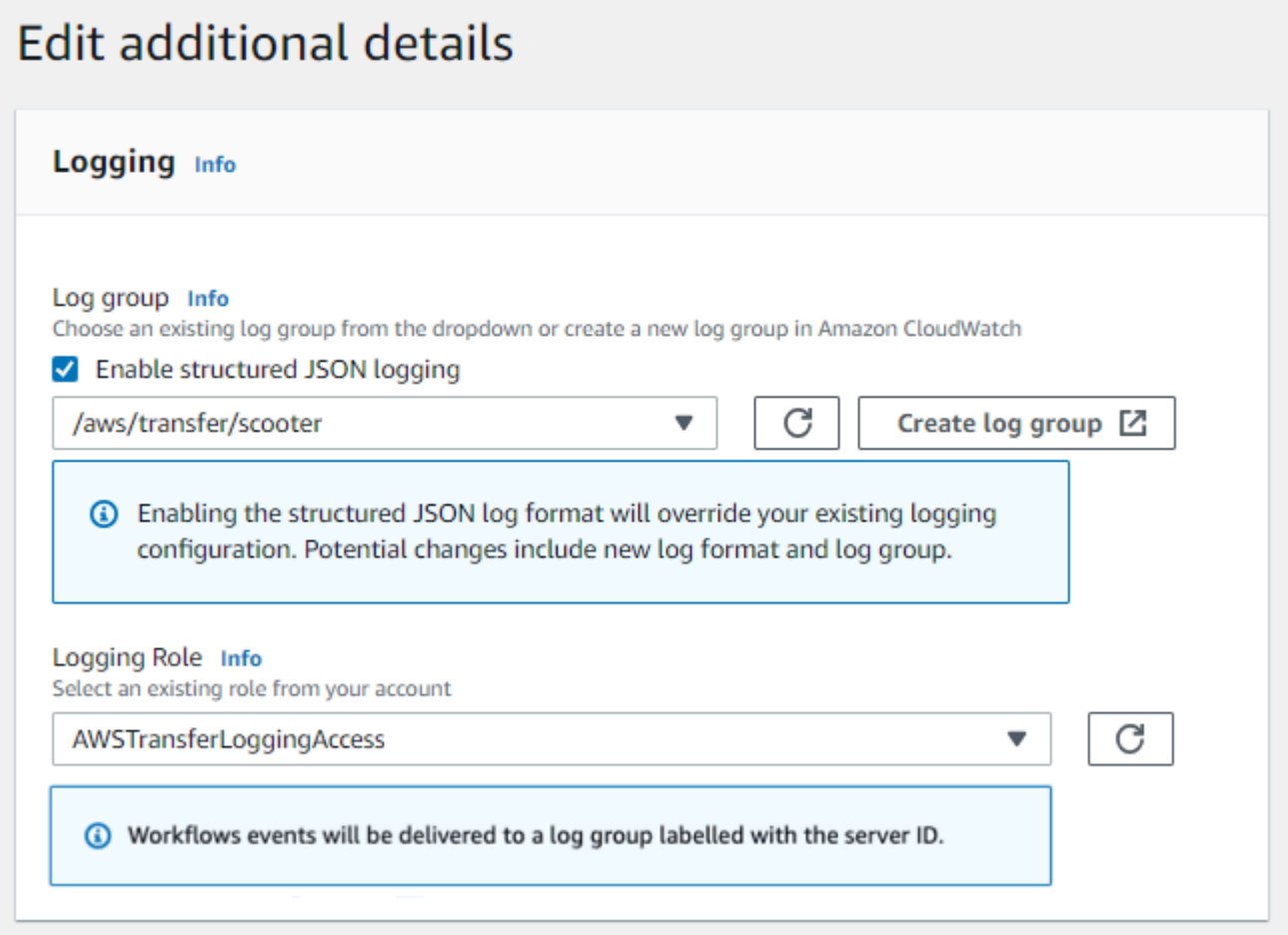
## Die Protokollierung für einen Server aktualisieren

Die Einzelheiten der Protokollierung hängen vom Szenario für Ihr Update ab.

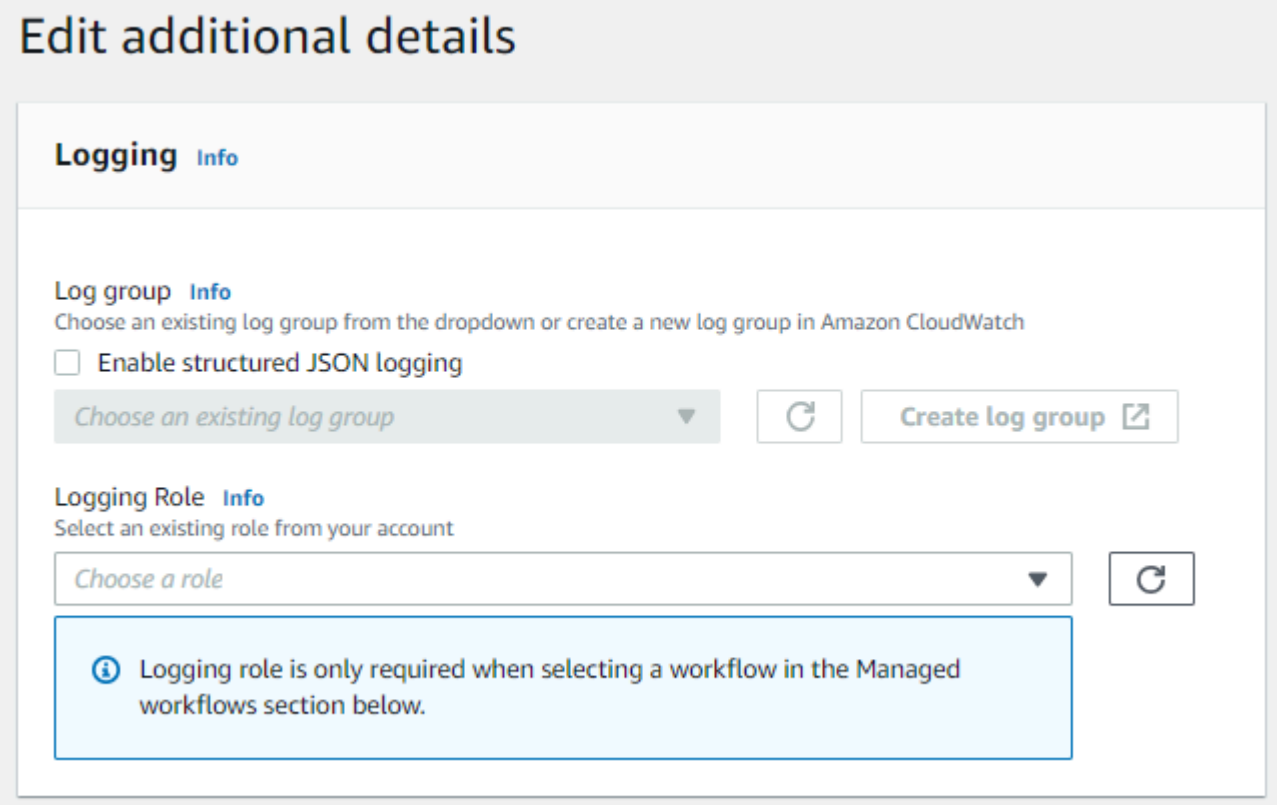
> **Note**
>
> Wenn Sie sich für die strukturierte JSON-Protokollierung entscheiden, kann es in seltenen Fällen zu Verzögerungen kommen, bei denen Transfer Family die Protokollierung im alten Format beendet, es jedoch einige Zeit dauert, bis die Protokollierung im neuen JSON-Format gestartet wird. Dies kann dazu führen, dass Ereignisse nicht protokolliert werden. Es wird keine Dienstunterbrechungen geben, aber Sie sollten vorsichtig sein, wenn Sie Dateien in der ersten Stunde nach der Änderung Ihrer Protokollierungsmethode übertragen, da Protokolle gelöscht werden könnten.

Wenn Sie einen vorhandenen Server bearbeiten, hängen Ihre Optionen vom Status des Servers ab.

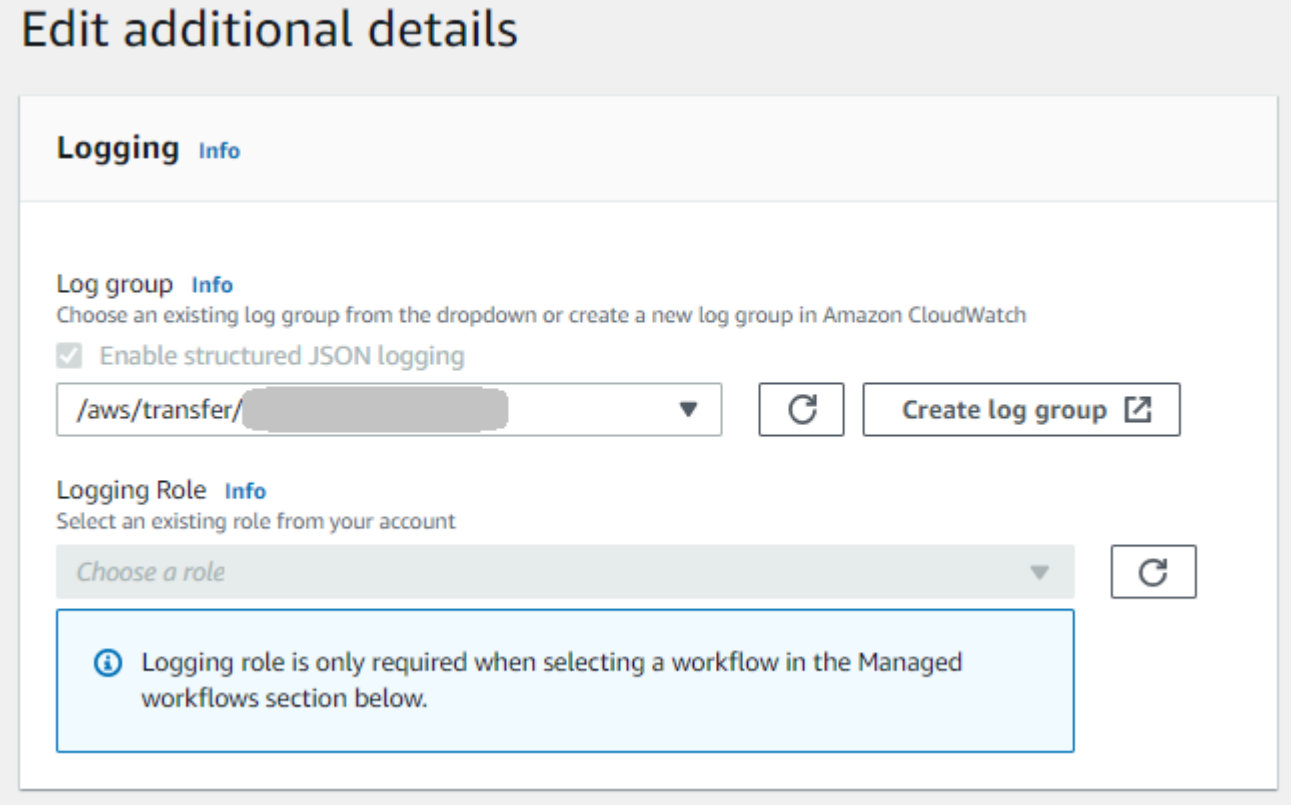
- Auf dem Server ist bereits eine Protokollierungsrolle aktiviert, aber die strukturierte JSON-Protokollierung ist nicht aktiviert.



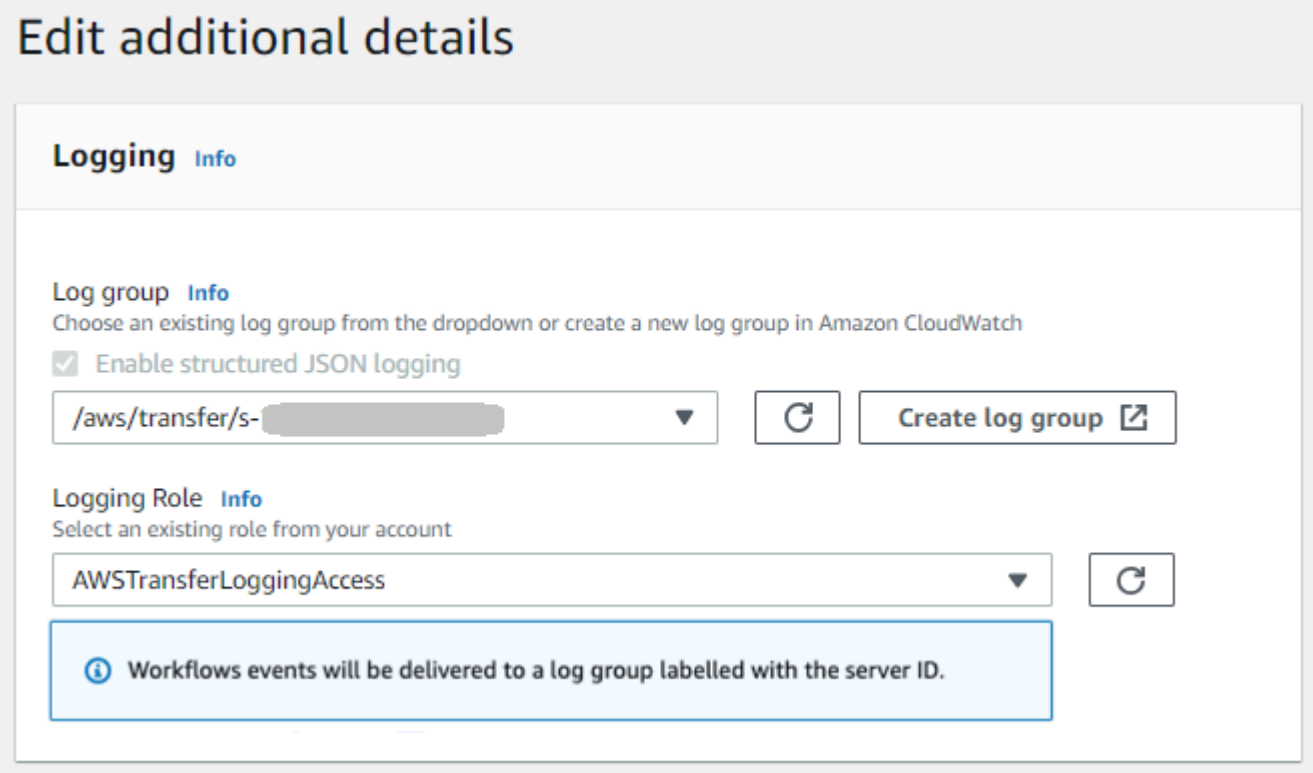

- Auf dem Server ist keine Protokollierung aktiviert.




- Auf dem Server ist die strukturierte JSON-Protokollierung bereits aktiviert, es wurde jedoch keine Protokollierungsrolle angegeben.



- Auf dem Server ist die strukturierte JSON-Protokollierung bereits aktiviert, und es wurde auch eine Protokollierungsrolle angegeben.

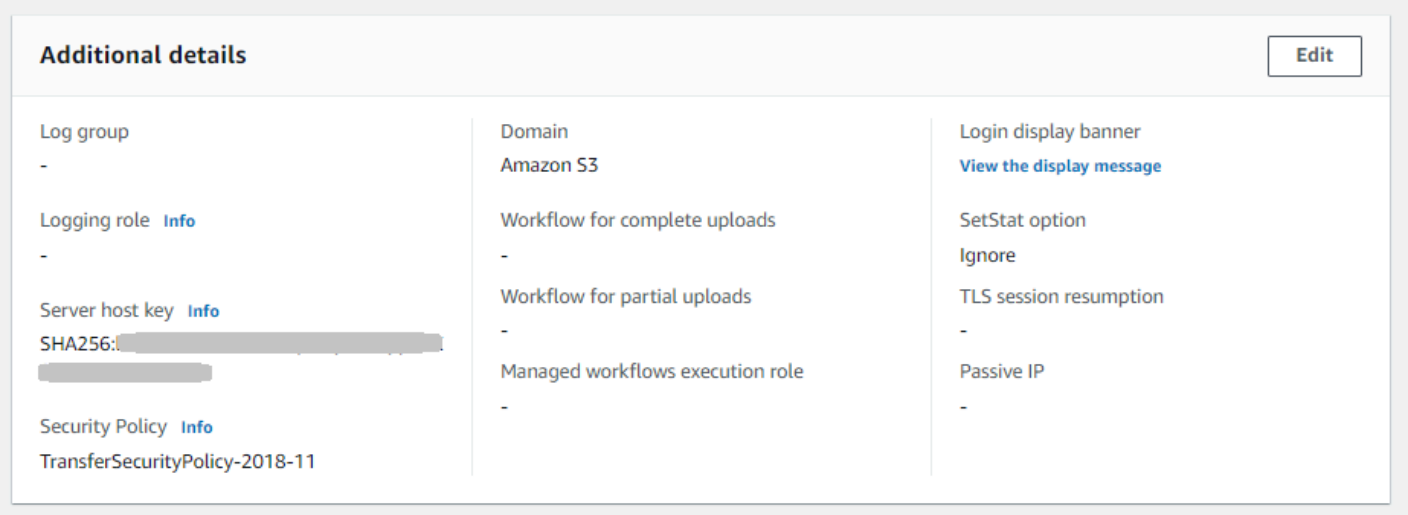



## Serverkonfiguration anzeigen

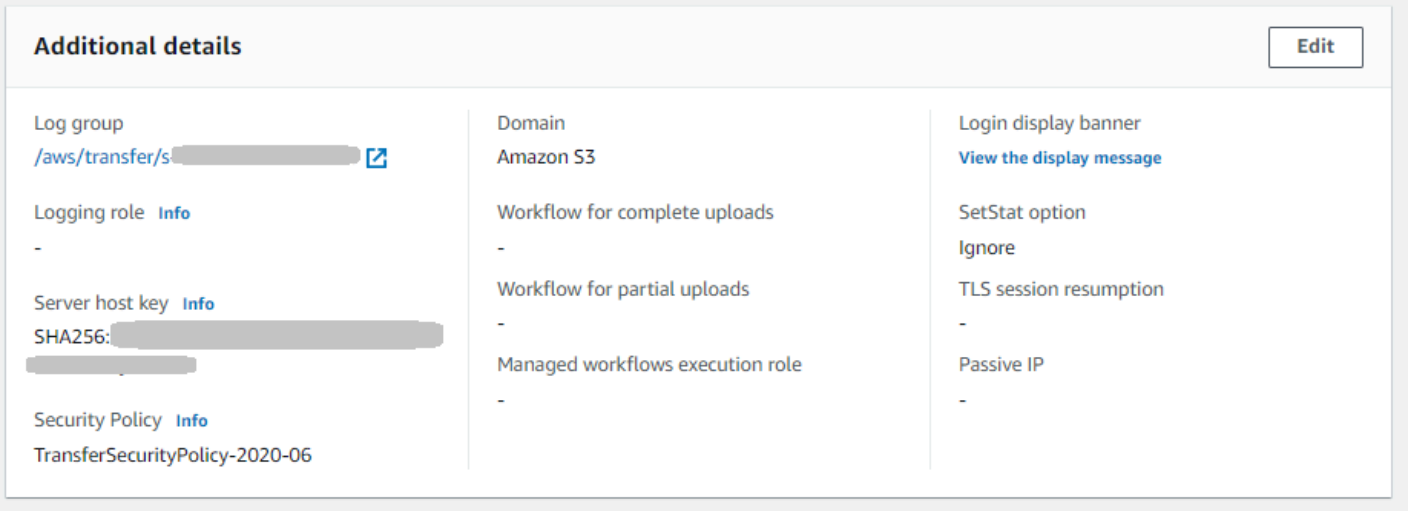
Die Details für die Serverkonfigurationsseite hängen von Ihrem Szenario ab:

Je nach Szenario könnte die Serverkonfigurationsseite wie eines der folgenden Beispiele aussehen:

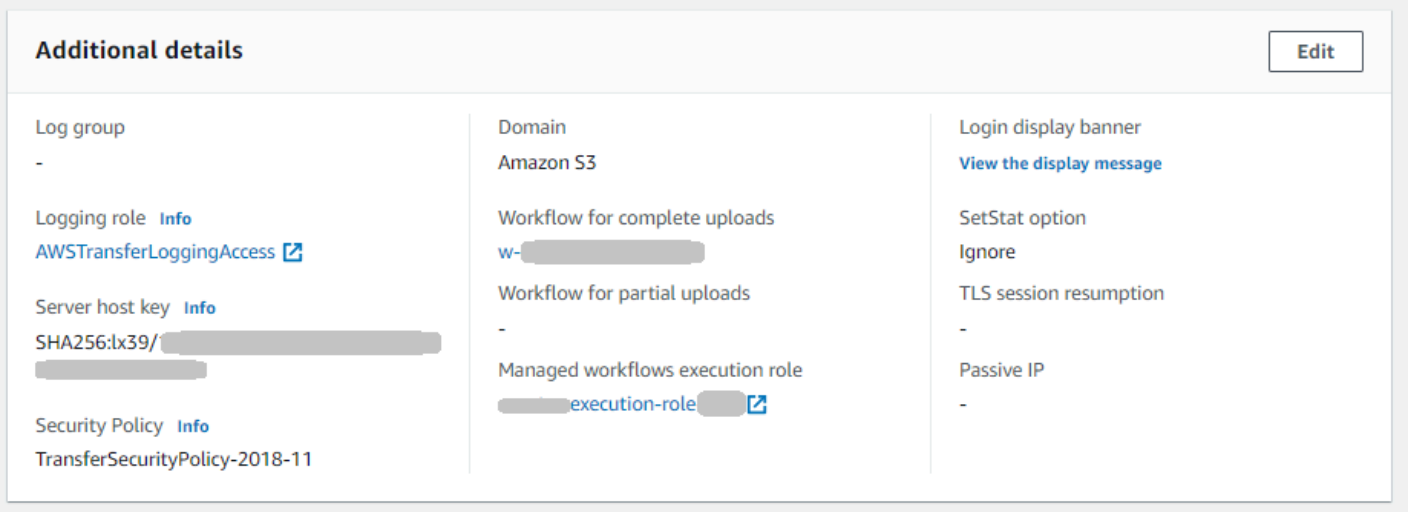
- Es ist keine Protokollierung aktiviert.



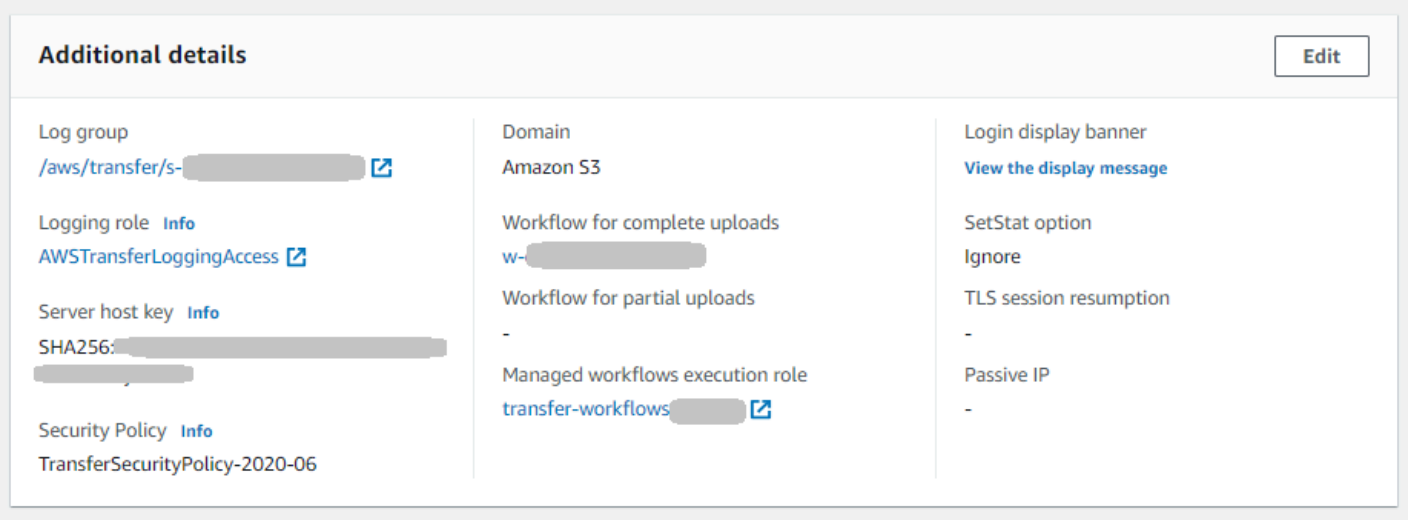

- Die strukturierte JSON-Protokollierung ist aktiviert.




- Die Protokollierungsrolle ist aktiviert, aber die strukturierte JSON-Protokollierung ist nicht aktiviert.



- Beide Arten der Protokollierung (Protokollierungsrolle und strukturierte JSON-Protokollierung) sind aktiviert.




## Verwaltung der Protokollierung für Workflows

CloudWatch bietet eine konsolidierte Prüfung und Protokollierung des Fortschritts und der Ergebnisse von Workflows. Darüber hinaus AWS Transfer Family bietet es mehrere Metriken für Workflows. Sie können Messwerte darüber anzeigen, wie viele Workflow-Ausführungen in der letzten Minute gestartet, erfolgreich abgeschlossen und fehlgeschlagen sind. Alle CloudWatch Metriken für Transfer Family werden unter beschrieben CloudWatch Metriken für Transfer Family verwenden.

CloudWatch Amazon-Protokolle für Workflows anzeigen

1. Öffnen Sie die CloudWatch Amazon-Konsole unter https://console.aws.amazon.com/cloudwatch/.
2. Wählen Sie im linken Navigationsbereich Logs und anschließend Log groups aus.
3. Wählen Sie auf der Seite Protokollgruppen in der Navigationsleiste die richtige Region für Ihren AWS Transfer Family Server aus.
4. Wählen Sie die Protokollgruppe aus, die Ihrem Server entspricht.

   Wenn Ihre Server-ID beispielsweise lautet`s-1234567890abcdef0`, ist Ihre Protokollgruppe`/aws/transfer/s-1234567890abcdef0`.
5. Auf der Seite mit den Protokollgruppendetails für Ihren Server werden die neuesten Protokollstreams angezeigt. Es gibt zwei Protokollstreams für den Benutzer, den Sie untersuchen:

- Einer für jede Secure Shell (SSH) File Transfer Protocol (SFTP) -Sitzung.
- Eine für den Workflow, der für Ihren Server ausgeführt wird. Das Format für den Protokollstream für den Workflow ist*username*.*workflowID*.*uniqueStreamSuffix*.

Wenn Ihr Benutzer beispielsweise ist`mary-major`, haben Sie die folgenden Protokollstreams:

```
mary-major-east.1234567890abcdef0
mary.w-abcdef01234567890.021345abcdef6789
```

> **Note**
>
> Die in diesem Beispiel aufgeführten 16-stelligen alphanumerischen Identifikatoren sind fiktiv. Die Werte, die Sie bei Amazon sehen, CloudWatch sind unterschiedlich.

Auf der Seite Ereignisse protokollieren für `mary-major-usa-east.1234567890abcdef0` werden die Details für jede Benutzersitzung angezeigt, und der `mary.w-abcdef01234567890.021345abcdef6789` Protokollstream enthält die Details für den Workflow.

Im Folgenden finden Sie ein Beispiel für einen Protokollstream für`mary.w-abcdef01234567890.021345abcdef6789`, der auf einem Workflow (`w-abcdef01234567890`) basiert, der einen Kopierschritt enthält.

```
{
    "type": "ExecutionStarted",
    "details": {
        "input": {
            "initialFileLocation": {
                "bucket": "DOC-EXAMPLE-BUCKET",
                "key": "mary/workflowSteps2.json",
                "versionId": "version-id",
                "etag": "etag-id"
            }
        }
    },
    "workflowId":"w-abcdef01234567890",
    "executionId":"execution-id",
    "transferDetails": {
        "serverId":"s-server-id",
```

```
        "username":"mary",
        "sessionId":"session-id"
    }
},
{
    "type":"StepStarted",
    "details": {
        "input": {
            "fileLocation": {
                "backingStore":"S3",
                "bucket":"DOC-EXAMPLE-BUCKET",
                "key":"mary/workflowSteps2.json",
                "versionId":"version-id",
                "etag":"etag-id"
            }
        },
        "stepType":"COPY",
        "stepName":"copyToShared"
    },
    "workflowId":"w-abcdef01234567890",
    "executionId":"execution-id",
    "transferDetails": {
        "serverId":"s-server-id",
        "username":"mary",
        "sessionId":"session-id"
    }
},
{
    "type":"StepCompleted",
    "details":{
        "output":{},
        "stepType":"COPY",
        "stepName":"copyToShared"
    },
    "workflowId":"w-abcdef01234567890",
    "executionId":"execution-id",
    "transferDetails":{
        "serverId":"server-id",
        "username":"mary",
        "sessionId":"session-id"
    }
},
{
    "type":"ExecutionCompleted",
```

```
    "details": {},
    "workflowId":"w-abcdef01234567890",
    "executionId":"execution-id",
    "transferDetails":{
        "serverId":"s-server-id",
        "username":"mary",
        "sessionId":"session-id"
    }
}
```

## Konfigurieren Sie die CloudWatch Protokollierungsrolle

Um den Zugriff festzulegen, erstellen Sie eine ressourcenbasierte IAM-Richtlinie und eine IAM-Rolle, die diese Zugriffsinformationen bereitstellt.

Um die CloudWatch Amazon-Protokollierung zu aktivieren, erstellen Sie zunächst eine IAM-Richtlinie, die die CloudWatch Protokollierung aktiviert. Anschließend erstellen Sie eine IAM-Rolle und fügen ihr die Richtlinie hinzu. Sie können dies tun, wenn Sie einen Server erstellen oder indem Sie einen vorhandenen Server bearbeiten. Weitere Informationen zu CloudWatch finden Sie unter Was ist Amazon CloudWatch? und Was ist Amazon CloudWatch Logs? im CloudWatch Amazon-Benutzerhandbuch.

Verwenden Sie die folgenden IAM-Beispielrichtlinien, um die CloudWatch Protokollierung zu ermöglichen.

Use a logging role

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogStream",
                "logs:DescribeLogStreams",
                "logs:CreateLogGroup",
                "logs:PutLogEvents"
            ],
            "Resource": "arn:aws:logs:*:*:log-group:/aws/transfer/*"
        }
    ]
```

```
}
```

Use structured logging

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogDelivery",
                "logs:GetLogDelivery",
                "logs:UpdateLogDelivery",
                "logs:DeleteLogDelivery",
                "logs:ListLogDeliveries",
                "logs:PutResourcePolicy",
                "logs:DescribeResourcePolicies",
                "logs:DescribeLogGroups"
            ],
            "Resource": "arn:aws:logs:region-id:AWS-Konto:log-group:/aws/transfer/*"
        }
    ]
}
```

Ersetzen Sie in der vorherigen Beispielrichtlinie für die die die **`Resource`** *Region-ID* und *AWS-Konto*durch Ihre Werte. Beispiel: **`"Resource": "arn:aws::logs:us-east-1:111122223333:log-group:/aws/transfer/*"`**

Anschließend erstellen Sie eine Rolle und fügen die von Ihnen erstellte CloudWatch Logs-Richtlinie hinzu.

So erstellen Sie eine IAM-Rolle und fügen dieser eine Richtlinie an

1. Wählen Sie im Navigationsbereich Roles (Rollen) und dann Create role (Rolle erstellen).

   Vergewissern Sie sich, dass auf der Seite Rolle erstellen der AWS Dienst ausgewählt ist.
2. Wählen Sie in der Service-Liste Transfer (Übertragung) und dann Next: Permissions (Weiter: Berechtigungen) aus. Dadurch wird eine Vertrauensbeziehung zwischen AWS Transfer Family und der IAM-Rolle hergestellt. Fügen Sie außerdem Schlüssel hinzu `aws:SourceAccount` und

geben Sie `aws:SourceArn` ihnen Bedingungen, um sich vor dem Problem mit dem verwirrten Stellvertreter zu schützen. Weitere Informationen finden Sie in der folgenden Dokumentation:

- Verfahren zum Aufbau einer Vertrauensbeziehung mit AWS Transfer Family: So stellen Sie eine Vertrauensbeziehung her
- Beschreibung des Problems mit dem verwirrten Stellvertreter: das Problem des verwirrten Stellvertreters

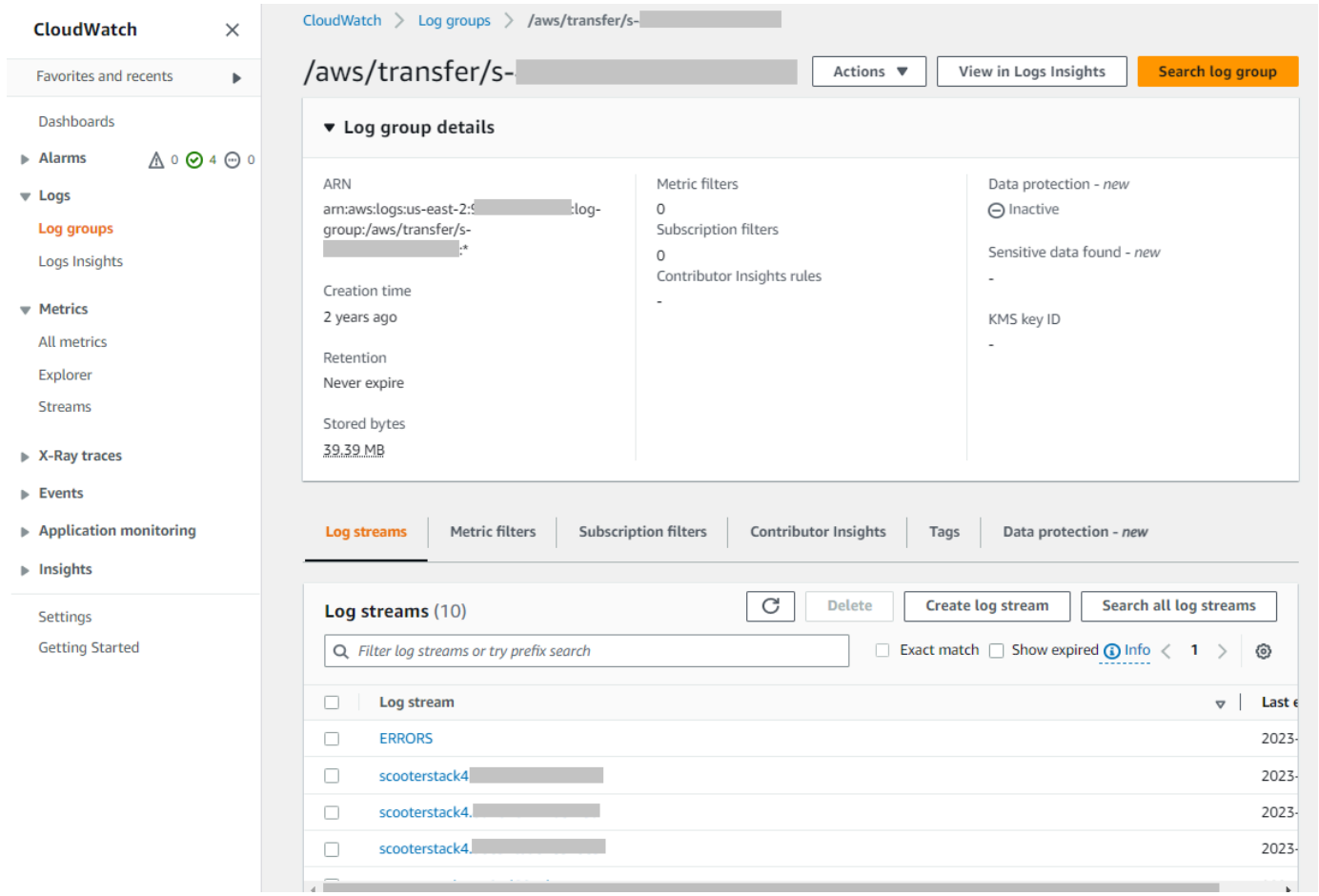
3. Suchen Sie im Abschnitt Zugriffsrichtlinien anhängen nach der CloudWatch Protokollrichtlinie, die Sie gerade erstellt haben, wählen Sie sie aus und klicken Sie dann auf Weiter: Tags.
4. (Optional) Geben Sie einen Schlüssel und einen Wert für ein Tag ein und wählen Sie Next: Review (Weiter: Prüfen) aus.
5. Geben Sie auf der Seite Review (Prüfen) einen Namen und eine Beschreibung für die neue Rolle ein und wählen Sie dann Create role (Rolle erstellen) aus.
6. Um die Protokolle anzuzeigen, wählen Sie die Server-ID aus, um die Serverkonfigurationsseite zu öffnen, und wählen Sie Protokolle anzeigen. Sie werden zur CloudWatch Konsole weitergeleitet, wo Sie Ihre Log-Streams sehen können.

Auf der CloudWatch Seite für Ihren Server finden Sie Aufzeichnungen zur Benutzerauthentifizierung (Erfolg und Misserfolg), zu Datenuploads (`PUT`Operationen) und zu Datendownloads (`GET`Operationen).

## Protokollstreams von Transfer Family anzeigen

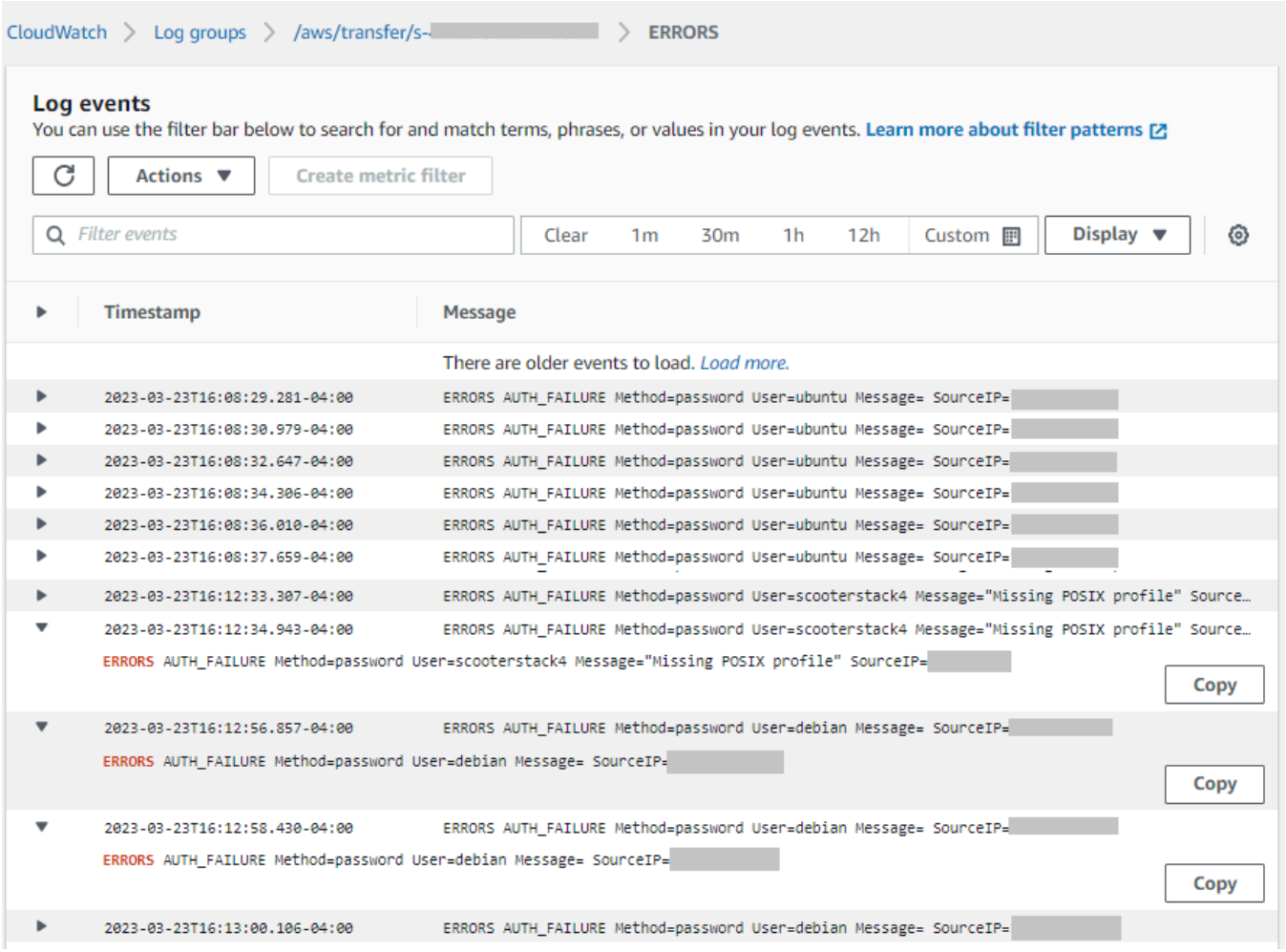
So zeigen Sie Ihre Transfer Family Family-Serverprotokolle an

1. Navigieren Sie zur Detailseite eines Servers.
2. Wählen Sie Protokolle anzeigen aus. Dadurch wird Amazon geöffnet CloudWatch.
3. Die Protokollgruppe für den ausgewählten Server wird angezeigt.

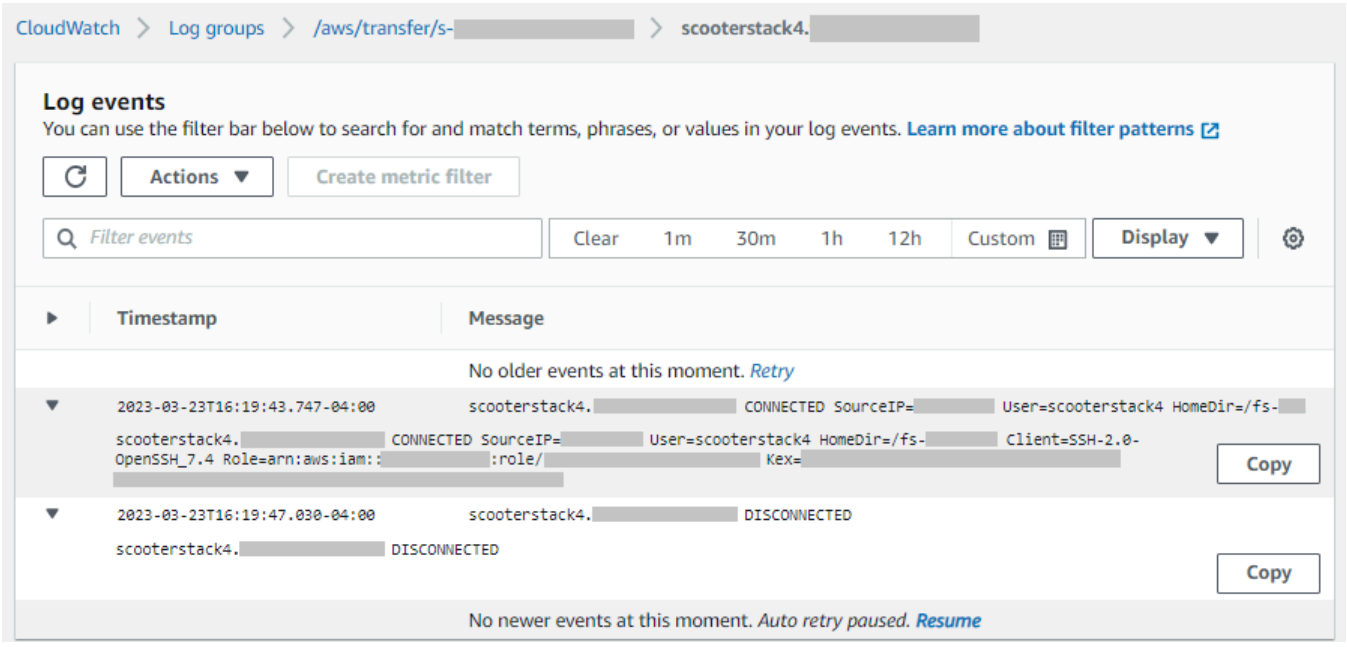



4. Sie können einen Protokollstream auswählen, um Details und einzelne Einträge für den Stream anzuzeigen.

   - Wenn es eine Liste für FEHLER gibt, können Sie diese auswählen, um Details zu den neuesten Fehlern auf dem Server anzuzeigen.

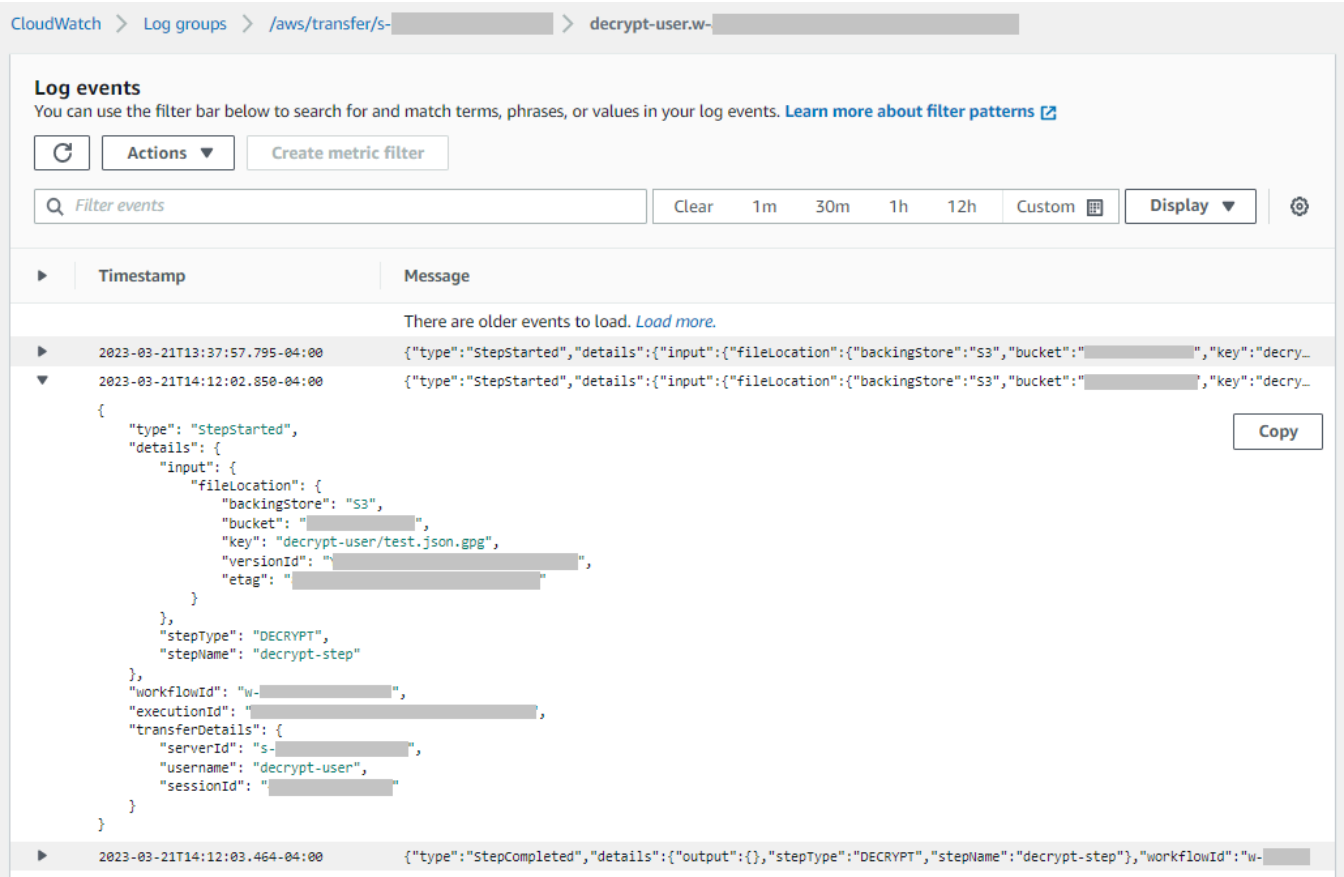



- Wählen Sie einen anderen Eintrag, um einen Beispiel-Logstream zu sehen.



- Wenn Ihrem Server ein verwalteter Workflow zugeordnet ist, können Sie die Protokolle der Workflow-Ausführungen anzeigen.

> **Note**
>
> Das Format für den Protokollstream für den Workflow ist*username*.*workflowId*.*uniqueStreamSuffix*. Beispielsweise könnte decrypt-user.w-a1111222233334444.aaaa1111bbbb2222 der Name eines Protokollstreams für Benutzer und Workflow sein. **`decrypt-user w-a1111222233334444`**




> **Note**
>
> Für jeden erweiterten Protokolleintrag können Sie den Eintrag in die Zwischenablage kopieren, indem Sie Kopieren wählen. Weitere Informationen zu CloudWatch Protokollen finden Sie unter Protokolldaten anzeigen.

## CloudWatch Amazon-Alarme erstellen

Das folgende Beispiel zeigt, wie CloudWatch Amazon-Alarme mithilfe der AWS Transfer Family Metrik erstellt `FilesIn` werden.

CDK

```
new cloudwatch.Metric({
  namespace: "AWS/Transfer",
  metricName: "FilesIn",
  dimensionsMap: { ServerId: "s-00000000000000000" },
  statistic: "Average",
  period: cdk.Duration.minutes(1),
}).createAlarm(this, "AWS/Transfer FilesIn", {
  threshold: 1000,
  evaluationPeriods: 10,
  datapointsToAlarm: 5,
  comparisonOperator:
 cloudwatch.ComparisonOperator.GREATER_THAN_OR_EQUAL_TO_THRESHOLD,
});
```

AWS CloudFormation

```
Type: AWS::CloudWatch::Alarm
Properties:
  Namespace: AWS/Transfer
  MetricName: FilesIn
  Dimensions:
    - Name: ServerId
      Value: s-00000000000000000
  Statistic: Average
  Period: 60
  Threshold: 1000
  EvaluationPeriods: 10
  DatapointsToAlarm: 5
  ComparisonOperator: GreaterThanOrEqualToThreshold
```

## Protokollierung Amazon S3 S3-API-Aufrufen in S3-Zugriffsprotokollen

Wenn Sie Amazon S3 S3-Zugriffsprotokolle verwenden, um S3-Anfragen zu identifizieren, die im Namen Ihrer Dateiübertragungsbenutzer gestellt wurden, `RoleSessionName` wird verwendet, um

anzuzeigen, welche IAM-Rolle für die Bearbeitung der Dateiübertragungen übernommen wurde. Außerdem werden zusätzliche Informationen wie der Benutzername, die Sitzungs-ID und die Server-ID angezeigt, die für die Übertragungen verwendet wurden. Das Format ist `[AWS:Role Unique Identifier]/username.sessionid@server-id` und ist im Feld Requester enthalten. Im Folgenden finden Sie beispielsweise den Inhalt eines Beispielfeldes „Requester“ aus einem S3-Zugriffsprotokoll für eine Datei, die in den S3-Bucket kopiert wurde.

`arn:aws:sts::AWS-Account-ID:assumed-role/`*`IamRoleName`*`/username.sessionid@server-id`

Im obigen Feld Requester wird die aufgerufene IAM-Rolle angezeigt. `IamRoleName` Weitere Informationen zu eindeutigen Identifikatoren der IAM-Rolle finden Sie unter Eindeutige Identifikatoren im Benutzerhandbuch.AWS Identity and Access Management

## Beispiele zur Begrenzung des Problems mit verwirrtem Stellvertreter

Das Problem des verwirrten Stellvertreters ist ein Sicherheitsproblem, bei dem eine Entität, die keine Berechtigung zur Durchführung einer Aktion hat, eine privilegiertere Entität zur Durchführung der Aktion zwingen kann. In der AWS Tat kann ein dienstübergreifendes Identitätswechsels zu einem Problem mit dem verwirrten Stellvertreter führen. Weitere Details finden Sie unter Serviceübergreifende Confused-Deputy-Prävention.

> **Note**
>
> Ersetzen Sie in den folgenden Beispielen alle *`Platzhalter für Benutzereingabe`* durch Ihre eigenen Informationen.
> In diesen Beispielen können Sie die ARN-Details für einen Workflow entfernen, wenn an Ihren Server keine Workflows angehängt sind.

Das folgende Beispiel für eine Protokollierungs-/Aufruf-Richtlinie ermöglicht es jedem Server (und Workflow) im Konto, die Rolle zu übernehmen.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowAllServersWithWorkflowAttached",
            "Effect": "Allow",
```

```
            "Principal": {
                "Service": "transfer.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "account-id"
                },
                "ArnLike": {
                   "aws:SourceArn": [
                     "arn:aws:transfer:region:account-id:server/*",
                     "arn:aws:transfer:region:account-id:workflow/*"
                   ]
                }
            }
        }
    ]
}
```

Das folgende Beispiel für eine Protokollierungs-/Aufruf-Richtlinie ermöglicht es einem bestimmten Server (und Workflow), die Rolle zu übernehmen.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowSpecificServerWithWorkflowAttached",
            "Effect": "Allow",
            "Principal": {
                "Service": "transfer.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "account-id"
                },
                "ArnEquals": {
                   "aws:SourceArn": [
                       "arn:aws:transfer:region:account-id:server/server-id",
                       "arn:aws:transfer:region:account-id:workflow/workflow-id"
                   ]
                }
            }
```

```
        }
    ]
}
```

# CloudWatch -Protokollstruktur für Transfer Family

In diesem Thema werden die Felder beschrieben, die in Transfer Family-Protokollen ausgefüllt werden: sowohl für strukturierte JSON-Protokolleinträge als auch für Legacy-Protokolleinträge.

Themen


- Strukturierte JSON-Protokolle für Transfer Family
- Legacy-Protokolle für Transfer Family


## Strukturierte JSON-Protokolle für Transfer Family

Die folgende Tabelle enthält Details zu Protokolleintragsfeldern für Transfer Family SFTP/FTP/FTPS-Aktionen im neuen strukturierten JSON-Protokollformat.

| Feld | Beschreibung | Beispieleintrag |
| --- | --- | --- |
| activity-type | The action by the user | OPEN \| CLOSE \| PARTIAL_CLOSE \| GETRENNT \| VERBUNDEN |
| bytes-in | Number of bytes uploaded by the user | 29238420042 |
| bytes-out | Number of bytes downloaded by the user | 23094032490328 |
| ciphers | Specifies the SSH cipher negotiated for the connection (available ciphers are listed in Kryptografische Algorithmen) | aes256-gcm@openssh.com |
| client | The user's client software | SSH-2.0-OpenSSH_7.4 |
| home-dir | The directory that the end user lands on when they connect | /user-home-bucket/test |

| Feld | Beschreibung | Beispieleintrag |
| --- | --- | --- |
| | to the endpoint if their home directory type is PATH: if they have a logical home directory, this value is always / | |
| kex | Specifies the negotiated SSH key exchange (KEX) for the connection (available KEX are listed in Kryptografische Algorithmen) | diffie-hellman-group14-sha256 |
| message | Provides more information related to the error | *<Zeichenfolge>* |
| method | The authentication method | publickey |
| mode | Specifies how a client opens a file | CREATE \| TRUNCATE \| WRITE |
| operation | The client operation on a file | OPEN \| CLOSE |
| path | Actual file path affected | /user-test-bucket/test-file-1.pdf |
| resource-arn | A system-assigned, unique identifier for a specific resource (for example, a server) | arn:aws:transfer:ap-northeast-1:12346789012:server/s-1234567890akeu2js2 |
| role | The IAM role of the user | arn:aws:iam::0293883675:role/testuser-role |
| session-id | A system-assigned, unique identifier for a single session | 9ca9a0e1cec6ad9d |
| source-ip | Client IP address | 18.323.0.129 |
| user | The end user's username | myname192 |

| Feld | Beschreibung | Beispieleintrag |
| --- | --- | --- |
| user-policy | The permissions specified for the end user: this field is populated if the user's policy is a session policy. | The JSON code for the session policy that is being used |

## Legacy-Protokolle für Transfer Family

Die folgende Tabelle enthält Details zu Protokolleinträgen für verschiedene Transfer Family-Aktionen.

> **Note**
>
> Diese Einträge haben nicht das neue strukturierte JSON-Protokollformat.

Die folgende Tabelle enthält Details zu Protokolleinträgen für verschiedene Transfer Family-Aktionen im neuen strukturierten JSON-Protokollformat.

| Aktion | Entsprechende Protokolle in Amazon CloudWatch Logs |
| --- | --- |
| Authentication failures (Authentifizierungsfehler) | ERRORS AUTH_FAILURE Method=publickey User=lhr Message="RSA SHA256:Lfz3R2nmLY4raK+b7Rb1rSvUIbAE+a+Hxg0c7l1JIZ0" SourceIP =3.8.172.211 |
| COPY/TAG/DELETE/DECRYPT-Workflow | {"type":"StepStarted","details":{"input":{"fileLocation ":{"backingStore ":"EFS ","filesystemId ":"fs-12345678","path":"lhr/regex.py"}},"stepType ":"TAG","stepName ":"successful_tag_step"},"workflowId ":"w-111aaaa222bb3","executionId ":"81234abcd-1234-efgh-5678-ijklmnopqr90","transferDetails "{"serverId ":"s-134aaa222222bb3"bb3", "exegid"efl"user sessionId1234567890 |

| Aktion | Entsprechende Protokolle in Amazon CloudWatch Logs |
| --- | --- |
| Benutzerdefinierter Schritt-Workflow | {"type":"CustomStepInvoked","details": {"output":{"token":"MzM4Mjg5YWUtYTEzMy 00YjIzLWI3OGMtYz U4OGI2ZjQyMzE5"}," stepType ":"CUSTOM","stepName ":"efs-s3 _copy_2"},"workflowId ":"w-9283e49d3297c 3f7","executionId ":"1234abcd-1234-efgh-5678-ijmnopqr90","transferDetails "{"serverId "":"s-zz1a1a23"sessionId1234567890 |
| Löschvorgänge | lhr.33a8fb495ffb383b DELETE Path=/bucket/user/123.jpg |
| Downloads | lhr.33a8fb495ffb383b OPEN Path=/bucket/user/123.jpg Mode=READ<br><br>llhr.33a8fb495ffb383b CLOSE Path=/bucket/user/123.jpg BytesOut=3618546 |
| Anmeldungen/Abmeldungen | user.914984e553bcdb6 CONNECTED SourceIP =1.22.111.222 User=lhr HomeDir=LOGICAL Client=SSH-2.0-OpenSSH _7.4 Role=arn:aws::iam::123456789012:role/sftp-s3-access<br><br>user.914984e553bcdb6 VERBUNDEN |
| Umbenennungen | lhr.33a8fb495ffb383b RENAME Path=/bucket/user/lambo.png NewPath=/bucket/user/ferrari.png |

| Aktion | Entsprechende Protokolle in Amazon CloudWatch Logs |
| --- | --- |
| Beispiel für ein Workflow-Fehlerprotokoll | {"type":"StepErrored","details":{"errorType ":"BAD_REQUEST","errorMessage ":"Cannot tag Efs file","stepType ":"TAG","stepName ":"successful_tag_step"},"workflowId ":"w-1234 abcd5678efghi","executionId ":"81234a bcd-1234-efgh-5678-ijklmnopqr90","transferDet ails ":{"serverId ":"s-1234abcd5678efghi", "username":"hr'sessionId" 123456789 0defrababab |
| Symlinks | lhr.eb49cf7b8651e6d5 CREATE_SYMLINK LinkPath=/fs-12345678/lhr/pqr.jpg TargetPat h=abc.jpg |
| Uploads | lhr.33a8fb495ffb383b OPEN Path=/bucket/user/ 123.jpg Mode=CREATE\|TRUNCATE\|WRITE<br><br>lhr.33a8fb495ffb383b CLOSE Path=/bucket/ user/123.jpg BytesIn=3618546 |

| Aktion | Entsprechende Protokolle in Amazon CloudWatch Logs |
| --- | --- |
| Workflows | {"type":"ExecutionStarted","details":{"input" :{initialFileLocation"":{"backingStore ":"EFS ","filesystemId ":"fs-12345678","path":"lhr/ regex.py"}}},"workflowId ":"w-111aaaaa2222b b3","executionId ":"1234abcd-1234-efgh-5678- ijklmnopqr90","transferDetails ":{"serverId ":"s- zz1111aaaa223", "username":"l"sessionId"123 4567890"111110x111111111111111222222 2222<br><br>{"type":"StepStarted","details":{"input":{"fi leLocation ":{"backingStore ":"EFS ","filesy stemId ":"fs-12345678","path":"/lhr/regex.p y"}},"stepType ":"CUSTOM","stepName ":"efs- s3_copy_2"},"workflowId ":"w-9283e49d33297 c3f7","executionId ":"1234abcd-1234-e fgh-5678-ijmnopqr90","transferDetails:{"serve rId:"8499ez"userid333333333"33r" Execution l"Executl sessionId1234567890" |

## Beispiel für CloudWatch Protokolleinträge

In diesem Thema werden Beispielprotokolleinträge vorgestellt.

Themen

- Beispiele für Protokolleinträge für Übertragungssitzungen
- Beispiel für Protokolleinträge für SFTP-Konnektoren
- Beispiel für Protokolleinträge für Fehler beim Schlüsselaustausch-Algorithmus

### Beispiele für Protokolleinträge für Übertragungssitzungen

In diesem Beispiel stellt ein SFTP-Benutzer eine Verbindung zu einem Transfer Family Family-Server her, lädt eine Datei hoch und trennt dann die Verbindung zur Sitzung.

Der folgende Protokolleintrag spiegelt einen SFTP-Benutzer wider, der eine Verbindung zu einem Transfer Family Family-Server herstellt.


```
{
   "role": "arn:aws:iam::500655546075:role/scooter-transfer-s3",
   "activity-type": "CONNECTED",
   "ciphers": "chacha20-poly1305@openssh.com,chacha20-poly1305@openssh.com",
   "client": "SSH-2.0-OpenSSH_7.4",
   "source-ip": "52.94.133.133",
   "resource-arn": "arn:aws:transfer:us-east-1:500655546075:server/
s-3fe215d89f074ed2a",
   "home-dir": "/scooter-test/log-me",
   "user": "log-me",
   "kex": "ecdh-sha2-nistp256",
   "session-id": "9ca9a0e1cec6ad9d"
}
```


Der folgende Protokolleintrag spiegelt den SFTP-Benutzer wider, der eine Datei in seinen Amazon S3 S3-Bucket hochlädt.


```
{
   "mode": "CREATE|TRUNCATE|WRITE",
   "path": "/scooter-test/log-me/config-file",
   "activity-type": "OPEN",
   "resource-arn": "arn:aws:transfer:us-east-1:500655546075:server/
s-3fe215d89f074ed2a",
   "session-id": "9ca9a0e1cec6ad9d"
}
```


Die folgenden Protokolleinträge geben an, dass der SFTP-Benutzer die Verbindung zu seiner SFTP-Sitzung beendet hat. Zuerst schließt der Client die Verbindung zum Bucket und dann trennt der Client die SFTP-Sitzung.


```
{
   "path": "/scooter-test/log-me/config-file",
   "activity-type": "CLOSE",
   "resource-arn": "arn:aws:transfer:us-east-1:500655546075:server/
s-3fe215d89f074ed2a",
   "bytes-in": "121",
   "session-id": "9ca9a0e1cec6ad9d"
}
```

```
{
   "activity-type": "DISCONNECTED",
   "resource-arn": "arn:aws:transfer:us-east-1:500655546075:server/
s-3fe215d89f074ed2a",
   "session-id": "9ca9a0e1cec6ad9d"
}
```

## Beispiel für Protokolleinträge für SFTP-Konnektoren

Dieser Abschnitt enthält Beispielprotokolle für eine erfolgreiche und eine erfolglose Übertragung. Protokolle werden in einer Protokollgruppe mit dem Namen generiert`/aws/transfer/`*`connector-id`*, wobei die *`Connector-ID`* die Kennung für Ihren SFTP-Connector ist.

> **Note**
>
> Protokolleinträge für SFTP-Konnektoren werden nur generiert, wenn Sie einen Befehl ausführen. `StartFileTransfer`

Dieser Protokolleintrag bezieht sich auf eine Übertragung, die erfolgreich abgeschlossen wurde.

```
{
    "operation": "RETRIEVE",
    "timestamp": "2023-10-25T16:33:27.373720Z",
    "connector-id": "connector-id",
    "transfer-id": "transfer-id",
    "file-transfer-id": "transfer-id/file-transfer-id",
    "url": "sftp://192.0.2.0",
    "file-path": "/remotebucket/remotefilepath",
    "status-code": "COMPLETED",
    "start-time": "2023-10-25T16:33:26.945481Z",
    "end-time": "2023-10-25T16:33:27.159823Z",
    "account-id": "480351544584",
    "connector-arn": "arn:aws:transfer:us-east-1:480351544584:connector/connector-id",
    "local-directory-path": "/connectors-localbucket"
    "bytes": 514
}
```

Dieser Protokolleintrag bezieht sich auf eine Übertragung, bei der das Zeitlimit überschritten wurde und die daher nicht erfolgreich abgeschlossen wurde.

```
{
    "operation": "RETRIEVE",
    "timestamp": "2023-10-25T22:33:47.625703Z",
    "connector-id": "connector-id",
    "transfer-id": "transfer-id",
    "file-transfer-id": "transfer-id/file-transfer-id",
    "url": "sftp://192.0.2.0",
    "file-path": "/remotebucket/remotefilepath",
    "status-code": "FAILED",
    "failure-code": "TIMEOUT_ERROR",
    "failure-message": "Transfer request timeout.",
    "account-id": "480351544584",
    "connector-arn": "arn:aws:transfer:us-east-1:480351544584:connector/connector-id",
    "local-directory-path": "/connectors-localbucket"
}
```

Dieser Protokolleintrag bezieht sich auf einen erfolgreichen SEND-Vorgang.

```
{
    "operation": "SEND",
    "timestamp": "2024-04-24T18:16:12.513207284Z",
    "connector-id": "connector-id",
    "transfer-id": "transfer-id",
    "file-transfer-id": "transfer-id/file-transfer-id",
    "url": "sftp://server-id.server.transfer.us-east-1.amazonaws.com",
    "file-path": "/DOC-EXAMPLE-BUCKET/my-test-folder/connector-metrics-us-
east-1-2024-01-02.csv",
    "status-code": "COMPLETED",
    "start-time": "2024-04-24T18:16:12.295235884Z",
    "end-time": "2024-04-24T18:16:12.461840732Z",
    "account-id": "255443218509",
    "connector-arn": "arn:aws:transfer:us-east-1:255443218509:connector/connector-id",
    "bytes": 275
}
```

Beschreibungen einiger Schlüsselfelder in den vorherigen Protokollbeispielen.

- `timestamp`gibt an, wann das Protokoll hinzugefügt wird CloudWatch. `start-time`und `end-time` entsprechen dem Zeitpunkt, zu dem der Konnektor eine Übertragung tatsächlich startet und beendet.

- `transfer-id`ist ein eindeutiger Bezeichner, der jeder `start-file-transfer` Anfrage zugewiesen wird. Wenn der Benutzer in einem einzigen `start-file-transfer` API-Aufruf mehrere Dateipfade übergibt, teilen sich alle Dateien dasselbe`transfer-id`.
- `file-transfer-id`ist ein eindeutiger Wert, der für jede übertragene Datei generiert wird. Beachten Sie, dass der erste Teil von derselbe `file-transfer-id` ist wie`transfer-id`.

## Beispiel für Protokolleinträge für Fehler beim Schlüsselaustausch-Algorithmus

Dieser Abschnitt enthält Beispielprotokolle, bei denen der Schlüsselaustauschalgorithmus (KEX) fehlgeschlagen ist. Dies sind Beispiele aus dem ERRORS-Protokollstream für strukturierte Protokolle.

Dieser Protokolleintrag ist ein Beispiel für einen Fehler beim Typ des Hostschlüssels.

```
{
    "activity-type": "KEX_FAILURE",
    "source-ip": "999.999.999.999",
    "resource-arn": "arn:aws:transfer:us-east-1:999999999999:server/
s-999999999999999999",
    "message": "no matching host key type found",
    "kex": "ecdsa-sha2-nistp256,ecdsa-sha2-nistp384,ecdsa-sha2-nistp521,ecdsa-sha2-
nistp256-cert-v01@openssh.com,ecdsa-sha2-nistp384-cert-v01@openssh.com,ecdsa-sha2-
nistp521-cert-v01@openssh.com,ssh-ed25519,ssh-rsa,ssh-dss"
}
```

Dieser Protokolleintrag ist ein Beispiel für eine KEX-Diskrepanz.

```
{
    "activity-type": "KEX_FAILURE",
    "source-ip": "999.999.999.999",
    "resource-arn": "arn:aws:transfer:us-east-1:999999999999:server/
s-999999999999999999",
    "message": "no matching key exchange method found",
    "kex": "diffie-hellman-group1-sha1,diffie-hellman-group14-sha1,diffie-hellman-
group14-sha256"
}
```

# CloudWatch Metriken für Transfer Family verwenden

> **Note**
>
> Sie können Metriken für Transfer Family auch in der Transfer Family Family-Konsole selbst abrufen. Details hierzu finden Sie unter Überwachung der Nutzung in der Konsole

Mithilfe von CloudWatch Metriken können Sie Informationen über Ihren Server abrufen. Eine Metrik steht für einen nach der Zeit geordneten Satz von Datenpunkten, die veröffentlicht werden. CloudWatch Wenn Sie Metriken verwenden, müssen Sie den Transfer Family Family-Namespace, den Metriknamen und die Dimension angeben. Weitere Informationen zu Metriken finden Sie unter Metriken im  CloudWatch Amazon-Benutzerhandbuch.

In der folgenden Tabelle werden die CloudWatch Metriken für Transfer Family beschrieben.

| Namespace | Metrik | Beschreibung |
|---|---|---|
| `AWS/Transfer` | `BytesIn` | Die Gesamtzahl der auf den Server übertragenen Byte.<br><br>Einheiten: Anzahl<br><br>Dauer: 5 Minuten |
| | `BytesOut` | Die Gesamtzahl der vom Server übertragenen Byte.<br><br>Einheit: Anzahl<br><br>Dauer: 5 Minuten |
| | `FilesIn` | Die Gesamtzahl der auf den Server übertragenen Dateien.<br><br>Bei Servern, die das AS2-Protokoll verwenden, stellt diese Metrik die Anzahl der empfangenen Nachrichten dar.<br><br>Einheiten: Anzahl<br><br>Dauer: 5 Minuten |

| Namespace | Metrik | Beschreibung |
| --- | --- | --- |
| | `FilesOut` | Die Gesamtzahl der vom Server übertragenen Dateien.<br><br>Einheiten: Anzahl<br><br>Dauer: 5 Minuten |
| | `InboundMessage` | Die Gesamtzahl der erfolgreich von einem Handelspartner empfangenen AS2-Nachrichten.<br><br>Einheiten: Anzahl<br><br>Dauer: 5 Minuten |
| | `InboundFailedMessage` | Die Gesamtzahl der AS2-Nachrichten, die von einem Handelspartner erfolglos empfangen wurden. Das heißt, ein Handelspartner hat eine Nachricht gesendet, aber der Transfer Family Family-Server konnte sie nicht erfolgreich verarbeiten.<br><br>Einheiten: Anzahl<br><br>Dauer: 5 Minuten |
| | `OnUploadExecutionsStarted` | Die Gesamtzahl der auf dem Server gestarteten Workflow-Ausführungen.<br><br>Einheiten: Anzahl<br><br>Zeitraum: 1 Minute |
| | `OnUploadExecutionsSuccess` | Die Gesamtzahl der erfolgreichen Workflow-Ausführungen auf dem Server.<br><br>Einheiten: Anzahl<br><br>Zeitraum: 1 Minute |

| Namespace | Metrik | Beschreibung |
| --- | --- | --- |
| | `OnUploadExecutionsFailed` | Die Gesamtzahl der erfolglosen Workflow-Ausführungen auf dem Server.<br><br>Einheiten: Anzahl<br><br>Zeitraum: 1 Minute |

### Familiendimensionen übertragen

Eine Dimension ist ein Name-Wert-Paar, das zur Identifizierung einer Metrik beiträgt. Weitere Informationen zu Abmessungen finden Sie unter Abmessungen im  CloudWatch Amazon-Benutzerhandbuch.

In der folgenden Tabelle wird die CloudWatch Dimension für Transfer Family beschrieben.

| Dimension | Beschreibung |
| --- | --- |
| `ServerId` | Die eindeutige ID des Servers. |

## Wird AWS-Benutzerbenachrichtigungen mit verwendet AWS Transfer Family

Um über AWS Transfer Family Ereignisse informiert AWS-Benutzerbenachrichtigungenzu werden, können Sie verschiedene Lieferkanäle einrichten. Wenn ein Ereignis einer von Ihnen angegebenen Regel entspricht, erhalten Sie eine Benachrichtigung.

Sie können Benachrichtigungen für Ereignisse über mehrere Kanäle erhalten, einschließlich E-Mail-, AWS Chatbot-Chat- oder AWS Console Mobile Application-Push-Benachrichtigungen. Sie können Benachrichtigungen auch im Benachrichtigungscenter der Konsole sehen. Benutzerbenachrichtigungen unterstützt die Aggregation, wodurch die Anzahl der Benachrichtigungen, die Sie bei bestimmten Ereignissen erhalten, reduziert werden kann.

Weitere Informationen finden Sie in den Blogbeiträgen Anpassen von Benachrichtigungen zur Dateizustellung mithilfe AWS Transfer Family verwalteter Workflows und Was ist AWS-Benutzerbenachrichtigungen? im AWS-Benutzerbenachrichtigungen Benutzerhandbuch.

# Verwenden von Abfragen zum Filtern von Protokolleinträgen

Sie können CloudWatch Abfragen verwenden, um Protokolleinträge für Transfer Family zu filtern und zu identifizieren. Dieser Abschnitt enthält einige Beispiele.

1. Melden Sie sich bei der an AWS Management Console und öffnen Sie die CloudWatch Konsole unter https://console.aws.amazon.com/cloudwatch/.
2. Sie können Abfragen oder Regeln erstellen.
   - Um eine Logs Insights-Abfrage zu erstellen, wählen Sie im linken Navigationsbereich Logs Insights aus und geben Sie dann die Details für Ihre Abfrage ein.
   - Um eine Contributor Insights-Regel zu erstellen, wählen Sie im linken Navigationsbereich Insights > Contributor Insights aus und geben Sie dann die Details für Ihre Regel ein.
3. Führen Sie die Abfrage oder Regel aus, die Sie erstellt haben.

Sehen Sie sich die häufigsten Ursachen für Authentifizierungsfehler an

In Ihren strukturierten Protokollen sieht ein Protokolleintrag für Authentifizierungsfehler in etwa wie folgt aus:

```
{
  "method":"password",
  "activity-type":"AUTH_FAILURE",
  "source-ip":"999.999.999.999",
  "resource-arn":"arn:aws:transfer:us-east-1:999999999999:server/s-0123456789abcdef",
  "message":"Invalid user name or password",
  "user":"exampleUser"
}
```

Führen Sie die folgende Abfrage aus, um die Hauptverursacher von Authentifizierungsfehlern anzuzeigen.

```
filter @logStream = 'ERRORS'
| filter `activity-type` = 'AUTH_FAILURE'
| stats count() as AuthFailures by user, method
| sort by AuthFailures desc
| limit 10
```

Anstatt CloudWatch Logs Insights zu verwenden, können Sie eine CloudWatch Contributors Insights-Regel erstellen, um Authentifizierungsfehler anzuzeigen. Erstellen Sie eine Regel, die der folgenden ähnelt.

```
{
    "AggregateOn": "Count",
    "Contribution": {
        "Filters": [
            {
                "Match": "$.activity-type",
                "In": [
                    "AUTH_FAILURE"
                ]
            }
        ],
        "Keys": [
            "$.user"
        ]
    },
    "LogFormat": "JSON",
    "Schema": {
        "Name": "CloudWatchLogRule",
        "Version": 1
    },
    "LogGroupARNs": [
        "arn:aws:logs:us-east-1:999999999999:log-group:/customer/structured_logs"
    ]
}
```

Protokolleinträge anzeigen, in denen eine Datei geöffnet wurde

In Ihren strukturierten Protokollen sieht ein Protokolleintrag zum Lesen einer Datei etwa wie folgt aus:

```
{
  "mode":"READ",
  "path":"/fs-0df669c89d9bf7f45/avtester/example",
  "activity-type":"OPEN",
  "resource-arn":"arn:aws:transfer:us-east-1:999999999999:server/s-0123456789abcdef",
  "session-id":"0049cd844c7536c06a89"
}
```

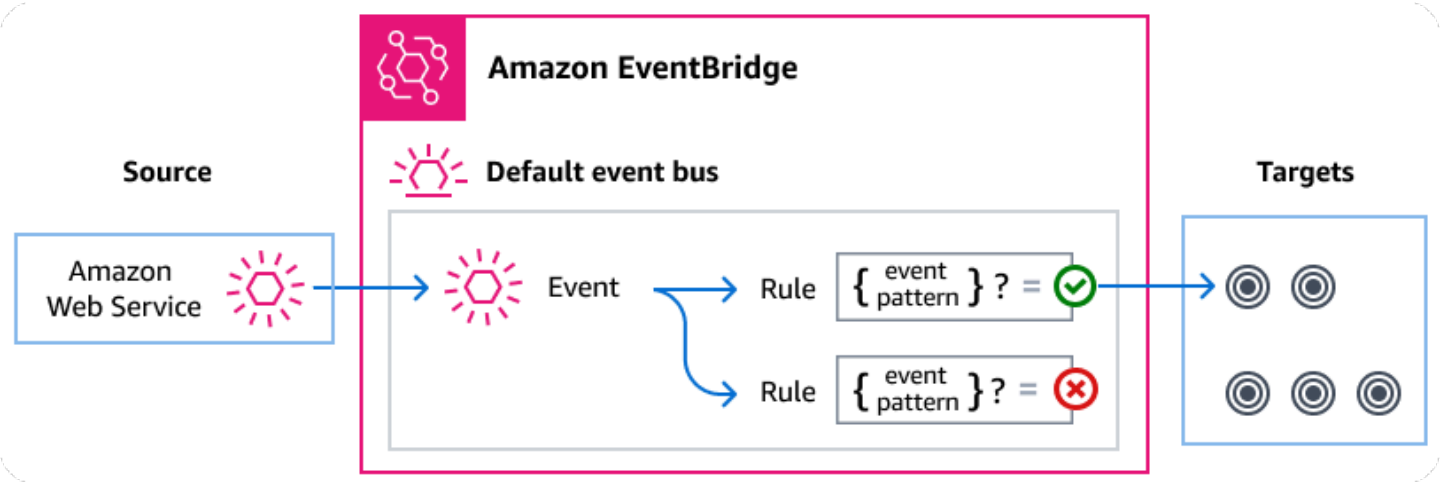
Führen Sie die folgende Abfrage aus, um Protokolleinträge anzuzeigen, die darauf hinweisen, dass eine Datei geöffnet wurde.

```
filter `activity-type` = 'OPEN'
| display @timestamp, @logStream, `session-id`, mode, path
```

# Transfer Family Ereignisse verwalten mit Amazon EventBridge

Amazon EventBridge ist ein serverloser Dienst, der Ereignisse verwendet, um Anwendungskomponenten miteinander zu verbinden. Dies kann Ihnen die Erstellung skalierbarer ereignisgesteuerter Anwendungen erleichtern. Bei der ereignisgesteuerten Architektur werden lose gekoppelte Softwaresysteme erstellt, die zusammenarbeiten, indem sie Ereignisse auslösen und darauf reagieren. Ereignisse stellen eine Veränderung in einer Ressource oder Umgebung dar.

Transfer Family Generiert wie bei vielen AWS Diensten Ereignisse und sendet sie an den EventBridge Standard-Event-Bus. Beachten Sie, dass der Standard-Event-Bus automatisch in jedem AWS Konto bereitgestellt wird. Ein Event Bus ist ein Router, der Ereignisse empfängt und sie an null oder mehr Ziele weiterleitet. Sie geben Regeln für den Event-Bus an, der Ereignisse auswertet, sobald sie eintreffen. Jede Regel prüft, ob ein Ereignis mit dem Ereignismuster der Regel übereinstimmt. Wenn das Ereignis übereinstimmt, sendet der Event-Bus das Ereignis an ein oder mehrere angegebene Ziele.




Themen

- Transfer Family Ereignisse
- Senden von Transfer Family Ereignissen mithilfe von Regeln EventBridge
- Amazon EventBridge Berechtigungen
- Zusätzliche EventBridge Ressourcen
- Transfer Family Referenz mit Einzelheiten zu Ereignissen

# Transfer Family Ereignisse

Transfer Family sendet Ereignisse automatisch an den EventBridge Standard-Event-Bus. Sie können Regeln auf dem Event-Bus erstellen, wobei jede Regel ein Ereignismuster und ein oder mehrere Ziele enthält. Ereignisse, die dem Ereignismuster einer Regel entsprechen, werden nach bestem Wissen und Gewissen an die angegebenen Ziele übermittelt. Es kann jedoch sein, dass einige Ereignisse nicht in der richtigen Reihenfolge zugestellt werden.

Die folgenden Ereignisse werden von generiert Transfer Family. Weitere Informationen finden Sie im Amazon EventBridge Benutzerhandbuch unter EventBridge Ereignisse.

## SFTP-, FTPS- und FTP-Serverereignisse

| Art der Einzelheiten des Ereignisses | Beschreibung |
| --- | --- |
| Der Download des FTP-Dateiservers ist abgeschlossen | Eine Datei wurde erfolgreich für das FTP-Protokoll heruntergeladen. |
| Der Download des FTP-Dateiservers ist fehlgeschlagen | Ein Versuch, eine Datei für das FTP-Protokoll herunterzuladen, ist fehlgeschlagen. |
| Der Upload des FTP-Dateiservers ist abgeschlossen | Eine Datei wurde erfolgreich für das FTP-Protokoll hochgeladen. |
| Der Upload des FTP-Dateiservers ist fehlgeschlagen | Ein Versuch, eine Datei hochzuladen, ist für das FTP-Protokoll fehlgeschlagen. |
| Der Download des FTPS-Dateiservers ist abgeschlossen | Eine Datei wurde erfolgreich für das FTPS-Protokoll heruntergeladen. |
| Der Download des FTPS-Dateiservers ist fehlgeschlagen | Ein Versuch, eine Datei für das FTPS-Protokoll herunterzuladen, ist fehlgeschlagen. |
| Der Upload des FTPS-Dateiservers ist abgeschlossen | Eine Datei wurde erfolgreich für das FTPS-Protokoll hochgeladen. |
| Der Upload des FTPS-Dateiservers ist fehlgeschlagen | Ein Versuch, eine Datei hochzuladen, ist für das FTPS-Protokoll fehlgeschlagen. |

| Art der Einzelheiten des Ereignisses | Beschreibung |
| --- | --- |
| Der Download der SFTP-Serverdatei ist abgeschlossen | Eine Datei wurde erfolgreich für das SFTP-Protokoll heruntergeladen. |
| Das Herunterladen der SFTP-Serverdatei ist fehlgeschlagen | Ein Versuch, eine Datei für das SFTP-Protokoll herunterzuladen, ist fehlgeschlagen. |
| Das Hochladen der SFTP-Serverdatei ist abgeschlossen | Eine Datei wurde erfolgreich für das SFTP-Protokoll hochgeladen. |
| Das Hochladen der SFTP-Serverdatei ist fehlgeschlagen | Ein Versuch, eine Datei hochzuladen, ist für das SFTP-Protokoll fehlgeschlagen. |

## Ereignisse des SFTP-Connectors

| Art der Einzelheiten des Ereignisses | Beschreibung |
| --- | --- |
| Das Senden der SFTP-Connector-Datei ist abgeschlossen | Eine Dateiübertragung von einem Connector zu einem Remote-SFTP-Server wurde erfolgreich abgeschlossen. |
| Das Senden der SFTP-Connector-Datei ist fehlgeschlagen | Eine Dateiübertragung von einem Connector zu einem Remote-SFTP-Server ist fehlgeschlagen. |
| Der Abruf der SFTP-Connector-Datei ist abgeschlossen | Eine Dateiübertragung von einem Remote-SFTP-Server zu einem Connector wurde erfolgreich abgeschlossen. |
| Fehler beim Abrufen der SFTP-Connector-Datei | Eine Dateiübertragung von einem Remote-SFTP-Server zu einem Connector ist fehlgeschlagen. |
| Die Verzeichnisliste des SFTP-Connectors ist abgeschlossen | Ein Aufruf zum Auflisten des Startdateiverzeichnisses, der erfolgreich abgeschlossen wurde. |

| Art der Einzelheiten des Ereignisses | Beschreibung |
| --- | --- |
| Die Verzeichnisliste des SFTP-Connectors ist fehlgeschlagen | Eine Verzeichnisliste für die Startdatei, die fehlgeschlagen ist. |

## A2S-Ereignisse

| Art der Einzelheiten des Ereignisses | Beschreibung |
| --- | --- |
| Empfang der AS2-Nutzlast abgeschlossen | Die Payload für eine AS2-Nachricht wurde empfangen. |
| Der Empfang der AS2-Nutzdaten ist fehlgeschlagen | Die Payload für eine AS2-Nachricht wurde nicht empfangen. |
| Der Versand der AS2-Nutzlast ist abgeschlossen | Die Payload für eine AS2-Nachricht wurde erfolgreich gesendet. |
| Das Senden der AS2-Nutzlast ist fehlgeschlagen | Die Payload für eine AS2-Nachricht konnte nicht gesendet werden. |
| AS2 MDN-Empfang abgeschlossen | Die Benachrichtigung über die Entsorgung von Nachrichten für eine AS2-Nachricht wurde empfangen. |
| AS2 MDN-Empfang ist fehlgeschlagen | Die Benachrichtigung über die Nachrichtenablage für eine AS2-Nachricht wurde nicht empfangen. |
| AS2 MDN Send abgeschlossen | Die Benachrichtigung über die Entsorgung von Nachrichten für eine AS2-Nachricht wurde erfolgreich gesendet. |
| AS2 MDN konnte nicht gesendet werden | Die Benachrichtigung über die Nachrichtenablage für eine AS2-Nachricht konnte nicht gesendet werden. |

# Senden von Transfer Family Ereignissen mithilfe von Regeln EventBridge

Wenn Sie möchten, dass der EventBridge Standardereignisbus Transfer Family Ereignisse an ein Ziel sendet, müssen Sie eine Regel erstellen, die ein Ereignismuster enthält, das den Daten in Ihren gewünschten Transfer Family Ereignissen entspricht.

Sie können eine Regel erstellen, indem Sie die folgenden allgemeinen Schritte ausführen:

1. Erstellen Sie ein Ereignismuster für die Regel, das Folgendes festlegt:
   - Transfer Family ist die Quelle der Ereignisse, die von der Regel ausgewertet werden.
   - (Optional) Alle anderen Ereignisdaten, mit denen sie abgeglichen werden soll.

   Weitere Informationen finden Sie unter ???.
2. (Optional) Erstellen Sie einen Eingangstransformator, der die Daten aus dem Ereignis anpasst, bevor die Informationen an das Ziel der Regel EventBridge gesendet werden.

   Weitere Informationen finden Sie unter Eingabetransformation im EventBridge Benutzerhandbuch.
3. Geben Sie die Ziele an, an die Sie Ereignisse senden EventBridge möchten, die dem Ereignismuster entsprechen.

   Ziele können andere AWS Dienste, SaaS-Anwendungen (Software as a Service), API-Ziele oder andere benutzerdefinierte Endpunkte sein. Weitere Informationen finden Sie unter Ziele im Benutzerhandbuch für EventBridge .

Umfassende Anweisungen zum Erstellen von Event-Bus-Regeln finden Sie im EventBridge Benutzerhandbuch unter Erstellen von Regeln, die auf Ereignisse reagieren.

## Ereignismuster für Transfer Family Ereignisse erstellen

Wenn ein Ereignis an den Standardereignisbus Transfer Family übermittelt, EventBridge verwendet das für jede Regel definierte Ereignismuster, um zu bestimmen, ob das Ereignis an die Ziele der Regel gesendet werden soll. Ein Ereignismuster entspricht den Daten in den gewünschten Transfer Family Ereignissen. Jedes Ereignismuster ist ein JSON-Objekt, das Folgendes enthält:

- Ein `source`-Attribut, das den Service identifiziert, der das Ereignis sendet. Für Transfer Family Ereignisse ist die Quelle`aws.transfer`.

- (Optional) Ein `detail-type` Attribut, das ein Array der zuzuordnenden Ereignistypen enthält.
- (Optional) Ein `detail` Attribut, das alle anderen Ereignisdaten enthält, für die ein Abgleich erforderlich ist.

Das folgende Ereignismuster entspricht beispielsweise allen Ereignissen von Transfer Family:

```
{
  "source": ["aws.transfer"]
}
```

Das folgende Beispiel für ein Ereignismuster entspricht allen Ereignissen des SFTP-Connectors:

```
{
  "source": ["aws.transfer"],
  "detail-type": ["SFTP Connector File Send Completed", "SFTP Connector File Retrieve
 Completed",
                  "SFTP Connector File Retrieve Failed", "SFTP Connector File Send
 Failed"]
}
```

Das folgende Beispiel für ein Ereignismuster entspricht allen fehlgeschlagenen Transfer Family Family-Ereignissen:

```
{
  "source": ["aws.transfer"],
  "detail-type": [{"wildcard", "*Failed"}]
}
```

Das folgende Beispiel für ein Ereignismuster entspricht erfolgreichen SFTP-Downloads für den *Benutzernamen:*

```
{
  "source": ["aws.transfer"],
  "detail-type": ["SFTP Server File Download Completed"],
  "detail": {
    "username": [username]
  }
}
```

Weitere Informationen zum Schreiben von Ereignismustern finden Sie unter Ereignismuster im EventBridge Benutzerhandbuch.

### Testen von Ereignismustern für Transfer Family Ereignisse in EventBridge

Sie können die EventBridge Sandbox verwenden, um schnell ein Ereignismuster zu definieren und zu testen, ohne den umfassenderen Prozess der Erstellung oder Bearbeitung einer Regel abschließen zu müssen. Mithilfe der Sandbox können Sie ein Ereignismuster definieren und anhand eines Beispielereignisses überprüfen, ob das Muster den gewünschten Ereignissen entspricht. EventBridge bietet Ihnen die Möglichkeit, eine neue Regel zu erstellen, indem Sie dieses Ereignismuster direkt aus der Sandbox verwenden.

Weitere Informationen finden Sie unter Testen eines Ereignismusters mithilfe der EventBridge Sandbox im EventBridge Benutzerhandbuch.

## Amazon EventBridge Berechtigungen

Transfer Family benötigt keine zusätzlichen Berechtigungen für die Übermittlung von Ereignissen an Amazon EventBridge.

Für die von Ihnen angegebenen Ziele sind möglicherweise bestimmte Berechtigungen oder Konfigurationen erforderlich. Weitere Informationen zur Verwendung bestimmter Dienste für Amazon EventBridge Ziele finden Sie im Amazon EventBridge Benutzerhandbuch unter Ziele.

## Zusätzliche EventBridge Ressourcen

Weitere Informationen zur Verarbeitung und Verwaltung von Ereignissen finden Sie EventBridge in den folgenden Themen im Amazon EventBridge Benutzerhandbuch.

- Ausführliche Informationen zur Funktionsweise von Eventbussen finden Sie unter Amazon EventBridge Event-Bus.
- Informationen zur Veranstaltungsstruktur finden Sie unter Ereignisse.
- Informationen zur Erstellung von Ereignismustern für EventBridge den Abgleich von Ereignissen mit Regeln finden Sie unter Ereignismuster.
- Informationen zum Erstellen von Regeln, mit denen angegeben wird, welche Ereignisse EventBridge verarbeitet werden, finden Sie unter Regeln.
- Informationen darüber, wie Sie angeben, an welche Dienste oder andere Ziele übereinstimmende Ereignisse EventBridge gesendet werden, finden Sie unter Ziele.

# Transfer Family Referenz mit Einzelheiten zu Ereignissen

Alle Ereignisse von AWS Diensten haben einen gemeinsamen Satz von Feldern, die Metadaten zu dem Ereignis enthalten. Zu diesen Metadaten können der AWS Dienst gehören, der die Quelle des Ereignisses ist, die Uhrzeit, zu der das Ereignis generiert wurde, das Konto und die Region, in der das Ereignis stattgefunden hat, und andere. Definitionen dieser allgemeinen Felder finden Sie unter Referenz zur Ereignisstruktur im Amazon EventBridge Benutzerhandbuch.

Darüber hinaus weist jedes Ereignis ein `detail`-Feld auf, das spezifische Daten für das betreffende Ereignis enthält. In der folgenden Referenz werden die Detailfelder für die verschiedenen Transfer Family Ereignisse definiert.

Beachten Sie EventBridge bei der Auswahl und Verwaltung von Transfer Family Ereignissen Folgendes:

- Das `source` Feld für alle Ereignisse von Transfer Family ist auf gesetzt`aws.transfer`.
- Das Feld `detail-type` gibt den Ereignistyp an.

  z. B. `FTP File Server Download Completed`.
- Das Feld `detail` enthält die Daten, die für das betreffende Ereignis spezifisch sind.

Informationen zur Erstellung von Ereignismustern, die es Regeln ermöglichen, Transfer Family Ereignissen zuzuordnen, finden Sie im Amazon EventBridge Benutzerhandbuch unter Ereignismuster.

Weitere Informationen zu Ereignissen und deren EventBridge Verarbeitung finden Sie im Amazon EventBridge Benutzerhandbuch unter Amazon EventBridge Ereignisse.

Themen


- SFTP-, FTPS- und FTP-Serverereignisse
- Ereignisse des SFTP-Connectors
- AS2-Ereignisse


## SFTP-, FTPS- und FTP-Serverereignisse

Im Folgenden finden Sie die Detailfelder für SFTP-, FTPS- und FTP-Serverereignisse:

- Der Download des FTP-Dateiservers ist abgeschlossen
- Der Download des FTP-Dateiservers ist fehlgeschlagen
- Upload des FTP-Dateiservers abgeschlossen
- Der Upload des FTP-Dateiservers ist fehlgeschlagen
- Der Download des FTPS-Dateiservers wurde abgeschlossen
- Der Download des FTPS-Dateiservers ist fehlgeschlagen
- Upload des FTPS-Dateiservers abgeschlossen
- Der Upload des FTPS-Dateiservers ist fehlgeschlagen
- Das Herunterladen der SFTP-Serverdatei ist abgeschlossen
- Das Herunterladen der SFTP-Serverdatei ist fehlgeschlagen
- Das Hochladen der SFTP-Serverdatei ist abgeschlossen
- Das Hochladen der SFTP-Serverdatei ist fehlgeschlagen

Die `detail-type` Felder `source` und sind unten aufgeführt, da sie spezifische Werte für Transfer Family Ereignisse enthalten. Definitionen der anderen Metadatenfelder, die in allen Ereignissen enthalten sind, finden Sie unter Referenz zur Ereignisstruktur im Amazon EventBridge Benutzerhandbuch.

```
{
  . . .,
  "detail-type": "string",
  "source": "aws.transfer",
  . . .,
  "detail": {
    "failure-code" : "string",
    "status-code" : "string",
    "protocol" : "string",
    "bytes" : "number",
    "client-ip" : "string",
    "failure-message" : "string",
    "end-timestamp" : "string",
    "etag" : "string",
    "file-path" : "string",
    "server-id" : "string",
    "username" : "string",
    "session-id" : "string",
    "start-timestamp" : "string"
```

```
  }
}
```

`detail-type`

Identifiziert den Ereignistyp.

Für dieses Ereignis ist der Wert einer der zuvor aufgeführten SFTP-, FTPS- oder FTP-Serverereignisnamen.

`source`

Identifiziert den Service, aus dem das Ereignis stammt. Für Transfer Family Family-Veranstaltungen ist dieser Wert`aws.transfer`.

`detail`

Ein JSON-Objekt, das Informationen zum Ereignis enthält. Der Service, der das Ereignis generiert, bestimmt den Inhalt dieses Feldes.

Für dieses Ereignis umfassen die Daten Folgendes:

`failure-code`

Kategorie, warum die Übertragung fehlgeschlagen ist. Werte: `PARTIAL_UPLOAD | PARTIAL_DOWNLOAD | UNKNOWN_ERROR`

`status-code`

Ob die Übertragung erfolgreich ist. Werte:`COMPLETED | FAILED`.

`protocol`

Das für die Übertragung verwendete Protokoll. Werte: `SFTP | FTPS | FTP`

`bytes`

Die Anzahl der übertragenen Bytes.

`client-ip`

Die IP-Adresse des an der Übertragung beteiligten Clients

`failure-message`

Bei fehlgeschlagenen Übertragungen die Details, warum die Übertragung fehlgeschlagen ist.

`end-timestamp`

Bei erfolgreichen Übertragungen der Zeitstempel, wann die Verarbeitung der Datei abgeschlossen ist.

`etag`

Das Entity-Tag (wird nur für Amazon S3 S3-Dateien verwendet).

`file-path`

Der Pfad zu der Datei, die übertragen wird.

`server-id`

Die eindeutige ID für den Transfer Family Family-Server.

`username`

Der Benutzer, der die Übertragung durchführt.

`session-id`

Die eindeutige Kennung für die Übertragungssitzung.

`start-timestamp`

Bei erfolgreichen Übertragungen der Zeitstempel für den Beginn der Dateiverarbeitung.

Example Beispielereignis: Fehler beim Herunterladen der SFTP-Serverdatei

Das folgende Beispiel zeigt ein Ereignis, bei dem ein Download auf einem SFTP-Server fehlgeschlagen Amazon EFS ist (wird der Speicher verwendet).

```
{
    "version": "0",
    "id": "event-ID",
    "detail-type": "SFTP Server File Download Failed",
    "source": "aws.transfer",
    "account": "958412138249",
    "time": "2024-01-29T17:20:27Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:transfer:us-east-1:958412138249:server/s-1234abcd5678efghi"
    ],
```

```
    "detail": {
        "failure-code": "PARTIAL_DOWNLOAD",
        "status-code": "FAILED",
        "protocol": "SFTP",
        "bytes": 4100,
        "client-ip": "IP-address",
        "failure-message": "File was partially downloaded.",
        "end-timestamp": "2024-01-29T17:20:27.749749117Z",
        "file-path": "/fs-1234abcd5678efghi/user0/test-file",
        "server-id": "s-1234abcd5678efghi",
        "username": "test",
        "session-id": "session-ID",
        "start-timestamp": "2024-01-29T17:20:16.706282454Z"
    }
}
```

Example Beispielereignis „FTP-Dateiserver-Upload abgeschlossen“

Das folgende Beispiel zeigt ein Ereignis, bei dem ein Upload auf einem FTP-Server erfolgreich abgeschlossen wurde (Amazon S3 wird der Speicher verwendet).

```
{
    "version": "0",
    "id": "event-ID",
    "detail-type": "FTP Server File Upload Completed",
    "source": "aws.transfer",
    "account": "958412138249",
    "time": "2024-01-29T16:31:43Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:transfer:us-east-1:958412138249:server/s-1111aaaa2222bbbb3"
    ],
    "detail": {
        "status-code": "COMPLETED",
        "protocol": "FTP",
        "bytes": 1048576,
        "client-ip": "10.0.0.141",
        "end-timestamp": "2024-01-29T16:31:43.311866408Z",
        "etag": "b6d81b360a5672d80c27430f39153e2c",
        "file-path": "/DOC-EXAMPLE-BUCKET/test/1mb_file",
        "server-id": "s-1111aaaa2222bbbb3",
        "username": "test",
        "session-id": "event-ID",
```

```
        "start-timestamp": "2024-01-29T16:31:42.462088327Z"
    }
}
```

## Ereignisse des SFTP-Connectors

Im Folgenden sind die Detailfelder für SFTP-Connector-Ereignisse aufgeführt:

- Das Senden der SFTP-Connector-Datei ist abgeschlossen
- Das Senden der SFTP-Connector-Datei ist fehlgeschlagen
- Der Abruf der SFTP-Connector-Datei ist abgeschlossen
- Fehler beim Abrufen der SFTP-Connector-Datei
- Die Verzeichnisliste für den SFTP-Connector ist abgeschlossen
- Die Verzeichnisliste für den SFTP-Connector ist fehlgeschlagen

Die `detail-type` Felder `source` und sind unten aufgeführt, da sie spezifische Werte für Transfer Family Ereignisse enthalten. Definitionen der anderen Metadatenfelder, die in allen Ereignissen enthalten sind, finden Sie unter Referenz zur Ereignisstruktur im Amazon EventBridge Benutzerhandbuch.

```
{
  . . .,
  "detail-type": "string",
  "source": "aws.transfer",
  . . .,
  "detail": {
    "operation" : "string",
    "max-items" : "number",
    "connector-id" : "string",
    "output-directory-path" : "string",
    "listing-id" : "string",
    "transfer-id" : "string",
    "file-transfer-id" : "string",
    "url" : "string",
    "file-path" : "string",
    "status-code" : "string",
    "failure-code" : "string",
    "failure-message" : "string",
    "start-timestamp" : "string",
```

```
    "end-timestamp" : "string",
    "local-directory-path" : "string",
    "remote-directory-path" : "string"
    "item-count" : "number"
    "truncated" : "boolean"
    "bytes" : "number",
    "local-file-location" : {
      "domain" : "string",
      "bucket" : "string",
      "key" : "string"
    },
    "output-file-location" : {
      "domain" : "string",
      "bucket" : "string",
      "key" : "string"
    }
  }
}
```

`detail-type`

Identifiziert den Ereignistyp.

Für dieses Ereignis entspricht der Wert einem der zuvor aufgeführten SFTP-Connector-Ereignisnamen.

`source`

Identifiziert den Service, aus dem das Ereignis stammt. Für Transfer Family Ereignisse ist `aws.transfer` dieser Wert.

`detail`

Ein JSON-Objekt, das Informationen zum Ereignis enthält. Der Dienst, der das Ereignis generiert, bestimmt den Inhalt dieses Felds.

Für dieses Ereignis umfassen die Daten Folgendes:

`max-items`

Die maximale Anzahl von Verzeichnis-/Dateinamen, die zurückgegeben werden sollen.

`operation`

Ob die `StartFileTransfer` Anfrage eine Datei sendet oder abruft. Werte:`SEND|RETRIEVE`.

`connector-id`

Die eindeutige Kennung für den verwendeten SFTP-Connector.

`output-directory-path`

Der Pfad (Bucket und Präfix) in Amazon S3 zum Speichern der Ergebnisse der Datei-/Verzeichnisauflistung.

`listing-id`

Eine eindeutige Kennung für den `StartDirectoryListing` API-Aufruf. Diese Kennung kann verwendet werden, um in den CloudWatch Protokollen den Status der Angebotsanfrage zu überprüfen.

`transfer-id`

Die eindeutige Kennung für das Übertragungsereignis (eine `StartFileTransfer` Anfrage).

`file-transfer-id`

Die eindeutige Kennung für die übertragene Datei.

`url`

Die URL des AS2- oder SFTP-Endpunkts des Partners.

`file-path`

Der Speicherort und die Datei, die gesendet oder abgerufen werden.

`status-code`

Ob die Übertragung erfolgreich ist. Werte:`FAILED | COMPLETED`.

`failure-code`

Bei fehlgeschlagenen Übertragungen der Ursachencode, warum die Übertragung fehlgeschlagen ist.

`failure-message`

Bei fehlgeschlagenen Übertragungen die Details, warum die Übertragung fehlgeschlagen ist.

`start-timestamp`

Bei erfolgreichen Übertragungen der Zeitstempel für den Beginn der Dateiverarbeitung.

`end-timestamp`

Bei erfolgreichen Übertragungen der Zeitstempel für den Abschluss der Dateiverarbeitung.

`local-directory-path`

Bei `RETRIEVE` Anfragen der Ort, an dem die abgerufene Datei abgelegt werden soll.

`remote-directory-path`

Bei `SEND` Anfragen das Dateiverzeichnis, in dem die Datei auf dem SFTP-Server des Partners abgelegt werden soll. Dies ist der Wert für den`RemoteDirectoryPath`, den der Benutzer an die `StartFileTransfer` Anfrage übergeben hat. Sie können ein Standardverzeichnis auf dem SFTP-Server des Partners angeben. Wenn ja, ist dieses Feld leer.

`item-count`

Die Anzahl der Elemente (Verzeichnisse und Dateien), die für die Angebotsanfrage zurückgegeben wurden.

`truncated`

Ob die Listenausgabe alle im Remote-Verzeichnis enthaltenen Elemente enthält oder nicht.

`bytes`

Die Anzahl der Byte, die übertragen werden. Der Wert ist 0 für fehlgeschlagene Übertragungen.

`local-file-location`

Dieser Parameter enthält die Details zum Speicherort der AWS Speicherdatei.

`domain`

Der Speicher, der verwendet wird. Derzeit ist der einzige Wert`S3`.

`bucket`

Der Container für das Objekt in Amazon S3.

`key`

Der dem Objekt in Amazon S3 zugewiesene Name.

`output-file-location`

Dieser Parameter enthält die Details des AWS Speicherorts, an dem die Ergebnisse der Verzeichnisliste gespeichert werden sollen.

`domain`

Der verwendete Speicher. Derzeit ist der einzige Wert`S3`.

`bucket`

Der Container für das Objekt in Amazon S3.

`key`

Der dem Objekt in Amazon S3 zugewiesene Name.

Example Beispielereignis beim Senden der SFTP-Connector-Datei fehlgeschlagen

Das folgende Beispiel zeigt ein Ereignis, bei dem ein SFTP-Connector beim Versuch, eine Datei an einen Remote-SFTP-Server zu senden, fehlgeschlagen ist.

```
{
    "version": "0",
    "id": "event-ID",
    "detail-type": "SFTP Connector File Send Failed",
    "source": "aws.transfer",
    "account": "123456789012",
    "time": "2024-01-24T19:30:45Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:transfer:us-east-1:123456789012:connector/c-f1111aaaa2222bbbb3"
    ],
    "detail": {
        "operation": "SEND",
        "connector-id": "c-f1111aaaa2222bbbb3",
        "transfer-id": "transfer-ID",
        "file-transfer-id": "file-transfer-ID",
        "url": "sftp://s-21a23456789012a.server.transfer.us-east-1.amazonaws.com",
        "file-path": "/DOC-EXAMPLE-BUCKET/testfile.txt",
        "status-code": "FAILED",
        "failure-code": "CONNECTION_ERROR",
        "failure-message": "Unknown Host",
        "remote-directory-path": "",
        "bytes": 0,
        "start-timestamp": "2024-01-24T18:29:33.658729Z",
        "end-timestamp": "2024-01-24T18:29:33.993196Z",
        "local-file-location": {
            "domain": "S3",
            "bucket": "DOC-EXAMPLE-BUCKET",
            "key": "testfile.txt"
        }
```

```
    }
}
```

Example Beispielereignis „SFTP-Connector File Retrieve Completed“

Das folgende Beispiel zeigt ein Ereignis, bei dem ein SFTP-Connector erfolgreich eine von einem Remote-SFTP-Server gesendete Datei abgerufen hat.

```
{
    "version": "0",
    "id": "event-ID",
    "detail-type": "SFTP Connector File Retrieve Completed",
    "source": "aws.transfer",
    "account": "123456789012",
    "time": "2024-01-24T18:28:08Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:transfer:us-east-1:123456789012:connector/c-f1111aaaa2222bbbb3"
    ],
    "detail": {
        "operation": "RETRIEVE",
        "connector-id": "c-fc68000012345aa18",
        "transfer-id": "file-transfer-ID",
        "file-transfer-id": "file-transfer-ID",
        "url": "sftp://s-21a23456789012a.server.transfer.us-east-1.amazonaws.com",
        "file-path": "testfile.txt",
        "status-code": "COMPLETED",
        "local-directory-path": "/DOC-EXAMPLE-BUCKET",
        "bytes": 63533,
        "start-timestamp": "2024-01-24T18:28:07.632388Z",
        "end-timestamp": "2024-01-24T18:28:07.774898Z",
        "local-file-location": {
            "domain": "S3",
            "bucket": "DOC-EXAMPLE-BUCKET",
            "key": "testfile.txt"
        }
    }
}
```

Example Beispielereignis mit der Verzeichnisliste des SFTP-Connectors abgeschlossen

Das folgende Beispiel zeigt ein Ereignis, bei dem ein Startaufruf für die Verzeichnisauflistung eine Auflistungsdatei von einem Remote-SFTP-Server abgerufen hat.

```
{
    "version": "0",
    "id": "event-ID",
    "detail-type": "SFTP Connector Directory Listing Completed",
    "source": "aws.transfer",
    "account": "123456789012",
    "time": "2024-01-24T18:28:08Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:transfer:us-east-1:123456789012:connector/c-f1111aaaa2222bbbb3"
    ],
    "detail": {
        "max-items": 10000,
        "connector-id": "c-fc68000012345aa18",
        "output-directory-path": "/DOC-EXAMPLE-BUCKET/example/file-listing-output",
        "listing-id": "123456-23aa-7980-abc1-1a2b3c4d5e",
        "url": "sftp://s-21a23456789012a.server.transfer.us-east-1.amazonaws.com",

        "status-code": "COMPLETED",
        "remote-directory-path": "/home",
        "item-count": 10000,
        "truncated": true,
        "start-timestamp": "2024-01-24T18:28:07.632388Z",
        "end-timestamp": "2024-01-24T18:28:07.774898Z",
        "output-file-location": {
            "domain": "S3",
            "bucket": "DOC-EXAMPLE-BUCKET",
            "key": "c-fc1ab90fd0d047e7a-70987273-49nn-4006-bab1-1a7290cc412ba.json"
        }
    }
}
```

## AS2-Ereignisse

Im Folgenden sind die Detailfelder für AS2-Ereignisse aufgeführt:

- Empfang der AS2-Nutzlast abgeschlossen
- Der Empfang der AS2-Nutzdaten ist fehlgeschlagen
- Der Versand der AS2-Nutzlast ist abgeschlossen
- Das Senden der AS2-Nutzdaten ist fehlgeschlagen
- AS2 MDN-Empfang abgeschlossen

- AS2 MDN-Empfang fehlgeschlagen
- AS2 MDN-Sendung abgeschlossen
- AS2 MDN-Sendung ist fehlgeschlagen

Die `detail-type` Felder `source` und sind unten aufgeführt, da sie spezifische Werte für Transfer Family Ereignisse enthalten. Definitionen der anderen Metadatenfelder, die in allen Ereignissen enthalten sind, finden Sie unter Referenz zur Ereignisstruktur im Amazon EventBridge Benutzerhandbuch.

```
{
   . . .,
  "detail-type": "string",
  "source": "aws.transfer",
  . . .,
  "detail": {
    "s3-attributes" : {
      "file-bucket" : "string",
      "file-key" : "string",
      "json-bucket" : "string",
      "json-key" : "string",
      "mdn-bucket" : "string",
      "mdn-key" : "string"
      }
    "mdn-subject" : "string",
    "mdn-message-id" : "string",
    "disposition" : "string",
    "bytes" : "number",
    "as2-from" : "string",
    "as2-message-id" : "string",
    "as2-to" : "string",
    "connector-id" : "string",
    "client-ip" : "string",
    "agreement-id" : "string",
    "server-id" : "string",
    "requester-file-name" : "string",
    "message-subject" : "string",
    "start-timestamp" : "string",
    "end-timestamp" : "string",
    "status-code" : "string",
    "failure-code" : "string",
    "failure-message" : "string",
```

```
    "transfer-id" : "string"
  }
}
```

`detail-type`

Identifiziert den Ereignistyp.

Für dieses Ereignis ist der Wert eines der zuvor aufgeführten AS2-Ereignisse.

`source`

Identifiziert den Service, aus dem das Ereignis stammt. Für Transfer Family Ereignisse ist `aws.transfer` dieser Wert.

`detail`

Ein JSON-Objekt, das Informationen zum Ereignis enthält. Der Service, der das Ereignis generiert, bestimmt den Inhalt dieses Feldes.

`s3-attributes`

Identifiziert den Amazon S3 S3-Bucket und den Schlüssel für die übertragene Datei. Bei MDN-Ereignissen identifiziert es auch den Bucket und den Schlüssel für die MDN-Datei.

`file-bucket`

Der Container für das Objekt in Amazon S3.

`file-key`

Der dem Objekt in Amazon S3 zugewiesene Name.

`json-bucket`

Bei ABGESCHLOSSENEN oder FEHLGESCHLAGENEN Übertragungen der Container für die JSON-Datei.

`json-key`

Bei ABGESCHLOSSENEN oder FEHLGESCHLAGENEN Übertragungen der Name, der der JSON-Datei in Amazon S3 zugewiesen wurde.

`mdn-bucket`

Bei MDN-Ereignissen der Container für die MDN-Datei.

`mdn-key`

Bei MDN-Ereignissen der Name, der der MDN-Datei in Amazon S3 zugewiesen wurde.

`mdn-subject`

Bei MDN-Ereignissen eine Textbeschreibung für die Nachrichtenverwaltung.

`mdn-message-id`

Bei MDN-Ereignissen eine eindeutige ID für die MDN-Nachricht.

`disposition`

Bei MDN-Ereignissen die Kategorie für die Disposition.

`bytes`

Die Anzahl der Byte in der Nachricht.

`as2-from`

Der AS2-Handelspartner, der die Nachricht sendet.

`as2-message-id`

Eine eindeutige Kennung für die AS2-Nachricht, die übertragen wird.

`as2-to`

Der AS2-Handelspartner, der die Nachricht empfängt.

`connector-id`

Für AS2-Nachrichten, die von einem Transfer Family Family-Server an einen Handelspartner gesendet werden, wird die eindeutige Kennung für den AS2-Connector verwendet.

`client-ip`

Bei Serverereignissen (Übertragungen von einem Geschäftspartner zu einem Transfer Family Family-Server) die IP-Adresse des an der Übertragung beteiligten Clients.

`agreement-id`

Bei Serverereignissen die eindeutige Kennung für die AS2-Vereinbarung.

`server-id`

Für Serverereignisse eine eindeutige ID nur für den Transfer Family Family-Server.

`requester-file-name`

Bei Payload-Ereignissen der ursprüngliche Name der Datei, die während der Übertragung empfangen wurde.

`message-subject`

Eine Textbeschreibung für den Betreff der Nachricht.

`start-timestamp`

Bei erfolgreichen Übertragungen der Zeitstempel für den Beginn der Dateiverarbeitung.

`end-timestamp`

Bei erfolgreichen Übertragungen der Zeitstempel für den Abschluss der Dateiverarbeitung.

`status-code`

Der Code, der dem Status des AS2-Nachrichtenübertragungsprozesses entspricht. Zulässige Werte: `COMPLETED | FAILED | PROCESSING`.

`failure-code`

Bei fehlgeschlagenen Übertragungen die Kategorie, warum die Übertragung fehlgeschlagen ist.

`failure-message`

Bei fehlgeschlagenen Übertragungen die Details, warum die Übertragung fehlgeschlagen ist.

`transfer-id`

Die eindeutige Kennung für das Übertragungsereignis.

Example Beispielereignis „AS2 Payload Receive Completed“

```
{
    "version": "0",
     "id": "event-ID",
    "detail-type": "AS2 Payload Receive Completed",
    "source": "aws.transfer",
    "account": "076722215406",
    "time": "2024-02-07T06:47:05Z",
    "region": "us-east-1",
    "resources": ["arn:aws:transfer:us-east-1:076722215406:connector/
c-1111aaaa2222bbbb3"],
```

```
    "detail": {
        "s3-attributes": {
            "file-key": "/inbound/processed/testAs2Message.dat",
            "file-bucket": "DOC-EXAMPLE-BUCKET"
        },
        "client-ip": "client-IP-address",
        "requester-file-name": "testAs2MessageVerifyFile.dat",
        "end-timestamp": "2024-02-07T06:47:06.040031Z",
        "as2-from": "as2-from-ID",
        "as2-message-id": "as2-message-ID",
        "message-subject": "Message from AS2 tests",
        "start-timestamp": "2024-02-07T06:47:05.410Z",
        "status-code": "PROCESSING",
        "bytes": 63,
        "as2-to": "as2-to-ID",
        "agreement-id": "a-1111aaaa2222bbbb3",
        "server-id": "s-1234abcd5678efghi"
    }
}
```

Example Beispielereignis „AS2 MDN Receive Failed“

```
{
  "version": "0",
  "id": "event-ID",
  "detail-type": "AS2 MDN Receive Failed",
  "source": "aws.transfer",
  "account": "889901007463",
  "time": "2024-02-06T22:05:09Z",
  "region": "us-east-1",
  "resources": ["arn:aws:transfer:us-east-1:076722215406:server/s-1111aaaa2222bbbb3"],
  "detail": {
      "mdn-subject": "Your Requested MDN Response re: Test run from Id 123456789abcde
 to partner ijklmnop987654",
      "s3-attributes": {
          "json-bucket": "DOC-EXAMPLE-BUCKET1",
          "file-key": "/as2Integ/TestOutboundWrongCert.dat",
          "file-bucket": "DOC-EXAMPLE-BUCKET2",
          "json-key": "/as2Integ/failed/TestOutboundWrongCert.dat.json"
      },
      "mdn-message-id": "MDN-message-ID",
      "end-timestamp": "2024-02-06T22:05:09.479878Z",
      "as2-from": "PartnerA",
```

```
      "as2-message-id": "as2-message-ID",
      "connector-id": "c-1234abcd5678efghj",
      "message-subject": "Test run from Id 123456789abcde to partner ijklmnop987654",
      "start-timestamp": "2024-02-06T22:05:03Z",
      "failure-code": "VERIFICATION_FAILED_NO_MATCHING_KEY_FOUND",
      "status-code": "FAILED",
      "as2-to": "MyCompany",
      "failure-message": "No public certificate matching message signature could be
 found in profile: p-1234abcd5678efghj",
      "transfer-id": "transfer-ID"
  }
}
```

# Sicherheit in AWS Transfer Family

Cloud-Sicherheit AWS hat höchste Priorität. Als AWS Kunde profitieren Sie von einer Rechenzentrums- und Netzwerkarchitektur, die darauf ausgelegt sind, die Anforderungen der sicherheitssensibelsten Unternehmen zu erfüllen.

Sicherheit ist eine gemeinsame Verantwortung von Ihnen AWS und Ihnen. Das Modell der geteilten Verantwortung beschreibt dies als Sicherheit der Cloud und Sicherheit in der Cloud:

Informationen darüber, ob AWS-Service ein AWS-Services in den Geltungsbereich bestimmter Compliance-Programme fällt, finden Sie unter Umfang nach Compliance-Programm AWS-Services unter . Wählen Sie dort das Compliance-Programm aus, an dem Sie interessiert sind. Allgemeine Informationen finden Sie unter AWS Compliance-Programme AWS .

Sie können Prüfberichte von Drittanbietern unter herunterladen AWS Artifact. Weitere Informationen finden Sie unter Berichte herunterladen unter .

Ihre Verantwortung für die Einhaltung der Vorschriften bei der Nutzung AWS-Services hängt von der Vertraulichkeit Ihrer Daten, den Compliance-Zielen Ihres Unternehmens und den geltenden Gesetzen und Vorschriften ab. AWS stellt die folgenden Ressourcen zur Verfügung, die Sie bei der Einhaltung der Vorschriften unterstützen:

- Schnellstartanleitungen zu Sicherheit und Compliance — In diesen Bereitstellungsleitfäden werden architektonische Überlegungen erörtert und Schritte für die Bereitstellung von Basisumgebungen beschrieben AWS , bei denen Sicherheit und Compliance im Mittelpunkt stehen.
- Architecting for HIPAA Security and Compliance on Amazon Web Services — In diesem Whitepaper wird beschrieben, wie Unternehmen HIPAA-fähige Anwendungen erstellen AWS können.

  > **Note**
  >
  > AWS-Services Nicht alle sind HIPAA-fähig. Weitere Informationen finden Sie in der Referenz für HIPAA-berechtigte Services.

- AWS Compliance-Ressourcen — Diese Sammlung von Arbeitsmappen und Leitfäden gilt möglicherweise für Ihre Branche und Ihren Standort.
- AWS Leitfäden zur Einhaltung von Vorschriften für Kunden — Verstehen Sie das Modell der gemeinsamen Verantwortung aus dem Blickwinkel der Einhaltung von Vorschriften. In den

Leitfäden werden die bewährten Verfahren zur Sicherung zusammengefasst AWS-Services 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.

- Evaluierung von Ressourcen anhand von Regeln im AWS Config Entwicklerhandbuch — Der AWS Config Service bewertet, wie gut Ihre Ressourcenkonfigurationen den internen Praktiken, Branchenrichtlinien und Vorschriften entsprechen.
- AWS Security Hub— Dies AWS-Service bietet einen umfassenden Überblick über Ihren internen Sicherheitsstatus. 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.
- Amazon GuardDuty — Dies AWS-Service erkennt potenzielle Bedrohungen für Ihre Workloads AWS-Konten, Container und Daten, indem es Ihre Umgebung auf verdächtige und böswillige Aktivitäten überwacht. GuardDuty kann Ihnen helfen, verschiedene Compliance-Anforderungen wie PCI DSS zu erfüllen, indem es die in bestimmten Compliance-Frameworks vorgeschriebenen Anforderungen zur Erkennung von Eindringlingen erfüllt.
- AWS Audit Manager— Auf diese AWS-Service Weise können Sie Ihre AWS Nutzung kontinuierlich überprüfen, um das Risikomanagement und die Einhaltung von Vorschriften und Industriestandards zu vereinfachen.

Diese Dokumentation hilft Ihnen zu verstehen, wie Sie das Modell der gemeinsamen Verantwortung bei der Verwendung anwenden können AWS Transfer Family. In den folgenden Themen erfahren Sie, wie Sie die Konfiguration vornehmen AWS Transfer Family , um Ihre Sicherheits- und Compliance-Ziele zu erreichen. Sie erfahren auch, wie Sie andere AWS Dienste nutzen können, die Sie bei der Überwachung und Sicherung Ihrer AWS Transfer Family Ressourcen unterstützen.

Wir bieten einen Workshop mit präskriptiven Anleitungen und einem praktischen Lab an, in dem Sie erfahren, wie Sie eine skalierbare und sichere Dateiübertragungsarchitektur aufbauen können, AWS ohne bestehende Anwendungen ändern oder die Serverinfrastruktur verwalten zu müssen. Die Einzelheiten zu diesem Workshop finden Sie hier.

Themen

- Sicherheitsrichtlinien für AWS Transfer Family Server
- AWS Transfer Family Sicherheitsrichtlinien für SFTP-Konnektoren

- Verwenden Sie den hybriden Schlüsselaustausch nach dem Quantum-Verfahren mit AWS Transfer Family
- Datenschutz in AWS Transfer Family
- Identitäts- und Zugriffsmanagement für AWS Transfer Family
- Konformitätsprüfung für AWS Transfer Family
- Resilienz in AWS Transfer Family
- Sicherheit der Infrastruktur in AWS Transfer Family
- Fügen Sie eine Firewall für Webanwendungen hinzu
- Serviceübergreifende Confused-Deputy-Prävention
- AWS verwaltete Richtlinien für AWS Transfer Family

# Sicherheitsrichtlinien für AWS Transfer Family Server

AWS Transfer Family Mithilfe der Serversicherheitsrichtlinien können Sie die Anzahl der mit Ihrem Server verknüpften kryptografischen Algorithmen (Message Authentication Codes (MACs), Key Exchanges (KEXs) und Cipher Suites) einschränken. Eine Liste der unterstützten kryptografischen Algorithmen finden Sie unter. Kryptografische Algorithmen Eine Liste der unterstützten Schlüsselalgorithmen für die Verwendung mit Serverhostschlüsseln und vom Dienst verwalteten Benutzerschlüsseln finden Sie unter. Unterstützte Algorithmen für Benutzer- und Serverschlüssel

> **Note**
>
> Wir empfehlen dringend, Ihre Server auf unsere neuesten Sicherheitsrichtlinien zu aktualisieren. Unsere neueste Sicherheitsrichtlinie ist die Standardeinstellung. Jedem Kunden, der einen Transfer Family Family-Server unter Verwendung der Standardsicherheitsrichtlinie erstellt CloudFormation und diese akzeptiert, wird automatisch die neueste Richtlinie zugewiesen. Wenn Sie Bedenken hinsichtlich der Client-Kompatibilität haben, geben Sie bitte ausdrücklich an, welche Sicherheitsrichtlinie Sie bei der Erstellung oder Aktualisierung eines Servers verwenden möchten, anstatt die Standardrichtlinie zu verwenden, die sich ändern kann.
> Informationen zum Ändern der Sicherheitsrichtlinie für einen Server finden Sie unter. Bearbeiten Sie die Sicherheitsrichtlinie

Weitere Informationen zur Sicherheit in Transfer Family finden Sie im Blogbeitrag Wie Transfer Family Ihnen helfen kann, eine sichere, konforme verwaltete Dateiübertragungslösung zu entwickeln.

Themen


- Kryptografische Algorithmen
- TransferSecurityRichtlinie 2024-01
- TransferSecurityRichtlinie 2023-05
- TransferSecurityRichtlinie — 2022 — 03
- TransferSecurityPolitik — 2020-06
- TransferSecurityRichtlinie 2018-11
- TransferSecurityRichtlinie-FIPS-2024-01/ Richtlinie-FIPS-2024-05 TransferSecurity
- TransferSecurityRichtlinie-FIPS-2023-05
- TransferSecurityRichtlinie-FIPS-2020-06
- Sicherheitsrichtlinien nach Quantum


> **Note**
>
> `TransferSecurityPolicy-2024-01`ist die Standard-Sicherheitsrichtlinie, die Ihrem Server zugewiesen wird, wenn Sie einen Server mithilfe der Konsole, API oder CLI erstellen.

## Kryptografische Algorithmen

Für Hostschlüssel unterstützen wir die folgenden Algorithmen:

- `rsa-sha2-256`
- `rsa-sha2-512`
- `ecdsa-sha2-nistp256`
- `ecdsa-sha2-nistp384`
- `ecdsa-sha2-nistp521`
- `ssh-ed25519`

Darüber hinaus ermöglichen die folgenden Sicherheitsrichtlinien`ssh-rsa`:

- TransferSecurityRichtlinie 2018-11
- TransferSecurityPolitik-2020-06
- TransferSecurityRichtlinie-FIPS-2020-06
- TransferSecurityRichtlinie-FIPS-2023-05
- TransferSecurityRichtlinie-FIPS-2024-01
- TransferSecurityRichtlinie-PQ-SSH-FIPS-Experimental-2023-04

> **Note**
>
> Es ist wichtig, den Unterschied zwischen dem RSA-Schlüsseltyp — der immer gilt — und dem RSA-Host-Schlüsselalgorithmus zu verstehen, bei dem es sich um einen der unterstützten Algorithmen handeln kann. `ssh-rsa`

Im Folgenden finden Sie eine Liste der unterstützten kryptografischen Algorithmen für jede Sicherheitsrichtlinie.

> **Note**
>
> Beachten Sie in der folgenden Tabelle und den Richtlinien die folgende Verwendung von Algorithmustypen.
>
> - SFTP-Server verwenden nur Algorithmen in den SshMacsAbschnitten SshCiphersSshKexs, und.
> - FTPS-Server verwenden in diesem Abschnitt nur Algorithmen. TlsCiphers
> - FTP-Server verwenden keinen dieser Algorithmen, da sie keine Verschlüsselung verwenden.
> - Die Sicherheitsrichtlinien FIPS-2024-05 und FIPS-2024-01 sind identisch, mit der Ausnahme, dass FIPS-2024-05 den Algorithmus nicht unterstützt. `ssh-rsa`

| Sicherheitsrichtlinie | 2024-01 | 2023-05 | 2022 03 | 2020-06 | FIPS-2024-05<br>FIPS-2024-01 | FIPS-2023-05 | FIPS-2020-06 | 2018-11 |
|---|---|---|---|---|---|---|---|---|
| SshCiphers | | | | | | | | |
| aes128-ctr | ♦ | | | ♦ | ♦ | | ♦ | ♦ |
| aes128-gcm@openssh.com | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ |
| aes192-ctr | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ |
| aes256-ctr | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ |
| aes256-gcm@openssh.com | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ |
| chacha20-poly1305@openssh.com | | | | ♦ | | | | ♦ |
| SshKexs | | | | | | | | |
| Kurve25519-sha256 | ♦ | ♦ | ♦ | | | | | ♦ |
| curve25519- | ♦ | ♦ | ♦ | | | | | ♦ |

| Sicherheitsrichtlinie | 2024-01 | 2023-05 | 2022 03 | 2020-06 | FIPS-2024-05<br>FIPS-2024-01 | FIPS-2023-05 | FIPS-2020-06 | 2018-11 |
|---|---|---|---|---|---|---|---|---|
| sha256@libssh.org | | | | | | | | |
| diffie-hellman-group-ppe14-sha1 | | | | | | | | ♦ |
| diffie-hellman-group-ppe-14-sha256 | | | | ♦ | | | ♦ | ♦ |
| diffie-hellman-group-ppe-16-sha512 | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ |
| diffie-hellman-group-ppe-18-sha512 | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ |

| Sicherheitsrichtlinie | 2024-01 | 2023-05 | 2022 03 | 2020-06 | FIPS-2024-05<br>FIPS-2024-01 | FIPS-2023-05 | FIPS-2020-06 | 2018-11 |
|---|---|---|---|---|---|---|---|---|
| Diffie-Hellman-Gruppenaustausch-SHA256 | | ♦ | ♦ | ♦ | | ♦ | ♦ | ♦ |
| ecdh-nistp256-kyber-512r3-sha256-d00@openquantumsafe.org | ♦ | | | | ♦ | | | |
| ecdh-nistp384-kyber-768r3-sha384-d00@openquantumsafe.org | ♦ | | | | ♦ | | | |

| Sicherheitsrichtlinie | 2024-01 | 2023-05 | 2022 03 | 2020-06 | FIPS-2024-05<br>FIPS-2024-01 | FIPS-2023-05 | FIPS-2020-06 | 2018-11 |
|---|---|---|---|---|---|---|---|---|
| ecdh-nistp521-kyber-1024r3-sha512-d00@openquantumsafe.org | ♦ | | | | ♦ | | | |
| ecdh-sha2-nistp256 | ♦ | | | ♦ | ♦ | | ♦ | ♦ |
| ecdh-sha2-nistp384 | ♦ | | | ♦ | ♦ | | ♦ | ♦ |
| ecdh-sha2-nistp521 | ♦ | | | ♦ | ♦ | | ♦ | ♦ |
| x25519-kyber-512r3-sha256-d00@amazon.com | ♦ | | | | | | | |

| Sicherheitsrichtlinie | 2024-01 | 2023-05 | 2022 03 | 2020-06 | FIPS-2024-05<br>FIPS-2024-01 | FIPS-2023-05 | FIPS-2020-06 | 2018-11 |
|---|---|---|---|---|---|---|---|---|
| SshMacs | | | | | | | | |
| hmac-sha 1 | | | | | | | | ♦ |
| hmac-sha1-etm@openssh.com | | | | | | | | ♦ |
| hmac-sha2-256 | | | ♦ | ♦ | | | ♦ | ♦ |
| hmac-sha2-256-etm@openssh.com | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ |
| hmac-sha2-512 | | | ♦ | ♦ | | | ♦ | ♦ |
| hmac-sha2-512-etm@openssh.com | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ |

| Sicherheitsrichtlinie | 2024-01 | 2023-05 | 2022 03 | 2020-06 | FIPS-2024-05<br>FIPS-2024-01 | FIPS-2023-05 | FIPS-2020-06 | 2018-11 |
|---|---|---|---|---|---|---|---|---|
| umac-128-etm@openssh.com | | | | ♦ | | | | ♦ |
| umac-128@openssh.com | | | | ♦ | | | | ♦ |
| umac-64-etm@openssh.com | | | | | | | | ♦ |
| umac-64@openssh.com | | | | | | | | ♦ |
| TlsCiphers | | | | | | | | |
| TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ |
| TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ |

| Sicherheitsrichtlinie | 2024-01 | 2023-05 | 2022 03 | 2020-06 | FIPS-2024-05<br>FIPS-2024-01 | FIPS-2023-05 | FIPS-2020-06 | 2018-11 |
|---|---|---|---|---|---|---|---|---|
| TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ | |
| TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ | |
| TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ | |
| TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ | |
| TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ | |

| Sicherheitsrichtlinie | 2024-01 | 2023-05 | 2022 03 | 2020-06 | FIPS-2024-05<br>FIPS-2024-01 | FIPS-2023-05 | FIPS-2020-06 | 2018-11 |
|---|---|---|---|---|---|---|---|---|
| TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ | ♦ |
| TLS_RSA_WITH_AES_128_CBC_SHA256 | | | | | | | | ♦ |
| TLS_RSA_WITH_AES_256_CBC_SHA256 | | | | | | | | ♦ |

## TransferSecurityRichtlinie 2024-01

Im Folgenden wird die Sicherheitsrichtlinie TransferSecurityPolicy -2024-01 dargestellt.

```
{
    "SecurityPolicy": {
        "Fips": false,
        "SecurityPolicyName": "TransferSecurityPolicy-2024-01",
        "SshCiphers": [
            "aes128-gcm@openssh.com",
            "aes256-gcm@openssh.com",
            "aes128-ctr",
            "aes256-ctr",
            "aes192-ctr"
        ],
        "SshKexs": [
            "ecdh-nistp384-kyber-768r3-sha384-d00@openquantumsafe.org",
```

```
            "x25519-kyber-512r3-sha256-d00@amazon.com",
            "ecdh-nistp256-kyber-512r3-sha256-d00@openquantumsafe.org",
            "ecdh-nistp521-kyber-1024r3-sha512-d00@openquantumsafe.org",
            "ecdh-sha2-nistp256",
            "ecdh-sha2-nistp384",
            "ecdh-sha2-nistp521",
            "curve25519-sha256",
            "curve25519-sha256@libssh.org",
            "diffie-hellman-group18-sha512",
            "diffie-hellman-group16-sha512",
            "diffie-hellman-group-exchange-sha256"
        ],
        "SshMacs": [
            "hmac-sha2-256-etm@openssh.com",
            "hmac-sha2-512-etm@openssh.com"
        ],
        "TlsCiphers": [
            "TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256",
            "TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256",
            "TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256",
            "TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256",
            "TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384",
            "TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384",
            "TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384",
            "TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384"
        ]
    }
}
```

## TransferSecurityRichtlinie 2023-05

Im Folgenden wird die Sicherheitsrichtlinie -2023-05 dargestellt. TransferSecurityPolicy

```
{
    "SecurityPolicy": {
        "Fips": false,
        "SecurityPolicyName": "TransferSecurityPolicy-2023-05",
        "SshCiphers": [
            "aes256-gcm@openssh.com",
            "aes128-gcm@openssh.com",
            "aes256-ctr",
            "aes192-ctr"
        ],
```

```
        "SshKexs": [
            "curve25519-sha256",
            "curve25519-sha256@libssh.org",
            "diffie-hellman-group16-sha512",
            "diffie-hellman-group18-sha512",
            "diffie-hellman-group-exchange-sha256"
        ],
        "SshMacs": [
            "hmac-sha2-512-etm@openssh.com",
            "hmac-sha2-256-etm@openssh.com"
        ],
        "TlsCiphers": [
            "TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256",
            "TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256",
            "TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256",
            "TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256",
            "TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384",
            "TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384",
            "TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384",
            "TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384"
        ]
    }
}
```

## TransferSecurityRichtlinie — 2022 — 03

Im Folgenden wird die Sicherheitsrichtlinie TransferSecurityPolicy -2022-03 dargestellt.

```
{
  "SecurityPolicy": {
    "Fips": false,
    "SecurityPolicyName": "TransferSecurityPolicy-2022-03",
    "SshCiphers": [
      "aes256-gcm@openssh.com",
      "aes128-gcm@openssh.com",
      "aes256-ctr",
      "aes192-ctr"
    ],
    "SshKexs": [
      "curve25519-sha256",
      "curve25519-sha256@libssh.org",
      "diffie-hellman-group16-sha512",
      "diffie-hellman-group18-sha512",
```

```
      "diffie-hellman-group-exchange-sha256"
    ],
    "SshMacs": [
      "hmac-sha2-512-etm@openssh.com",
      "hmac-sha2-256-etm@openssh.com",
      "hmac-sha2-512",
      "hmac-sha2-256"
    ],
    "TlsCiphers": [
      "TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256",
      "TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256",
      "TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256",
      "TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256",
      "TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384",
      "TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384",
      "TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384",
      "TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384"
    ]
  }
}
```

## TransferSecurityPolitik — 2020-06

Im Folgenden wird die Sicherheitsrichtlinie TransferSecurityPolicy -2020-06 dargestellt.

```
{
  "SecurityPolicy": {
    "Fips": false,
    "SecurityPolicyName": "TransferSecurityPolicy-2020-06",
    "SshCiphers": [
      "chacha20-poly1305@openssh.com",
      "aes128-ctr",
      "aes192-ctr",
      "aes256-ctr",
      "aes128-gcm@openssh.com",
      "aes256-gcm@openssh.com"
    ],
    "SshKexs": [
      "ecdh-sha2-nistp256",
      "ecdh-sha2-nistp384",
      "ecdh-sha2-nistp521",
      "diffie-hellman-group-exchange-sha256",
      "diffie-hellman-group16-sha512",
```

```
      "diffie-hellman-group18-sha512",
      "diffie-hellman-group14-sha256"
    ],
    "SshMacs": [
      "umac-128-etm@openssh.com",
      "hmac-sha2-256-etm@openssh.com",
      "hmac-sha2-512-etm@openssh.com",
      "umac-128@openssh.com",
      "hmac-sha2-256",
      "hmac-sha2-512"
    ],
    "TlsCiphers": [
      "TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256",
      "TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256",
      "TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256",
      "TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256",
      "TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384",
      "TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384",
      "TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384",
      "TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384"
    ]
  }
}
```

## TransferSecurityRichtlinie 2018-11

Im Folgenden wird die Sicherheitsrichtlinie TransferSecurityPolicy -2018-11 dargestellt.

```
{
  "SecurityPolicy": {
    "Fips": false,
    "SecurityPolicyName": "TransferSecurityPolicy-2018-11",
    "SshCiphers": [
      "chacha20-poly1305@openssh.com",
      "aes128-ctr",
      "aes192-ctr",
      "aes256-ctr",
      "aes128-gcm@openssh.com",
      "aes256-gcm@openssh.com"
    ],
    "SshKexs": [
      "curve25519-sha256",
      "curve25519-sha256@libssh.org",
```

```
      "ecdh-sha2-nistp256",
      "ecdh-sha2-nistp384",
      "ecdh-sha2-nistp521",
      "diffie-hellman-group-exchange-sha256",
      "diffie-hellman-group16-sha512",
      "diffie-hellman-group18-sha512",
      "diffie-hellman-group14-sha256",
      "diffie-hellman-group14-sha1"
    ],
    "SshMacs": [
      "umac-64-etm@openssh.com",
      "umac-128-etm@openssh.com",
      "hmac-sha2-256-etm@openssh.com",
      "hmac-sha2-512-etm@openssh.com",
      "hmac-sha1-etm@openssh.com",
      "umac-64@openssh.com",
      "umac-128@openssh.com",
      "hmac-sha2-256",
      "hmac-sha2-512",
      "hmac-sha1"
    ],
    "TlsCiphers": [
      "TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256",
      "TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256",
      "TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256",
      "TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256",
      "TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384",
      "TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384",
      "TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384",
      "TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384",
      "TLS_RSA_WITH_AES_128_CBC_SHA256",
      "TLS_RSA_WITH_AES_256_CBC_SHA256"
    ]
  }
}
```

## TransferSecurityRichtlinie-FIPS-2024-01/ Richtlinie-FIPS-2024-05 TransferSecurity

Im Folgenden werden die Sicherheitsrichtlinien -FIPS-2024-01 und -FIPS-2024-05 aufgeführt TransferSecurityPolicy. TransferSecurityPolicy

> **Note**
>
> Der FIPS-Dienstendpunkt und die Sicherheitsrichtlinien -FIPS-2024-01 und -FIPS-2024-05 sind nur in einigen Regionen verfügbar. TransferSecurityPolicy TransferSecurityPolicy AWS Weitere Informationen finden Sie unter AWS Transfer Family -Endpunkte und -Kontingente in der Allgemeine AWS-Referenz.
> Der einzige Unterschied zwischen diesen beiden Sicherheitsrichtlinien besteht darin, dass -FIPS-2024-01 den Algorithmus unterstützt und -FIPS-2024-05 nicht. TransferSecurityPolicy `ssh-rsa` TransferSecurityPolicy

```
{
    "SecurityPolicy": {
        "Fips": true,
        "SecurityPolicyName": "TransferSecurityPolicy-FIPS-2024-01",
        "SshCiphers": [
            "aes128-gcm@openssh.com",
            "aes256-gcm@openssh.com",
            "aes128-ctr",
            "aes256-ctr",
            "aes192-ctr"
        ],
        "SshKexs": [
            "ecdh-nistp384-kyber-768r3-sha384-d00@openquantumsafe.org",
            "ecdh-nistp256-kyber-512r3-sha256-d00@openquantumsafe.org",
            "ecdh-nistp521-kyber-1024r3-sha512-d00@openquantumsafe.org",
            "ecdh-sha2-nistp256",
            "ecdh-sha2-nistp384",
            "ecdh-sha2-nistp521",
            "diffie-hellman-group18-sha512",
            "diffie-hellman-group16-sha512",
            "diffie-hellman-group-exchange-sha256"
        ],
        "SshMacs": [
            "hmac-sha2-256-etm@openssh.com",
            "hmac-sha2-512-etm@openssh.com"
        ],
        "TlsCiphers": [
            "TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256",
            "TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256",
            "TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256",
```

```
            "TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256",
            "TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384",
            "TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384",
            "TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384",
            "TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384"
        ]
    }
}
```

## TransferSecurityRichtlinie-FIPS-2023-05

Die Einzelheiten zur FIPS-Zertifizierung für finden Sie unter AWS Transfer Family https://csrc.nist.gov/projects/cryptographic-module-validation-program/validated-modules/search/all

Im Folgenden wird die Sicherheitsrichtlinie TransferSecurityPolicy -FIPS-2023-05 beschrieben.

> **Note**
>
> Der FIPS-Dienstendpunkt und die Sicherheitsrichtlinie TransferSecurityPolicy -FIPS-2023-05 sind nur in einigen Regionen verfügbar. AWS Weitere Informationen finden Sie unter AWS Transfer Family -Endpunkte und -Kontingente in der Allgemeine AWS-Referenz.

```
{
    "SecurityPolicy": {
        "Fips": true,
        "SecurityPolicyName": "TransferSecurityPolicy-FIPS-2023-05",
        "SshCiphers": [
            "aes256-gcm@openssh.com",
            "aes128-gcm@openssh.com",
            "aes256-ctr",
            "aes192-ctr"
        ],
        "SshKexs": [
            "diffie-hellman-group16-sha512",
            "diffie-hellman-group18-sha512",
            "diffie-hellman-group-exchange-sha256"
        ],
        "SshMacs": [
            "hmac-sha2-256-etm@openssh.com",
            "hmac-sha2-512-etm@openssh.com"
        ],
```

```
        "TlsCiphers": [
            "TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256",
            "TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256",
            "TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256",
            "TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256",
            "TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384",
            "TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384",
            "TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384",
            "TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384"
        ]
    }
}
```

## TransferSecurityRichtlinie-FIPS-2020-06

Die Einzelheiten zur FIPS-Zertifizierung für finden Sie unter AWS Transfer Family https://csrc.nist.gov/projects/cryptographic-module-validation-program/validated-modules/search/all

Im Folgenden wird die Sicherheitsrichtlinie TransferSecurityPolicy -FIPS-2020-06 dargestellt.

> **Note**
>
> Der FIPS-Dienstendpunkt und die Sicherheitsrichtlinie TransferSecurityPolicy -FIPS-2020-06 sind nur in einigen Regionen verfügbar. AWS Weitere Informationen finden Sie unter AWS Transfer Family -Endpunkte und -Kontingente in der Allgemeine AWS-Referenz.

```
{
  "SecurityPolicy": {
    "Fips": true,
    "SecurityPolicyName": "TransferSecurityPolicy-FIPS-2020-06",
    "SshCiphers": [
      "aes128-ctr",
      "aes192-ctr",
      "aes256-ctr",
      "aes128-gcm@openssh.com",
      "aes256-gcm@openssh.com"
    ],
    "SshKexs": [
      "ecdh-sha2-nistp256",
      "ecdh-sha2-nistp384",
      "ecdh-sha2-nistp521",
```

```
      "diffie-hellman-group-exchange-sha256",
      "diffie-hellman-group16-sha512",
      "diffie-hellman-group18-sha512",
      "diffie-hellman-group14-sha256"
    ],
    "SshMacs": [
      "hmac-sha2-256-etm@openssh.com",
      "hmac-sha2-512-etm@openssh.com",
      "hmac-sha2-256",
      "hmac-sha2-512"
    ],
    "TlsCiphers": [
      "TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256",
      "TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256",
      "TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256",
      "TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256",
      "TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384",
      "TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384",
      "TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384",
      "TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384"
    ]
  }
}
```

## Sicherheitsrichtlinien nach Quantum

In dieser Tabelle sind die Algorithmen für die Post-Quantum-Sicherheitsrichtlinien der Transfer Family aufgeführt. Diese Richtlinien werden unter ausführlich Verwenden Sie den hybriden Schlüsselaustausch nach dem Quantum-Verfahren mit AWS Transfer Family beschrieben.

Die Richtlinienlisten folgen der Tabelle.

| Sicherheitsrichtlinie | TransferSecurityPolicy-PQ-SSH-Experimental-2023-04 | TransferSecurityRichtlinie-PQ-SSH-FIPS-Experimentell-2023-04 |
|---|---|---|
| SSH ciphers | | |
| aes128-ctr | | ♦ |
| aes128-gcm@openssh.com | ♦ | ♦ |

| Sicherheitsrichtlinie | TransferSecurityPolicy-PQ-SSH-Experimental-2023-04 | TransferSecurityRichtlinie-PQ-SSH-FIPS-Experimentell-2023-04 |
|---|---|---|
| aes192-ctr | ♦ | ♦ |
| aes256-ctr | ♦ | ♦ |
| aes256-gcm@openssh.com | ♦ | ♦ |
| KEXs | | |
| ecdh-nistp256-kyber-512r3-sha256-d00@openquantumsafe.org | ♦ | ♦ |
| ecdh-nistp384-kyber-768r3-sha384-d00@openquantumsafe.org | ♦ | ♦ |
| ecdh-nistp521-kyber-1024r3-sha512-d00@openquantumsafe.org | ♦ | ♦ |
| x25519-kyber-512r3-sha256-d00@amazon.com | ♦ | |
| diffie-hellman-gruppe14-sha256 | | ♦ |
| diffie-hellman-gruppe-16-sha512 | ♦ | ♦ |
| diffie-hellman-gruppe-18-sha512 | ♦ | ♦ |
| ecdh-Sha2-Nistp384 | | ♦ |
| ecdh-sha2-nistp521 | | ♦ |

| Sicherheitsrichtlinie | TransferSecurityPolicy-PQ-SSH-Experimental-2023-04 | TransferSecurityRichtlinie-PQ-SSH-FIPS-Experimentell-2023-04 |
|---|---|---|
| Diffie-Hellman-Gruppenaustausch-SHA256 | ♦ | ♦ |
| ecdh-sha2-nistp256 | | ♦ |
| curve25519-sha256@libssh.org | ♦ | |
| Kurve 25519-sha256 | ♦ | |
| MACs | | |
| hmac-sha2-256-etm@openssh.com | ♦ | ♦ |
| hmac-sha2-256 | ♦ | ♦ |
| hmac-sha2-512-etm@openssh.com | ♦ | ♦ |
| hmac-sha2-512 | ♦ | ♦ |
| TLS ciphers | | |
| TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 | ♦ | ♦ |
| TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 | ♦ | ♦ |
| TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 | ♦ | ♦ |
| TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 | ♦ | ♦ |

| Sicherheitsrichtlinie | TransferSecurityPolicy-PQ-SSH-Experimental-2023-04 | TransferSecurityRichtlinie-PQ-SSH-FIPS-Experimentell-2023-04 |
|---|---|---|
| TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 | ♦ | ♦ |
| TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 | ♦ | ♦ |
| TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 | ♦ | ♦ |
| TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 | ♦ | ♦ |

## TransferSecurityRichtlinie-PQ-SSH-Experimental-2023-04

Im Folgenden wird die Sicherheitsrichtlinie -pq-SSH-Experimental-2023-04 dargestellt. TransferSecurityPolicy

```
{
    "SecurityPolicy": {
        "Fips": false,
        "SecurityPolicyName": "TransferSecurityPolicy-PQ-SSH-Experimental-2023-04",
        "SshCiphers": [
            "aes256-gcm@openssh.com",
            "aes128-gcm@openssh.com",
            "aes256-ctr",
            "aes192-ctr"
        ],
        "SshKexs": [
            "ecdh-nistp384-kyber-768r3-sha384-d00@openquantumsafe.org",
            "x25519-kyber-512r3-sha256-d00@amazon.com",
            "ecdh-nistp256-kyber-512r3-sha256-d00@openquantumsafe.org",
            "ecdh-nistp521-kyber-1024r3-sha512-d00@openquantumsafe.org",
            "curve25519-sha256",
            "curve25519-sha256@libssh.org",
            "diffie-hellman-group16-sha512",
            "diffie-hellman-group18-sha512",
            "diffie-hellman-group-exchange-sha256"
```

```
        ],
        "SshMacs": [
            "hmac-sha2-512-etm@openssh.com",
            "hmac-sha2-256-etm@openssh.com",
            "hmac-sha2-512",
            "hmac-sha2-256"
        ],
        "TlsCiphers": [
            "TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256",
            "TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256",
            "TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256",
            "TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256",
            "TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384",
            "TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384",
            "TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384",
            "TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384"
        ]
    }
}
```

## TransferSecurityRichtlinie-PQ-SSH-FIPS-Experimental-2023-04

Im Folgenden wird die Sicherheitsrichtlinie -pq-SSH-FIPS-Experimental-2023-04 dargestellt. TransferSecurityPolicy

```
{
    "SecurityPolicy": {
        "Fips": true,
        "SecurityPolicyName": "TransferSecurityPolicy-PQ-SSH-FIPS-
Experimental-2023-04",
        "SshCiphers": [
            "aes256-gcm@openssh.com",
            "aes128-gcm@openssh.com",
            "aes256-ctr",
            "aes192-ctr",
            "aes128-ctr"
        ],
        "SshKexs": [
            "ecdh-nistp384-kyber-768r3-sha384-d00@openquantumsafe.org",
            "ecdh-nistp256-kyber-512r3-sha256-d00@openquantumsafe.org",
            "ecdh-nistp521-kyber-1024r3-sha512-d00@openquantumsafe.org",
            "ecdh-sha2-nistp256",
            "ecdh-sha2-nistp384",
```

```
            "ecdh-sha2-nistp521",
            "diffie-hellman-group-exchange-sha256",
            "diffie-hellman-group16-sha512",
            "diffie-hellman-group18-sha512",
            "diffie-hellman-group14-sha256"
        ],
        "SshMacs": [
            "hmac-sha2-512-etm@openssh.com",
            "hmac-sha2-256-etm@openssh.com",
            "hmac-sha2-512",
            "hmac-sha2-256"
        ],
        "TlsCiphers": [
            "TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256",
            "TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256",
            "TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256",
            "TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256",
            "TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384",
            "TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384",
            "TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384",
            "TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384"
        ]
    }
}
```

# AWS Transfer Family Sicherheitsrichtlinien für SFTP-Konnektoren

Mit den Sicherheitsrichtlinien für den SFTP-Connector AWS Transfer Family können Sie die Anzahl der mit Ihrem SFTP-Connector verknüpften kryptografischen Algorithmen (Message Authentication Codes (MACs), Key Exchanges (KEXs) und Cipher Suites) einschränken. Im Folgenden finden Sie eine Liste der unterstützten kryptografischen Algorithmen für jede Sicherheitsrichtlinie für den SFTP-Connector.

> **Note**
>
> `TransferSFTPConnectorSecurityPolicy-2024-03`ist die Standardsicherheitsrichtlinie, die auf SFTP-Konnektoren angewendet wird.

Sie können die Sicherheitsrichtlinie für Ihren Connector ändern. Wählen Sie im linken Navigationsbereich der Transfer Family die Option Connectors und wählen Sie Ihren Connector

aus. Wählen Sie dann im Abschnitt SFTP-Konfiguration die Option Bearbeiten aus. Wählen Sie im Abschnitt Optionen für kryptografische Algorithmen eine beliebige verfügbare Sicherheitsrichtlinie aus der Dropdownliste im Feld Sicherheitsrichtlinie aus.

| Sicherheitsrichtlinie | FTP-Richtlinie für Übertragungen-2024-03 Connector Security | ConnectorSecurityFTP-Richtlinie für Übertragungen-2023-07 |
|---|---|---|
| Ciphers | | |
| aes128-ctr | | ♦ |
| aes128-gcm@openssh.com | ♦ | ♦ |
| aes192-ctr | ♦ | ♦ |
| aes256-ctr | ♦ | ♦ |
| aes256-gcm@openssh.com | ♦ | ♦ |
| Kexs | | |
| Kurve 25519-sha256 | ♦ | ♦ |
| curve25519-sha256@libssh.org | ♦ | ♦ |
| diffie-hellman-gruppe14-sha1 | | ♦ |
| diffie-hellman-gruppe-16-sha512 | ♦ | ♦ |
| diffie-hellman-gruppe-18-sha512 | ♦ | ♦ |
| Diffie-Hellman-Gruppenaustausch-SHA256 | ♦ | ♦ |
| Macs | | |

| Sicherheitsrichtlinie | FTP-Richtlinie für Übertragungen-2024-03 Connector Security | ConnectorSecurityFTP-Richtlinie für Übertragungen-2023-07 |
|---|---|---|
| hmac-sha2-512-etm@openssh.com | ♦ | ♦ |
| hmac-sha2-256-etm@openssh.com | ♦ | ♦ |
| hmac-sha2-512 | ♦ | ♦ |
| hmac-sha2-256 | ♦ | ♦ |
| hmac-sha1 | | ♦ |
| hmac-sha-196 | | ♦ |
| Host Key Algorithms | | |
| rsa-sha2-256 | ♦ | ♦ |
| rsa-sha2-512 | ♦ | ♦ |
| ecdsa-sha2-nistp256 | ♦ | ♦ |
| ecdsa-sha2-nistp384 | ♦ | ♦ |
| ecdsa-sha2-nistp521 | ♦ | ♦ |
| ssh-rsa | | ♦ |

# Verwenden Sie den hybriden Schlüsselaustausch nach dem Quantum-Verfahren mit AWS Transfer Family

AWS Transfer Family unterstützt für das Secure Shell (SSH) -Protokoll eine hybride Option zur Einrichtung eines Schlüssels nach der Installation eines Quantenschlüssels. Die Einrichtung von Quantenschlüsseln ist erforderlich, da es bereits möglich ist, den Netzwerkverkehr aufzuzeichnen

und für die future Entschlüsselung durch einen Quantencomputer zu speichern, was als Store-Now-Harvest-Later-Angriff bezeichnet wird.

Sie können diese Option verwenden, wenn Sie eine Verbindung zu Transfer Family herstellen, um Dateien sicher in und aus dem Amazon Simple Storage Service (Amazon S3) -Speicher oder Amazon Elastic File System (Amazon EFS) zu übertragen. Die Einrichtung von hybriden Schlüsseln nach dem Quantenprozess in SSH führt Mechanismen zur Einrichtung von Schlüsseln ein, die nach dem Quantenverfahren eingerichtet wurden. Diese werden in Verbindung mit klassischen Schlüsselaustauschalgorithmen verwendet. SSH-Schlüssel, die mit klassischen Chiffrier-Suites erstellt wurden, sind mit der aktuellen Technologie vor Brute-Force-Angriffen geschützt. Es wird jedoch nicht erwartet, dass die klassische Verschlüsselung nach dem Aufkommen von Quantencomputern in future sicher bleibt.

Wenn Ihr Unternehmen auf die langfristige Vertraulichkeit von Daten angewiesen ist, die über eine Transfer Family Family-Verbindung übertragen werden, sollten Sie einen Plan zur Umstellung auf Post-Quanten-Kryptografie in Betracht ziehen, bevor große Quantencomputer für den Einsatz verfügbar sind.

Um heute verschlüsselte Daten vor möglichen future Angriffen zu schützen, beteiligt AWS sich die Kryptografie-Community an der Entwicklung quantenresistenter oder Post-Quanten-Algorithmen. Wir haben in Transfer Family hybride Verschlüsselungssuiten für den Schlüsselaustausch nach dem Quantenaustausch implementiert, die klassische Elemente und Post-Quanten-Elemente kombinieren.

Diese hybriden Verschlüsselungssammlungen sind in den meisten Regionen für den Einsatz auf Ihren Produktions-Workloads verfügbar. AWS Da sich die Leistungsmerkmale und Bandbreitenanforderungen von Hybrid-Cipher Suites jedoch von denen klassischer Schlüsselaustauschmechanismen unterscheiden, empfehlen wir, sie auf Ihren Transfer Family Family-Verbindungen zu testen.

Weitere Informationen zur Post-Quanten-Kryptografie finden Sie im Sicherheits-Blogbeitrag Post-Quantum Cryptography.

Inhalt


- Informationen zum hybriden Schlüsselaustausch in SSH nach der Quantenzeit
- So funktioniert die Etablierung hybrider Schlüssel nach der Quantenentwicklung in Transfer Family
  - Warum Kyber?
  - Hybrider SSH-Schlüsselaustausch und kryptografische Anforderungen (FIPS 140)
- Testen des hybriden Schlüsselaustauschs nach dem Quantenprozess in der Transfer Family

- Aktivieren Sie den hybriden Schlüsselaustausch nach dem Quantum-Verfahren auf Ihrem SFTP-Endpunkt
- Richten Sie einen SFTP-Client ein, der den hybriden Schlüsselaustausch nach dem Quantum-Verfahren unterstützt
- Bestätigen Sie den Hybrid-Schlüsselaustausch nach dem Quantum-Verfahren in SFTP

## Informationen zum hybriden Schlüsselaustausch in SSH nach der Quantenzeit

Transfer Family unterstützt Post-Quantum-Hybrid-Schlüsselaustausch-Verschlüsselungssuiten, die sowohl den klassischen Elliptic Curve Diffie-Hellman (ECDH) -Schlüsselaustauschalgorithmus als auch CRYSTALS Kyber verwenden. Kyber ist ein Public-Key-Verschlüsselungs- und Schlüsseletablierungsalgorithmus, den das National Institute for Standards and Technology (NIST) als seinen ersten Standardalgorithmus nach der Quantenschlüsselvereinbarung bezeichnet hat.

Der Client und der Server führen immer noch einen ECDH-Schlüsselaustausch durch. Darüber hinaus kapselt der Server einen gemeinsamen geheimen Schlüssel für die Zeit nach dem Quantum in den öffentlichen KEM-Schlüssel des Clients ein, der in der SSH-Schlüsselaustauschnachricht des Clients angekündigt wird. Diese Strategie kombiniert die hohe Sicherheit eines klassischen Schlüsselaustauschs mit der Sicherheit des geplanten Schlüsselaustauschs nach dem Quantenaustausch, um sicherzustellen, dass die Handshakes geschützt sind, solange das ECDH oder das gemeinsame geheime Post-Quantum-Geheimnis nicht geknackt werden können.

## So funktioniert die Etablierung hybrider Schlüssel nach der Quantenentwicklung in Transfer Family

AWS hat kürzlich die Unterstützung für den Austausch von Schlüsseln nach dem Quantenzugriff bei SFTP-Dateiübertragungen in angekündigt. AWS Transfer Family Transfer Family skaliert business-to-business Dateiübertragungen an AWS Speicherdienste mithilfe von SFTP und anderen Protokollen sicher. SFTP ist eine sicherere Version des File Transfer Protocol (FTP), das über SSH läuft. Die Unterstützung von Transfer Family nach dem Quanten-Schlüsselaustausch legt die Sicherheitslatte für Datenübertragungen über SFTP höher.

Die SFTP-Unterstützung für den Post-Quanten-Hybrid-Schlüsselaustausch in der Transfer Family umfasst die Kombination der Post-Quanten-Algorithmen Kyber-512, Kyber-768 und Kyber-1024 mit ECDH über P256-, P384-, P521- oder Curve25519-Kurven. Die folgenden entsprechenden Methoden

für den SSH-Schlüsselaustausch sind im Entwurf für den Austausch von SSH-Schlüsseln nach dem Quanten-Hybrid-Schlüsselaustausch spezifiziert.

- `ecdh-nistp256-kyber-512r3-sha256-d00@openquantumsafe.org`
- `ecdh-nistp384-kyber-768r3-sha384-d00@openquantumsafe.org`
- `ecdh-nistp521-kyber-1024r3-sha512-d00@openquantumsafe.org`
- `x25519-kyber-512r3-sha256-d00@amazon.com`

> **Note**
>
> Diese neuen Methoden für den Schlüsselaustausch können sich ändern, wenn sich der Entwurf in Richtung Standardisierung weiterentwickelt oder wenn NIST den Kyber-Algorithmus ratifiziert.

## Warum Kyber?

AWS engagiert sich für die Unterstützung standardisierter, interoperabler Algorithmen. Kyber ist der erste Post-Quantum-Verschlüsselungsalgorithmus, der vom NIST-Projekt Post-Quantum Cryptography für die Standardisierung ausgewählt wurde. Einige Normungsgremien integrieren Kyber bereits in Protokolle. AWS unterstützt Kyber bereits in TLS auf einigen AWS API-Endpunkten.

Im Rahmen dieser Verpflichtung AWS hat das Unternehmen der IETF einen Vorschlagsentwurf für Post-Quanten-Kryptografie vorgelegt, der Kyber mit NIST-zugelassenen Kurven wie P256 für SSH kombiniert. Um die Sicherheit für unsere Kunden zu verbessern, folgt die AWS Implementierung des Post-Quantum-Schlüsselaustauschs in SFTP und SSH diesem Entwurf. Wir planen, future Aktualisierungen zu unterstützen, bis unser Vorschlag von der IETF angenommen wird und ein Standard wird.

Die neuen Methoden für den Schlüsselaustausch (im Abschnitt aufgeführtSo funktioniert die Etablierung hybrider Schlüssel nach der Quantenentwicklung in Transfer Family) könnten sich ändern, wenn sich der Entwurf in Richtung Standardisierung weiterentwickelt oder wenn NIST den Kyber-Algorithmus ratifiziert.

> **Note**
>
> Unterstützung für Post-Quantum-Algorithmen ist derzeit für den Austausch von Hybrid-Schlüsseln in TLS für AWS KMS (siehe Hybrid-Post-Quantum-TLS verwenden mit), und API-Endpunkten verfügbar. AWS KMSAWS Certificate Manager AWS Secrets Manager

### Hybrider SSH-Schlüsselaustausch und kryptografische Anforderungen (FIPS 140)

Für Kunden, die FIPS-Konformität benötigen, bietet Transfer Family FIPS-zertifizierte Kryptografie in SSH mithilfe der AWS FIPS 140-zertifizierten Open-Source-Kryptografiebibliothek -LC. AWSDie im TransferSecurityPolicy -PQ-SSH-FIPS-Experimental-2023-04 in Transfer Family unterstützten Post-Quantum-Hybrid-Schlüsselaustauschmethoden sind gemäß SP 800-56Cr2 von NIST (Abschnitt 2) FIPS-zugelassen. Das deutsche Bundesamt für Sicherheit in der Informationstechnik (BSI) und die französische Agence nationale de la sécurité des systèmes d'Information (ANSSI) empfehlen ebenfalls solche Methoden für den hybriden Schlüsselaustausch nach dem Quantenverfahren.

## Testen des hybriden Schlüsselaustauschs nach dem Quantenprozess in der Transfer Family

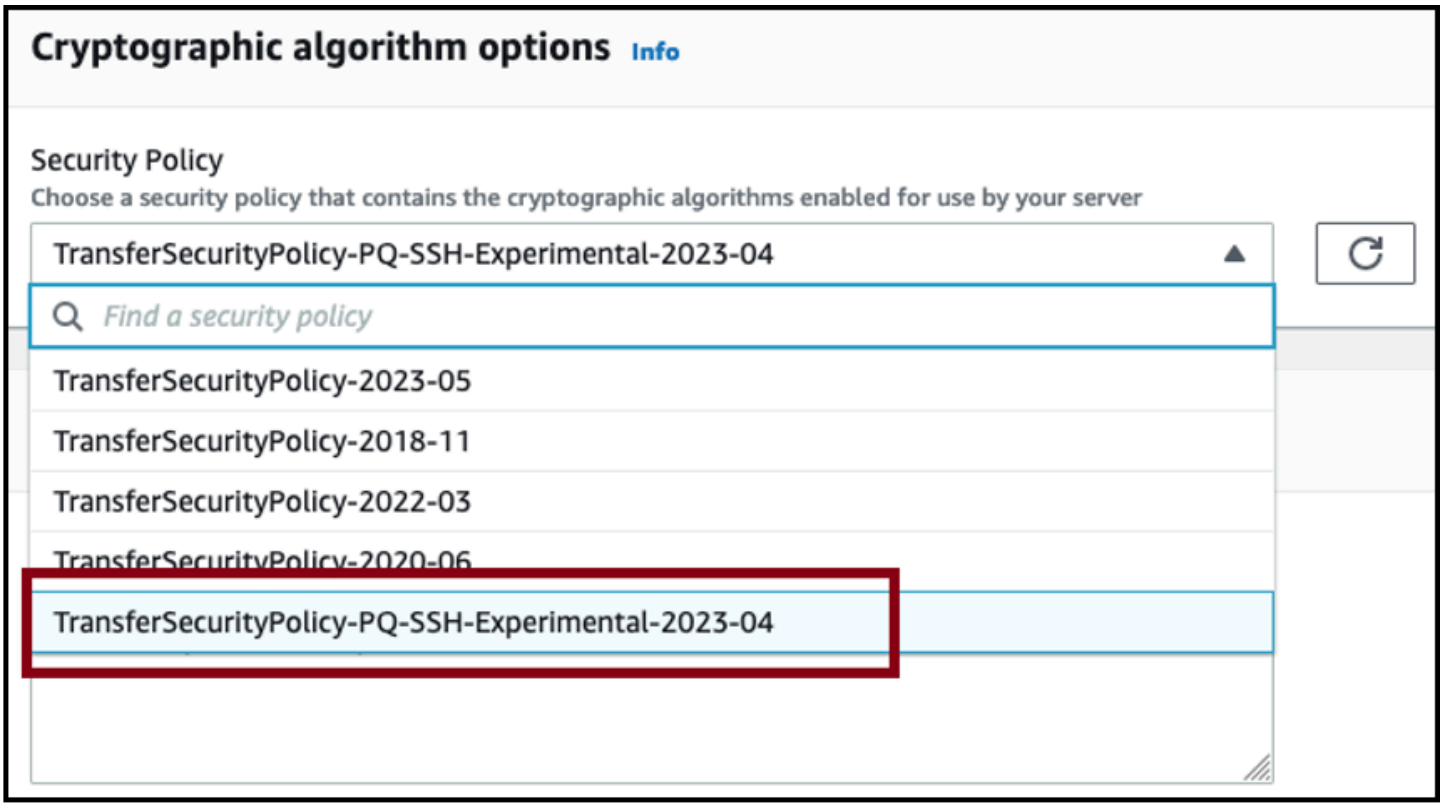
In diesem Abschnitt werden die Schritte beschrieben, die Sie unternehmen, um den hybriden Schlüsselaustausch nach dem Quantenverfahren zu testen.

1. Aktivieren Sie den hybriden Schlüsselaustausch nach dem Quantum-Verfahren auf Ihrem SFTP-Endpunkt.
2. Verwenden Sie einen SFTP-Client (z. B.Richten Sie einen SFTP-Client ein, der den hybriden Schlüsselaustausch nach dem Quantum-Verfahren unterstützt), der den hybriden Schlüsselaustausch nach dem Quantenverfahren unterstützt, und folgen Sie dabei den Anweisungen im oben genannten Spezifikationsentwurf.
3. Übertragen Sie eine Datei mit einem Transfer Family Family-Server.
4. Bestätigen Sie den Hybrid-Schlüsselaustausch nach dem Quantum-Verfahren in SFTP.

### Aktivieren Sie den hybriden Schlüsselaustausch nach dem Quantum-Verfahren auf Ihrem SFTP-Endpunkt

Sie können die SSH-Richtlinie auswählen, wenn Sie einen neuen SFTP-Serverendpunkt in Transfer Family erstellen, oder indem Sie die Optionen für den kryptografischen Algorithmus in einem

vorhandenen SFTP-Endpunkt bearbeiten. Der folgende Snapshot zeigt ein Beispiel dafür, AWS Management Console wo Sie die SSH-Richtlinie aktualisieren.




Die SSH-Richtliniennamen, die den Schlüsselaustausch nach dem Quantum unterstützen, lauten Policy-PQ-SSH-Experimental-2023-04 und TransferSecurity Policy-PQ-SSH-FIPS-Experimental-2023-04. TransferSecurity Weitere Informationen zu den Richtlinien von Transfer Family finden Sie unterSicherheitsrichtlinien für AWS Transfer Family Server.

## Richten Sie einen SFTP-Client ein, der den hybriden Schlüsselaustausch nach dem Quantum-Verfahren unterstützt

Nachdem Sie die richtige Post-Quantum-SSH-Richtlinie in Ihrem SFTP Transfer Family-Endpunkt ausgewählt haben, können Sie in Transfer Family mit Post-Quantum-SFTP experimentieren. Sie können einen SFTP-Client (wie OQS OpenSSH) verwenden, der den Post-Quantum-Hybrid-Schlüsselaustausch unterstützt, indem Sie die Anweisungen im oben genannten Spezifikationsentwurf befolgen.

OQS OpenSSH ist ein Open-Source-Fork von OpenSSH, der SSH mithilfe von quantensicherer Kryptografie erweitert. `liboqs` `liboqs`ist eine Open-Source-C-Bibliothek, die quantenresistente kryptografische Algorithmen implementiert. OQS OpenSSH und `liboqs` sind Teil des Open Quantum Safe (OQS) -Projekts.

Um den Post-Quantum-Hybrid-Schlüsselaustausch in Transfer Family SFTP mit OQS OpenSSH zu testen, müssen Sie OQS OpenSSH erstellen, wie in der README-Datei des Projekts beschrieben. Nachdem Sie OQS OpenSSH erstellt haben, können Sie den Beispiel-SFTP-Client ausführen, um eine Verbindung zu Ihrem SFTP-Endpunkt herzustellen (z. B.`s-1111aaaa2222bbbb3.server.transfer.us-west-2.amazonaws.com`), indem Sie die Post-Quantum-Hybrid-Schlüsselaustauschmethoden verwenden, wie im folgenden Befehl gezeigt.

```
./sftp -S ./ssh -v -o \
   KexAlgorithms=ecdh-nistp384-kyber-768r3-sha384-d00@openquantumsafe.org \
   -i username_private_key_PEM_file \
   username@server-id.server.transfer.region-id.amazonaws.com
```

Ersetzen Sie im vorherigen Befehl die folgenden Elemente durch Ihre eigenen Informationen:

- Ersetzen Sie *Username_Private_Key_PEM_File durch die PEM-codierte Datei mit dem privaten Schlüssel des SFTP-Benutzers*
- Ersetzen *Sie den Benutzernamen durch den* SFTP-Benutzernamen
- Ersetzen Sie die *Server-ID* durch die Transfer Family Family-Server-ID
- Ersetzen Sie die *Region-ID* durch die tatsächliche Region, in der sich Ihr Transfer Family Family-Server befindet

## Bestätigen Sie den Hybrid-Schlüsselaustausch nach dem Quantum-Verfahren in SFTP

Überprüfen Sie die Client-Ausgabe, um sicherzustellen, dass der hybride Schlüsselaustausch nach dem Quantenverfahren während einer SSH-Verbindung für SFTP zu Transfer Family verwendet wurde. Optional können Sie ein Programm zur Paketerfassung verwenden. Wenn Sie den Open Quantum Safe OpenSSH-Client verwenden, sollte die Ausgabe etwa wie folgt aussehen (wobei der Kürze halber irrelevante Informationen weggelassen werden):

```
$./sftp -S ./ssh -v -o KexAlgorithms=ecdh-nistp384-kyber-768r3-sha384-
d00@openquantumsafe.org -
i username_private_key_PEM_file username@s-1111aaaa2222bbbb3.server.transfer.us-
west-2.amazonaws.com
OpenSSH_8.9-2022-01_p1, Open Quantum Safe 2022-08, OpenSSL 3.0.2 15 Mar 2022
debug1: Reading configuration data /home/lab/openssh/oqs-test/tmp/ssh_config
debug1: Authenticator provider $SSH_SK_PROVIDER did not resolve; disabling
debug1: Connecting to s-1111aaaa2222bbbb3.server.transfer.us-west-2.amazonaws.com
 [xx.yy.zz..12] port 22.
debug1: Connection established.
```

```
[...]
debug1: Local version string SSH-2.0-OpenSSH_8.9-2022-01_
debug1: Remote protocol version 2.0, remote software version AWS_SFTP_1.1
debug1: compat_banner: no match: AWS_SFTP_1.1
debug1: Authenticating to s-1111aaaa2222bbbb3.server.transfer.us-
west-2.amazonaws.com:22 as 'username'
debug1: load_hostkeys: fopen /home/lab/.ssh/known_hosts2: No such file or directory
[...]
debug1: SSH2_MSG_KEXINIT sent
debug1: SSH2_MSG_KEXINIT received
debug1: kex: algorithm: ecdh-nistp384-kyber-768r3-sha384-d00@openquantumsafe.org
debug1: kex: host key algorithm: ssh-ed25519
debug1: kex: server->client cipher: aes192-ctr MAC: hmac-sha2-256-etm@openssh.com
 compression: none
debug1: kex: client->server cipher: aes192-ctr MAC: hmac-sha2-256-etm@openssh.com
 compression: none
debug1: expecting SSH2_MSG_KEX_ECDH_REPLY
debug1: SSH2_MSG_KEX_ECDH_REPLY received
debug1: Server host key: ssh-ed25519 SHA256:e3b0c44298fc1c149afbf4c8996fb92427ae41e4649
[...]
debug1: rekey out after 4294967296 blocks
debug1: SSH2_MSG_NEWKEYS sent
debug1: expecting SSH2_MSG_NEWKEYS
debug1: SSH2_MSG_NEWKEYS received
debug1: rekey in after 4294967296 blocks
[...]
Authenticated to AWS.Tranfer.PQ.SFTP.test-endpoint.aws.com ([xx.yy.zz..12]:22) using
 "publickey".s
debug1: channel 0: new [client-session]
[...]
Connected to s-1111aaaa2222bbbb3.server.transfer.us-west-2.amazonaws.com.
sftp>
```

Die Ausgabe zeigt, dass die Kundenverhandlung mithilfe der `ecdh-nistp384-kyber-768r3-sha384-d00@openquantumsafe.org` Post-Quantum-Hybridmethode stattgefunden hat und erfolgreich eine SFTP-Sitzung eingerichtet wurde.

# Datenschutz in AWS Transfer Family

Das AWS Modell der mit gilt für den Datenschutz in AWS Transfer Family (Transfer Family). Wie in diesem Modell beschrieben, AWS ist es für den Schutz der globalen Infrastruktur verantwortlich, auf der die gesamte AWS Cloud betrieben wird. Sie sind dafür verantwortlich, die Kontrolle über Ihre in

dieser Infrastruktur gehosteten Inhalte zu behalten. Dieser Inhalt umfasst die Sicherheitskonfiguration und die Verwaltungsaufgaben für die AWS Dienste, die Sie verwenden. Weitere Informationen zum Datenschutz finden Sie unter Häufig gestellte Fragen zum Datenschutz. Informationen zum Datenschutz in Europa finden Sie im AWS Sicherheitsblog im Modell der AWS gemeinsamen Verantwortung und im DSGVO-Blogbeitrag.

Aus Datenschutzgründen empfehlen wir Ihnen, Ihre AWS Kontoanmeldeinformationen zu schützen und individuelle Benutzerkonten mit AWS Identity and Access Management (IAM) einzurichten. So erhält jeder Benutzer nur die Berechtigungen, die zum Durchführen seiner Aufgaben erforderlich sind. Außerdem sollten Sie die Daten mit folgenden Methoden schützen:

- Verwenden Sie für jedes Konto die Multi-Faktor-Authentifizierung (MFA).
- Verwenden Sie SSL/TLS, um mit Ressourcen zu kommunizieren. AWS Wir unterstützen TLS 1.2.
- Richten Sie die API und die Protokollierung von Benutzeraktivitäten mit ein AWS CloudTrail.
- Verwenden Sie AWS Verschlüsselungslösungen zusammen mit allen Standardsicherheitskontrollen innerhalb der AWS Dienste.
- Verwenden Sie erweiterte verwaltete Sicherheitsservices wie Amazon Macie, die dabei helfen, in Amazon S3 gespeicherte persönliche Daten zu erkennen und zu sichern.
- Wenn Sie für den Zugriff auf AWS über eine Befehlszeilenschnittstelle oder über 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 wie z. B. im Feld Name keine sensiblen, identifizierenden Informationen wie Kontonummern von Kunden einzugeben. Dies gilt auch, wenn Sie mit Transfer Family oder anderen AWS Diensten über die Konsole AWS CLI, API oder AWS SDKs arbeiten. Alle Konfigurationsdaten, die Sie in die Transfer Family Family-Dienstkonfiguration oder in die Konfigurationen anderer Dienste eingeben, werden möglicherweise zur Aufnahme in Diagnoseprotokolle aufgenommen. Wenn Sie eine URL für einen externen Server bereitstellen, schließen Sie keine Anmeldeinformationen zur Validierung Ihrer Anforderung an den betreffenden Server in die URL ein.

Im Gegensatz dazu werden Daten aus Upload- und Download-Vorgängen zu und von Transfer Family Family-Servern als vollständig privat behandelt und existieren nie außerhalb verschlüsselter Kanäle, wie z. B. einer SFTP- oder FTPS-Verbindung. Diese Daten sind immer nur autorisierten Personen zugänglich.

Themen


- Datenverschlüsselung in Amazon S3
- Verwaltung von SSH- und PGP-Schlüsseln in Transfer Family


## Datenverschlüsselung in Amazon S3

AWS Transfer Family verwendet die Standardverschlüsselungsoptionen, die Sie für Ihren Amazon S3 S3-Bucket festgelegt haben, um Ihre Daten zu verschlüsseln. Wenn Sie die Standardverschlüsselung für einen Bucket aktivieren, werden alle Objekte verschlüsselt, wenn sie im Bucket gespeichert werden. Die Objekte werden mithilfe serverseitiger Verschlüsselung entweder mit von Amazon S3 verwalteten Schlüsseln (SSE-S3) oder () verwalteten Schlüsseln AWS Key Management Service (SSE-KMS AWS KMS) verschlüsselt. Informationen zur serverseitigen Verschlüsselung finden Sie unter Schützen von Daten mithilfe serverseitiger Verschlüsselung im Amazon Simple Storage Service-Benutzerhandbuch.

Die folgenden Schritte zeigen Ihnen, wie Sie Daten in verschlüsseln. AWS Transfer Family

Um die Verschlüsselung in zuzulassen AWS Transfer Family

1. Aktivieren Sie die Standardverschlüsselung für Ihren Amazon S3 S3-Bucket. Anweisungen finden Sie unter Amazon S3 S3-Standardverschlüsselung für S3-Buckets im Amazon Simple Storage Service-Benutzerhandbuch.
2. Aktualisieren Sie die AWS Identity and Access Management (IAM) -Rollenrichtlinie, die dem Benutzer zugewiesen ist, um die erforderlichen AWS Key Management Service (AWS KMS) Berechtigungen zu gewähren.
3. Wenn Sie eine Sitzungsrichtlinie für den Benutzer verwenden, muss die Sitzungsrichtlinie die erforderlichen AWS KMS Berechtigungen gewähren.

Das folgende Beispiel zeigt eine IAM-Richtlinie, die die Mindestberechtigungen gewährt, die bei der Verwendung AWS Transfer Family mit einem Amazon S3 S3-Bucket erforderlich sind, der für die AWS KMS Verschlüsselung aktiviert ist. Nehmen Sie diese Beispielrichtlinie sowohl in die Benutzer-IAM-Rollenrichtlinie als auch in die Sitzungsrichtlinie auf, falls Sie eine verwenden.

```
{
 "Sid": "Stmt1544140969635",
 "Action": [
  "kms:Decrypt",
```

```
  "kms:Encrypt",
  "kms:GenerateDataKey"
 ],
 "Effect": "Allow",
 "Resource": "arn:aws:kms:region:account-id:key/kms-key-id"
}
```

> **Note**
>
> Die KMS-Schlüssel-ID, die Sie in dieser Richtlinie angeben, muss mit der in Schritt 1 für die Standardverschlüsselung angegebenen identisch sein.
> Root oder die IAM-Rolle, die für den Benutzer verwendet wird, muss in der AWS KMS Schlüsselrichtlinie zulässig sein. Informationen zur AWS KMS Schlüsselrichtlinie finden Sie unter Verwenden von Schlüsselrichtlinien in AWS KMS im AWS Key Management Service Entwicklerhandbuch.

## Verwaltung von SSH- und PGP-Schlüsseln in Transfer Family

In diesem Abschnitt finden Sie Informationen zu SSH-Schlüsseln, einschließlich deren Generierung und Rotation. Einzelheiten zur Verwendung von Transfer Family with AWS Lambda zur Schlüsselverwaltung finden Sie im Blogbeitrag Aktivieren der Self-Service-Schlüsselverwaltung für Benutzer mit A AWS Transfer Family und AWS Lambda.

> **Note**
>
> AWS Transfer Family akzeptiert RSA-, ECDSA- und ED25519-Schlüssel.

In diesem Abschnitt wird auch beschrieben, wie Sie Pretty Good Privacy (PGP) -Schlüssel generieren und verwalten.

Themen

- Unterstützte Algorithmen für Benutzer- und Serverschlüssel
- Generieren Sie SSH-Schlüssel für vom Service verwaltete Benutzer
- Drehen Sie die SSH-Schlüssel
- Generieren und verwalten Sie PGP-Schlüssel
- Unterstützte PGP-Clients

## Unterstützte Algorithmen für Benutzer- und Serverschlüssel

Die folgenden Schlüsselalgorithmen werden für Benutzer- und Serverschlüsselpaare innerhalb unterstützt. AWS Transfer Family

> **Note**
>
> Informationen zu Algorithmen, die zusammen mit der PGP-Entschlüsselung in Workflows verwendet werden können, finden Sie unter Unterstützte Algorithmen für PGP-Schlüsselpaare.

- Für ED25519: `ssh-ed25519`
- Für RSA:
  - `rsa-sha2-256`
  - `rsa-sha2-512`
- Für ECDSA:
  - `ecdsa-sha2-nistp256`
  - `ecdsa-sha2-nistp384`
  - `ecdsa-sha2-nistp521`

> **Note**
>
> Wir unterstützen unsere älteren Sicherheitsrichtlinien `ssh-rsa` mit SHA1. Details hierzu finden Sie unter Kryptografische Algorithmen.

## Generieren Sie SSH-Schlüssel für vom Service verwaltete Benutzer

Sie können Ihren Server so einrichten, dass Benutzer mithilfe der vom Service verwalteten Authentifizierungsmethode authentifiziert werden, bei der Benutzernamen und SSH-Schlüssel innerhalb des Dienstes gespeichert werden. Der öffentliche SSH-Schlüssel des Benutzers wird als Eigentum des Benutzers auf den Server hochgeladen. Dieser Schlüssel wird vom Server als Teil eines standardmäßigen schlüsselbasierten Authentifizierungsprozesses verwendet. Jeder Benutzer kann mehrere öffentliche SSH-Schlüssel bei einem einzelnen Server registrieren. Informationen

zur Begrenzung der Anzahl der Schlüssel, die pro Benutzer gespeichert werden können, finden Sie unter AWS Transfer Family Endpunkte und Kontingente in der. Allgemeine Amazon Web Services-Referenz

Als Alternative zur vom Dienst verwalteten Authentifizierungsmethode können Sie Benutzer mithilfe eines benutzerdefinierten Identitätsanbieters authentifizieren, oder. AWS Directory Service for Microsoft Active Directory Weitere Informationen finden Sie unter Mit Anbietern benutzerdefinierter Identitäten arbeiten oder Verwenden des AWS Directory Service-Identitätsanbieters.

Ein Server kann Benutzer nur mit einer Methode authentifizieren (vom Dienst verwaltet, Verzeichnisdienst oder benutzerdefinierter Identitätsanbieter), und diese Methode kann nicht geändert werden, nachdem der Server erstellt wurde.

Themen

- SSH-Schlüssel auf macOS, Linux oder Unix erstellen
- SSH-Schlüssel unter Microsoft Windows erstellen
- Konvertiert einen öffentlichen SSH2-Schlüssel in das PEM-Format

## SSH-Schlüssel auf macOS, Linux oder Unix erstellen

Auf den Betriebssystemen macOS, Linux oder Unix verwenden Sie den `ssh-keygen` Befehl, um einen öffentlichen SSH-Schlüssel und einen privaten SSH-Schlüssel zu erstellen, die auch als key pair bezeichnet werden.

Um SSH-Schlüssel auf einem macOS-, Linux- oder Unix-Betriebssystem zu erstellen

1. Öffnen Sie auf macOS-, Linux- oder Unix-Betriebssystemen ein Befehlsterminal.
2. AWS Transfer Family akzeptiert Schlüssel im RSA-, ECDSA- und ED25519-Format. Wählen Sie je nach Art des Schlüsselpaars, das Sie generieren, den entsprechenden Befehl aus.

> **Note**
>
> In den folgenden Beispielen geben wir keine Passphrase an: In diesem Fall fordert das Tool Sie auf, Ihre Passphrase einzugeben und sie dann zur Überprüfung zu wiederholen. Die Erstellung einer Passphrase bietet einen besseren Schutz für Ihren privaten Schlüssel und kann auch die allgemeine Systemsicherheit verbessern. Sie können Ihre Passphrase nicht wiederherstellen: Wenn Sie sie vergessen, müssen Sie einen neuen Schlüssel erstellen.

Wenn Sie jedoch einen Server-Hostschlüssel generieren, müssen Sie eine leere Passphrase angeben, indem Sie die `-N ""` Option im Befehl angeben (oder indem Sie **`Enter`** zweimal drücken, wenn Sie dazu aufgefordert werden), da Transfer Family Family-Server beim Start kein Passwort anfordern können.

- Um ein RSA 4096-Bit-Schlüsselpaar zu generieren:

```
ssh-keygen -t rsa -b 4096 -f key_name
```

- Um ein 521-Bit-ECDSA-Schlüsselpaar zu generieren (ECDSA hat Bitgrößen von 256, 384 und 521):

```
ssh-keygen -t ecdsa -b 521 -f key_name
```

- Um ein ED25519-Schlüsselpaar zu generieren:

```
ssh-keygen -t ed25519 -f key_name
```

**Note**

*key_name*ist der Dateiname des SSH-Schlüsselpaars.

Das Folgende zeigt ein Beispiel für die `ssh-keygen` Ausgabe.

```
ssh-keygen -t rsa -b 4096 -f key_name
Generating public/private rsa key pair.

Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in key_name.
Your public key has been saved in key_name.pub.
The key fingerprint is:
SHA256:8tDDwPmanTFcEzjTwPGETVWOGW1nVz+gtCCE8hL7PrQ bob.amazon.com
The key's randomart image is:
+---[RSA 4096]----+
|    . ....E      |
| .   = ...       |
```

```
|. . . = ..o      |
| . o +  oo =     |
|  + =  .S.= *    |
| . o o ..B + o   |
|     .o.+.* .    |
|     =o*+*.      |
|    ..*o*+.      |
+----[SHA256]-----+
```

> **Note**
>
> Wenn Sie den Befehl `ssh-keygen` ausführen (siehe oben), erstellt er die öffentlichen und privaten Schlüssel als Dateien im aktuellen Verzeichnis.

Ihr SSH-Schlüsselpaar ist jetzt einsatzbereit. Folgen Sie den Schritten 3 und 4, um den öffentlichen SSH-Schlüssel für Ihre vom Service verwalteten Benutzer zu speichern. Diese Benutzer verwenden die Schlüssel, wenn sie Dateien auf Transfer Family Family-Serverendpunkten übertragen.

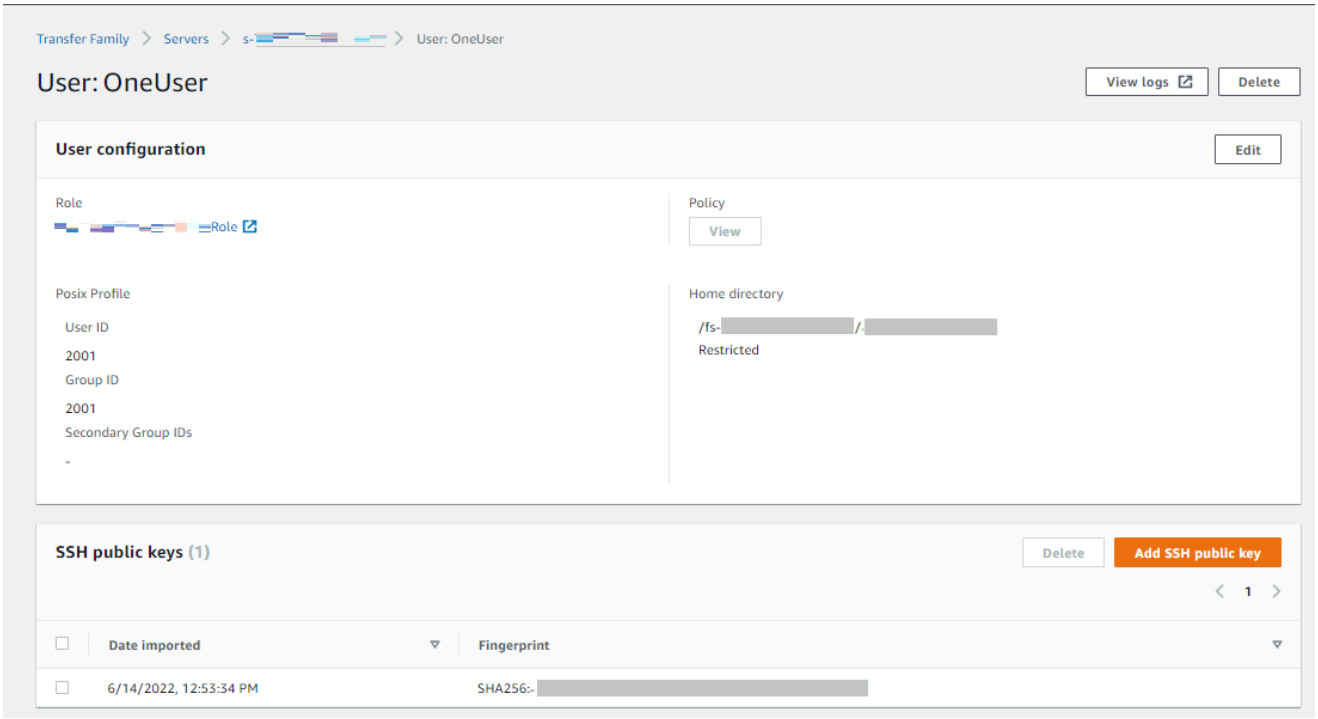
3. Navigieren Sie zu der *`key_name`*`.pub` Datei und öffnen Sie sie.
4. Kopieren Sie den Text und fügen Sie ihn in den öffentlichen SSH-Schlüssel für den vom Service verwalteten Benutzer ein.

   a. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/ und wählen Sie dann im Navigationsbereich Server aus.

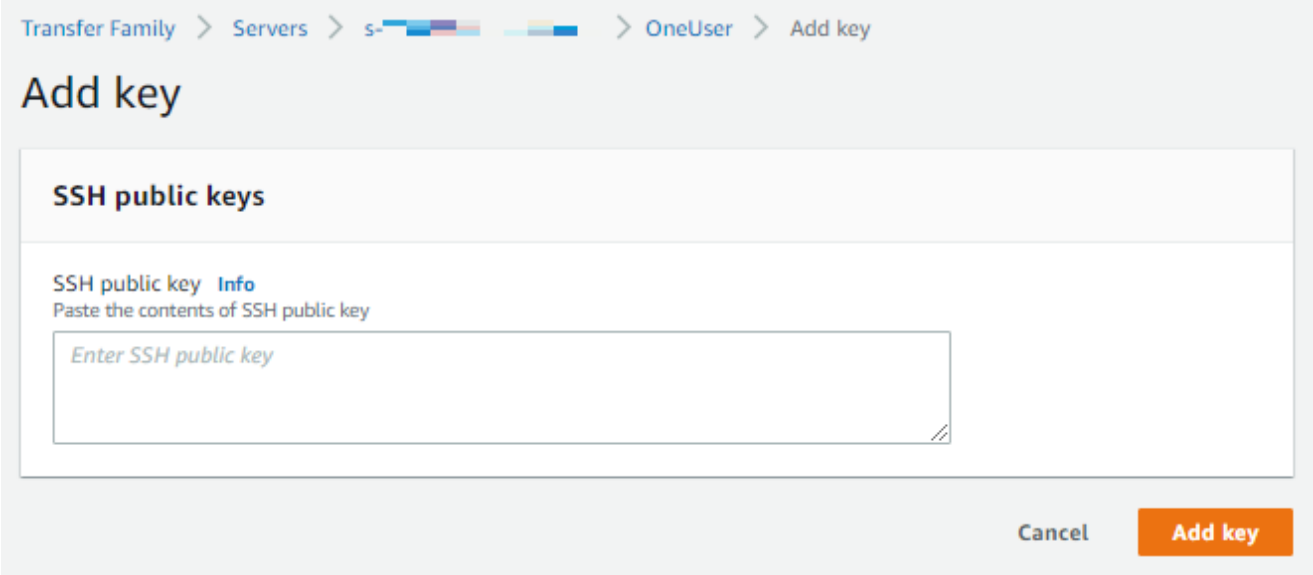
   b. Wählen Sie auf der Seite Server die Server-ID für den Server aus, der den Benutzer enthält, den Sie aktualisieren möchten.

   c. Wählen Sie den Benutzer aus, für den Sie einen öffentlichen Schlüssel hinzufügen möchten.

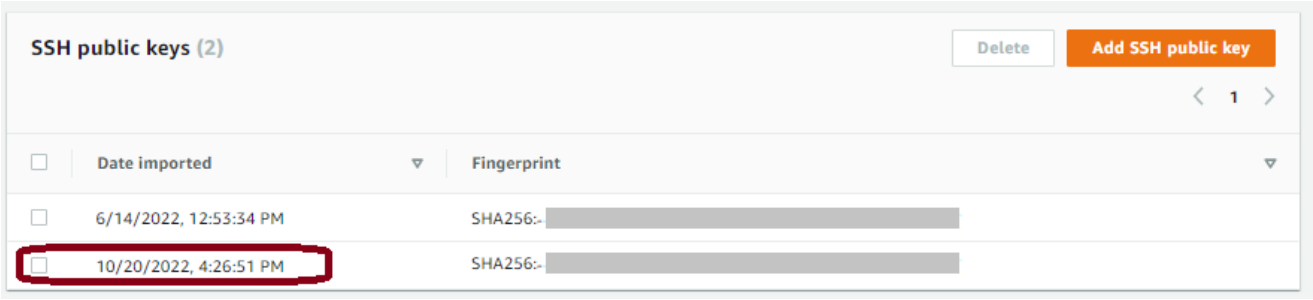
   d. Wählen Sie im Bereich Öffentliche SSH-Schlüssel die Option Öffentlichen SSH-Schlüssel hinzufügen aus.




e. Fügen Sie den Text des öffentlichen Schlüssels, den Sie generiert haben, in das Textfeld für den öffentlichen SSH-Schlüssel ein und wählen Sie dann Schlüssel hinzufügen aus.




Der neue Schlüssel ist im Bereich mit öffentlichen SSH-Schlüsseln aufgeführt.




SSH-Schlüssel unter Microsoft Windows erstellen

Windows verwendet ein etwas anderes SSH-Schlüsselpaarformat. Der öffentliche Schlüssel muss im PUB-Format und der private Schlüssel im PPK-Format vorliegen. Unter Windows können Sie PuTTYgen verwenden, um ein SSH-Schlüsselpaar in den entsprechenden Formaten zu erstellen. Sie können PuTTYgen auch verwenden, um einen mit `ssh-keygen` erstellten privaten Schlüssel in eine `.ppk`-Datei zu konvertieren.

> **Note**
>
> Wenn Sie WinSCP eine private Schlüsseldatei präsentieren, die nicht im `.ppk` Format ist, bietet dieser Client an, den Schlüssel für Sie in ein `.ppk` Format zu konvertieren.

Ein Tutorial zum Erstellen von SSH-Schlüsseln mithilfe von PuTTYgen unter Windows finden Sie auf der SSH.com-Website.

Konvertiert einen öffentlichen SSH2-Schlüssel in das PEM-Format

AWS Transfer Family akzeptiert nur öffentliche Schlüssel im PEM-Format. Wenn Sie einen öffentlichen SSH2-Schlüssel haben, müssen Sie ihn konvertieren. Ein öffentlicher SSH2-Schlüssel hat das folgende Format:

```
---- BEGIN SSH2 PUBLIC KEY ----
Comment: "rsa-key-20160402"
AAAAB3NzaC1yc2EAAAABJQAAAgEAiL0jjDdFqK/kYThqKt7THrjABTPWvXmB3URI
:
:
---- END SSH2 PUBLIC KEY ----
```

Ein öffentlicher PEM-Schlüssel hat das folgende Format:

```
ssh-rsa AAAAB3NzaC1yc2EAAAABJQAAA...
```

Führen Sie den folgenden Befehl aus, um einen öffentlichen Schlüssel im SSH2-Format in einen öffentlichen Schlüssel im PEM-Format zu konvertieren. *Ersetzen Sie ssh2-key durch den Namen Ihres SSH2-Schlüssels und PEM-Key durch den Namen Ihres PEM-Schlüssels.*

```
ssh-keygen -i -f ssh2-key.pub > PEM-key.pub
```

## Drehen Sie die SSH-Schlüssel

Aus Sicherheitsgründen empfehlen wir die bewährte Methode, Ihre SSH-Schlüssel zu rotieren. In der Regel wird diese Rotation als Teil einer Sicherheitsrichtlinie festgelegt und automatisiert implementiert. Je nach Sicherheitsstufe kann ein SSH-Schlüsselpaar für eine hochsensible Kommunikation nur einmal verwendet werden. Dadurch werden Risiken vermieden, die aus dem Speichern von Schlüsseln erwachsen. Es ist jedoch viel üblicher, SSH-Anmeldeinformationen für einen bestimmten Zeitraum zu speichern und ein Intervall festzulegen, das die Benutzer nicht übermäßig belastet. Dieser Zeitraum hat typischerweise eine Länge von drei Monaten.

Es gibt zwei Methoden zum Durchführen der SSH-Schlüsselrotation:

- Auf der Konsole können Sie einen neuen öffentlichen SSH-Schlüssel hochladen und einen vorhandenen öffentlichen SSH-Schlüssel löschen.
- Mithilfe der API können Sie bestehende Benutzer aktualisieren, indem Sie die DeleteSshPublicKeyAPI verwenden, um den öffentlichen Secure Shell (SSH) -Schlüssel eines Benutzers zu löschen, und die ImportSshPublicKeyAPI, um dem Konto des Benutzers einen neuen öffentlichen Secure Shell (SSH) -Schlüssel hinzuzufügen.

### Console

Um eine Schlüsselrotation in der Konsole durchzuführen

1. Öffnen Sie die AWS Transfer Family Konsole unter https://console.aws.amazon.com/transfer/.
2. Navigieren Sie zur Seite Server.
3. Wählen Sie den Bezeichner in der Spalte Server-ID aus, um die Seite mit den Serverdetails aufzurufen.

4. Wählen Sie unter Benutzer das Kontrollkästchen des Benutzers aus, dessen öffentlichen SSH-Schlüssel Sie rotieren möchten. Wählen Sie dann Aktionen und anschließend Schlüssel hinzufügen aus, um die Seite Schlüssel hinzufügen aufzurufen.

   or

   Wählen Sie den Benutzernamen aus, um die Seite mit den Benutzerdetails aufzurufen, und wählen Sie dann Öffentlichen SSH-Schlüssel hinzufügen aus, um die Seite Schlüssel hinzufügen aufzurufen.

5. Geben Sie den neuen öffentlichen SSH-Schlüssel ein und wählen Sie Schlüssel hinzufügen.

   > **Important**
   >
   > Das Format des öffentlichen SSH-Schlüssels hängt von der Art des von Ihnen generierten Schlüssels ab.
   >
   > - Für RSA-Schlüssel lautet das Format. `ssh-rsa` *`string`*
   > - Für ED25519-Schlüssel lautet das Format. `ssh-ed25519` *`string`*
   > - Bei ECSA-Schlüsseln beginnt der Schlüssel je nach Größe des von `ecdsa-sha2-nistp384` Ihnen generierten Schlüssels mit `ecdsa-sha2-nistp256ecdsa-sha2-nistp521`, oder. Auf die Anfangszeichenfolge folgt dann*`string`*, ähnlich wie bei den anderen Schlüsseltypen.

   Sie kehren zur Seite mit den Benutzerdetails zurück, und der neue öffentliche SSH-Schlüssel, den Sie gerade eingegeben haben, wird im Abschnitt Öffentliche SSH-Schlüssel angezeigt.

6. Aktivieren Sie das Kontrollkästchen des alten YOU-Schlüssels, den Sie löschen möchten, und wählen Sie dann Löschen.

7. Bestätigen Sie den Löschvorgang, indem Sie das Wort eingeben`delete`, und wählen Sie dann Löschen.

API

Um eine Schlüsselrotation mithilfe der API durchzuführen

1. Öffnen Sie auf macOS-, Linux- oder Unix-Betriebssystemen ein Befehlsterminal.

2. Rufen Sie den SSH-Schlüssel ab, den Sie löschen möchten, indem Sie den folgenden Befehl eingeben. Um diesen Befehl zu verwenden, *serverID* ersetzen Sie ihn durch die Server-ID für Ihren Transfer Family Family-Server und *username* durch Ihren Benutzernamen.

```
aws transfer describe-user --server-id='serverID' --user-name='username'
```

Der Befehl gibt Details über den Benutzer zurück. Kopieren Sie den Inhalt des `"SshPublicKeyId":` Felds. Sie müssen diesen Wert später in diesem Verfahren eingeben.

```
"SshPublicKeys": [ { "SshPublicKeyBody": "public-key", "SshPublicKeyId":
 "keyID",
   "DateImported": 1621969331.072 } ],
```

3. Importieren Sie als Nächstes einen neuen SSH-Schlüssel für Ihren Benutzer. Geben Sie an der -Eingabeaufforderung folgenden Befehl ein. Um diesen Befehl zu verwenden, *serverID* ersetzen Sie ihn durch die Server-ID für Ihren Transfer Family Family-Server, *username* ersetzen Sie ihn durch Ihren Benutzernamen und *public-key* ersetzen Sie ihn durch den Fingerabdruck Ihres neuen öffentlichen Schlüssels.

```
aws transfer import-ssh-public-key --server-id='serverID' --user-name='username'
   --ssh-public-key-body='public-key'
```

Wenn der Befehl erfolgreich ist, wird keine Ausgabe zurückgegeben.

4. Löschen Sie abschließend den alten Schlüssel, indem Sie den folgenden Befehl ausführen. Um diesen Befehl zu verwenden, *serverID* ersetzen Sie ihn durch die Server-ID für Ihren Transfer Family Family-Server, *username* ersetzen Sie ihn durch Ihren Benutzernamen und *keyID-from-step-2* ersetzen Sie ihn durch den Schlüssel-ID-Wert, den Sie in Schritt 2 dieses Verfahrens kopiert haben.

```
aws transfer delete-ssh-public-key --server-id='serverID' --user-name='username'
   --ssh-public-key-id='keyID-from-step-2'
```

5. (Optional) Um zu bestätigen, dass der alte Schlüssel nicht mehr existiert, wiederholen Sie Schritt 2.

## Generieren und verwalten Sie PGP-Schlüssel

Sie können die PGP-Entschlüsselung (Pretty Good Privacy) für die Dateien verwenden, die Transfer Family mit Workflows verarbeitet. Um die Entschlüsselung in einem Workflow-Schritt zu verwenden, geben Sie einen PGP-Schlüssel an.

Der AWS Speicher-Blog enthält einen Beitrag, in dem beschrieben wird, wie Dateien mithilfe von Transfer Family Managed Workflows, Verschlüsseln und Entschlüsseln von Dateien mit PGP und einfach entschlüsselt werden können, ohne Code zu schreiben, beschrieben werden. AWS Transfer Family

Generieren Sie PGP-Schlüssel

Der Operator, den Sie zum Generieren Ihrer PGP-Schlüssel verwenden, hängt von Ihrem Betriebssystem und der Version der Software zur Schlüsselgenerierung ab, die Sie verwenden.

Wenn Sie Linux oder Unix verwenden, verwenden Sie zur Installation Ihr Paketinstallationsprogramm. `gpg` Abhängig von Ihrer Linux-Distribution sollte einer der folgenden Befehle für Sie funktionieren.

```
sudo yum install gnupg
```

```
sudo apt-get install gnupg
```

Für Windows oder macOS können Sie alles, was Sie benötigen, von https://gnupg.org/download/ herunterladen.

Nachdem Sie Ihre PGP-Schlüsselgenerator-Software installiert haben, führen Sie den `gpg --gen-key` Befehl `gpg --full-gen-key` or aus, um ein key pair zu generieren.

> **Note**
>
> Wenn Sie `GnuPG` Version 2.3.0 oder neuer verwenden, müssen Sie Folgendes ausführen. `gpg --full-gen-key` Wenn Sie nach dem Typ des zu erstellenden Schlüssels gefragt werden, wählen Sie RSA oder ECC. Wenn Sie jedoch ECC wählen, stellen Sie sicher, dass Sie BrainPool für die elliptische Kurve entweder NIST oder wählen. Wählen Sie nicht. Curve 25519

Algorithmen, die für PGP-Schlüsselpaare unterstützt werden

Wir unterstützen die folgenden Algorithmen für PGP-Schlüsselpaare:

- RSA
- Elgamal
- ECC:
  - NIST
  - BrainPool

> **Note**
>
> Wir unterstützen keine Curve25519-Schlüssel.

### **gpg**Nützliche Unterbefehle

Im Folgenden finden Sie einige nützliche Unterbefehle für: `gpg`

- `gpg --help`— Dieser Befehl listet die verfügbaren Optionen auf und kann einige Beispiele enthalten.
- `gpg --list-keys`— Dieser Befehl listet die Details für alle Schlüsselpaare auf, die Sie erstellt haben.
- `gpg --fingerprint`— Dieser Befehl listet die Details für alle Ihre Schlüsselpaare auf, einschließlich des Fingerabdrucks der einzelnen Schlüssel.
- `gpg --export -a` *`user-name`*— Dieser Befehl exportiert den öffentlichen Schlüsselteil des Schlüssels für den*`user-name`*, der bei der Generierung des Schlüssels verwendet wurde.

### PGP-Schlüssel verwalten

Um Ihre PGP-Schlüssel zu verwalten, verwenden Sie. AWS Secrets Manager

> **Note**
>
> Ihr geheimer Name beinhaltet Ihre Transfer Family Family-Server-ID. Das bedeutet, dass Sie bereits einen Server identifiziert oder erstellt haben sollten, bevor Sie Ihre PGP-Schlüsselinformationen darin AWS Secrets Manager speichern können.

Wenn Sie einen Schlüssel und eine Passphrase für alle Ihre Benutzer verwenden möchten, können Sie die PGP-Schlüsselblockinformationen unter dem geheimen Namen speichern`aws/transfer/`*`server-id`*`/@pgp-default`, wo sich die ID für Ihren Transfer Family Family-Server *`server-id`* befindet. Transfer Family verwendet diesen Standardschlüssel, wenn es keinen Schlüssel gibt, der dem Benutzer *`user-name`* entspricht, der den Workflow ausführt.

Sie können einen Schlüssel für einen bestimmten Benutzer erstellen. In diesem Fall lautet das Format für den geheimen Name`aws/transfer/`*`server-id`*`/`*`user-name`*, wobei dem Benutzer *`user-name`* entspricht, der den Workflow für einen Transfer Family Family-Server ausführt.

> **Note**
>
> Sie können maximal 3 private PGP-Schlüssel pro Transfer Family Family-Server pro Benutzer speichern.

Um PGP-Schlüssel für die Verwendung bei der Entschlüsselung zu konfigurieren

1. Führen Sie je nach verwendeter GPG-Version einen der folgenden Befehle aus, um ein PGP-Schlüsselpaar zu generieren, das keinen Curve 25519-Verschlüsselungsalgorithmus verwendet.

   - Wenn Sie **GnuPG** Version 2.3.0 oder neuer verwenden, führen Sie den folgenden Befehl aus:

     ```
     gpg --full-gen-key
     ```

     Sie können wählen**RSA**, oder, falls Sie möchten**ECC**, können Sie entweder **NIST** oder **BrainPool** für die elliptische Kurve wählen. Wenn Sie `gpg --gen-key` stattdessen ausführen, erstellen Sie ein key pair, das den ECC Curve 25519-Verschlüsselungsalgorithmus verwendet, den wir derzeit nicht für PGP-Schlüssel unterstützen.

   - Für Versionen **GnuPG** vor 2.3.0 können Sie den folgenden Befehl verwenden, da RSA der Standardverschlüsselungstyp ist.

     ```
     gpg --gen-key
     ```

> **Important**
>
> Während der Schlüsselgenerierung müssen Sie eine Passphrase und eine E-Mail-Adresse angeben. Achten Sie darauf, diese Werte zu notieren. Sie müssen die Passphrase angeben, wenn Sie die Schlüsseldetails AWS Secrets Manager später in diesem Verfahren eingeben. Und Sie müssen dieselbe E-Mail-Adresse angeben, um den privaten Schlüssel im nächsten Schritt zu exportieren.

2. Führen Sie den folgenden Befehl aus, um den privaten Schlüssel zu exportieren. Um diesen Befehl zu verwenden, *`private.pgp`* ersetzen Sie ihn durch den Namen der Datei, in der der private Schlüsselblock gespeichert werden soll, und *`marymajor@example.com`* durch die E-Mail-Adresse, die Sie bei der Generierung des key pair verwendet haben.

```
gpg --output private.pgp --armor --export-secret-key marymajor@example.com
```

3. Dient AWS Secrets Manager zum Speichern Ihres PGP-Schlüssels.

   a. Melden Sie sich bei der an AWS Management Console und öffnen Sie die AWS Secrets Manager Konsole unter https://console.aws.amazon.com/secretsmanager/.

   b. Wählen Sie im linken Navigationsbereich Secrets aus.

   c. Wählen Sie auf der Seite Secrets die Option Neues Geheimnis speichern aus.

   d. Wählen Sie auf der Seite Geheimtyp auswählen für Geheimtyp die Option Anderer Geheimtyp aus.

   e. Wählen Sie im Abschnitt Schlüssel/Wert-Paare die Registerkarte Schlüssel/Wert aus.

      - Schlüssel — Geben Sie ein. **`PGPPrivateKey`**

        > **Note**
        >
        > Sie müssen die **`PGPPrivateKey`** Zeichenfolge exakt eingeben: Fügen Sie vor oder zwischen den Zeichen keine Leerzeichen hinzu.

      - Wert — Fügen Sie den Text Ihres privaten Schlüssels in das Wertfeld ein. Sie finden den Text Ihres privaten Schlüssels in der Datei (z. B.`private.pgp`), die Sie beim Exportieren Ihres Schlüssels zu Beginn dieses Verfahrens angegeben haben. Der Schlüssel beginnt

mit `-----BEGIN PGP PRIVATE KEY BLOCK-----` und endet mit`-----END PGP PRIVATE KEY BLOCK-----`.

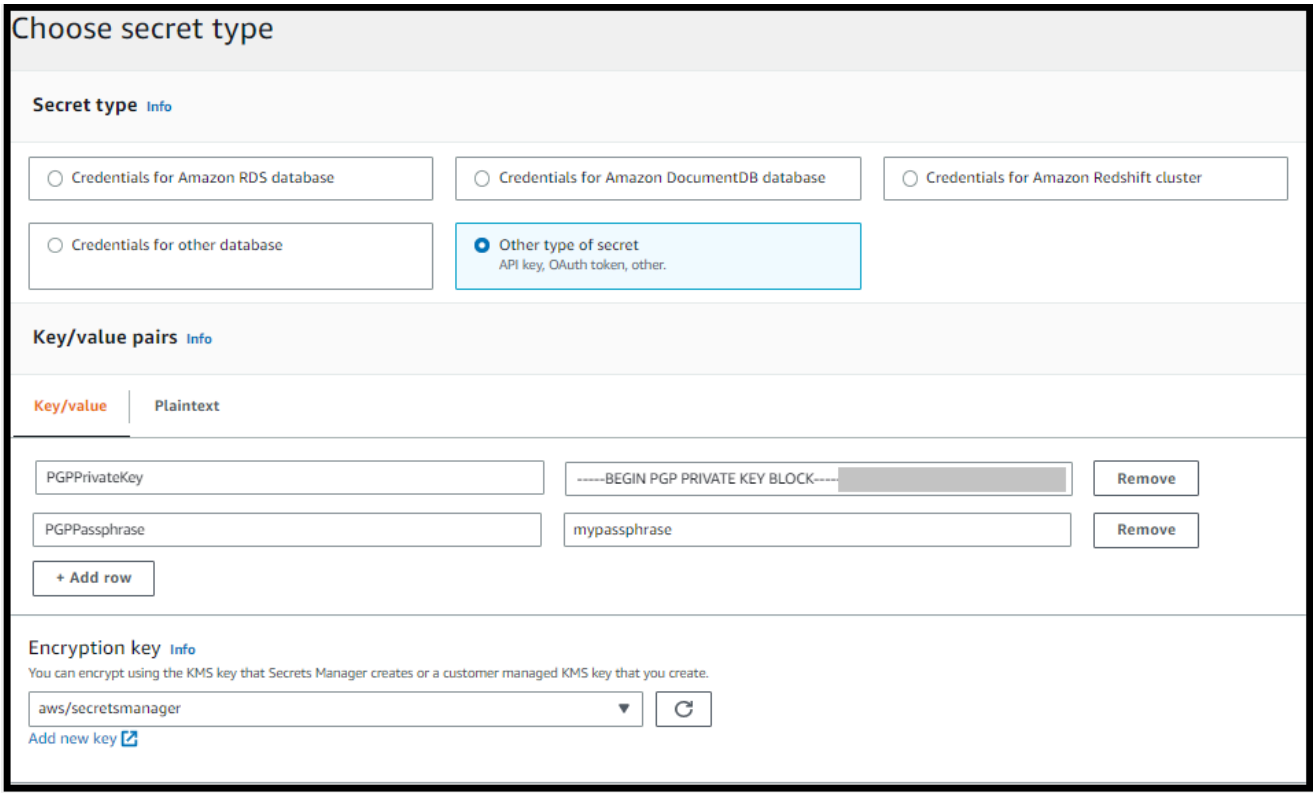
> **Note**
>
> Stellen Sie sicher, dass der Textblock nur den privaten Schlüssel und nicht auch den öffentlichen Schlüssel enthält.

f. Wählen Sie Zeile hinzufügen und wählen Sie im Abschnitt Schlüssel/Wert-Paare die Registerkarte Schlüssel/Wert-Paare aus.

- Schlüssel — Geben Sie ein. **`PGPPassphrase`**

> **Note**
>
> Sie müssen die **`PGPPassphrase`** Zeichenfolge exakt eingeben: Fügen Sie vor oder zwischen den Zeichen keine Leerzeichen hinzu.

- Wert — Geben Sie die Passphrase ein, die Sie bei der Generierung Ihres PGP-Schlüsselpaars verwendet haben.




> **Note**
>
> Sie können bis zu 3 Sätze von Schlüsseln und Passphrasen hinzufügen. Um einen zweiten Satz hinzuzufügen, fügen Sie zwei neue Zeilen hinzu, geben Sie und **`PGPPassphrase2`** für die Schlüssel ein **`PGPPrivateKey2`** und fügen Sie einen weiteren privaten Schlüssel und eine Passphrase ein. Um einen dritten Satz hinzuzufügen, müssen die Schlüsselwerte und sein. **`PGPPrivateKey3`** **`PGPPassphrase3`**

g. Wählen Sie Weiter aus.

h. Geben Sie auf der Seite Geheimen Schlüssel konfigurieren einen Namen und eine Beschreibung für Ihr Geheimnis ein.

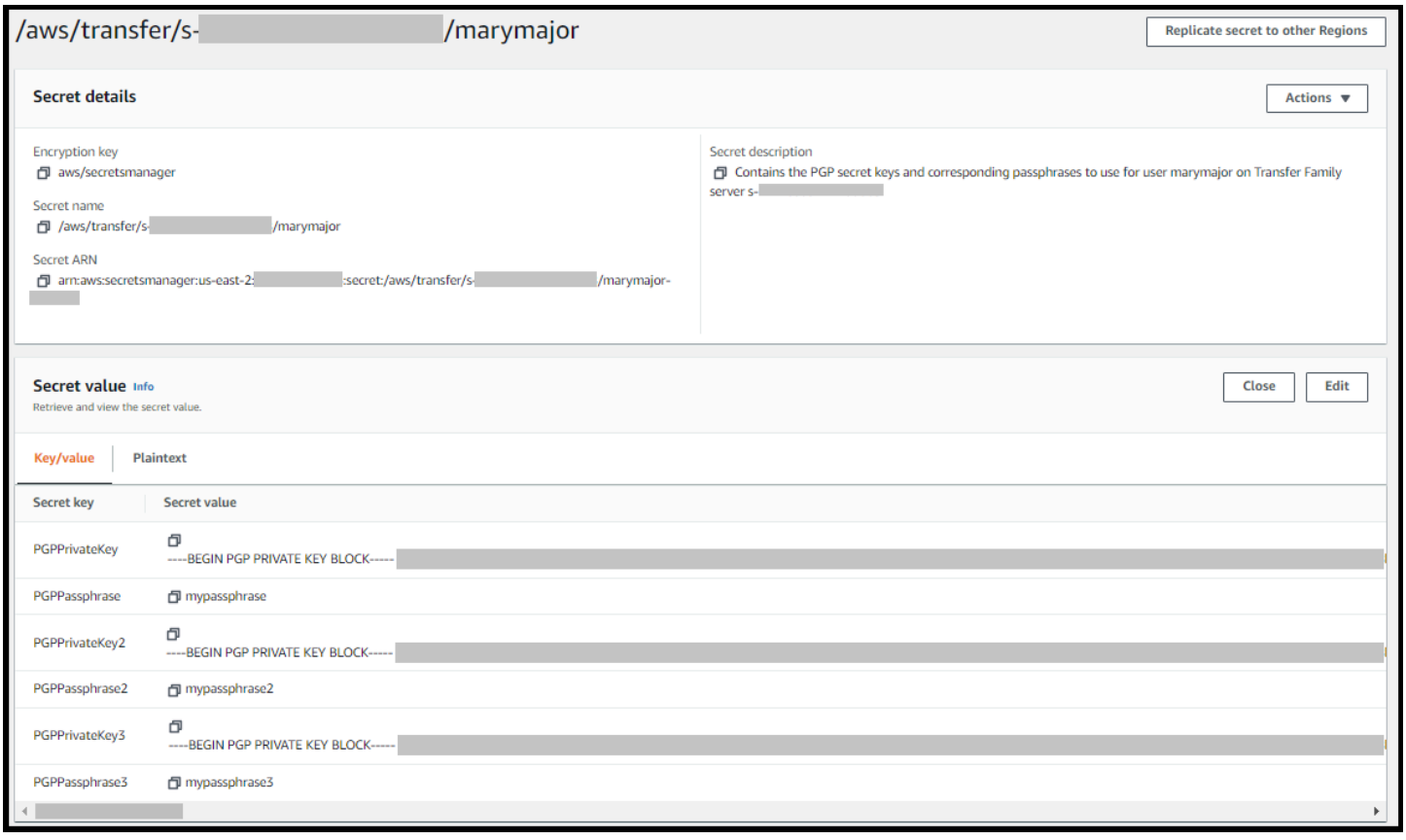
- Wenn Sie einen Standardschlüssel erstellen, d. h. einen Schlüssel, der von jedem Transfer Family Family-Benutzer verwendet werden kann, geben Sie ein**`aws/transfer/`***`server-id`***`/@pgp-default`**. *`server-id`*Ersetzen Sie ihn durch die ID des Servers, der den Workflow enthält, der einen Entschlüsselungsschritt enthält.

- Wenn Sie einen Schlüssel erstellen, der von einem bestimmten Transfer Family Family-Benutzer verwendet werden soll, geben Sie ein**aws/transfer/*server-id*/*user-name***. *server-id*Ersetzen Sie ihn durch die ID des Servers, der den Workflow enthält, der einen Entschlüsselungsschritt enthält, und *user-name* ersetzen Sie ihn durch den Namen des Benutzers, der den Workflow ausführt. Das *user-name* wird in dem Identitätsanbieter gespeichert, den der Transfer Family Family-Server verwendet.

i. Wählen Sie Weiter und akzeptieren Sie die Standardeinstellungen auf der Seite „Rotation konfigurieren“. Wählen Sie anschließend Weiter.

j. Wählen Sie auf der Seite „Überprüfen“ die Option Speichern aus, um das Geheimnis zu erstellen und zu speichern.

Der folgende Screenshot zeigt die Details für den Benutzer **marymajor** für einen bestimmten Transfer Family Family-Server. Dieses Beispiel zeigt drei Schlüssel und die entsprechenden Passphrasen.

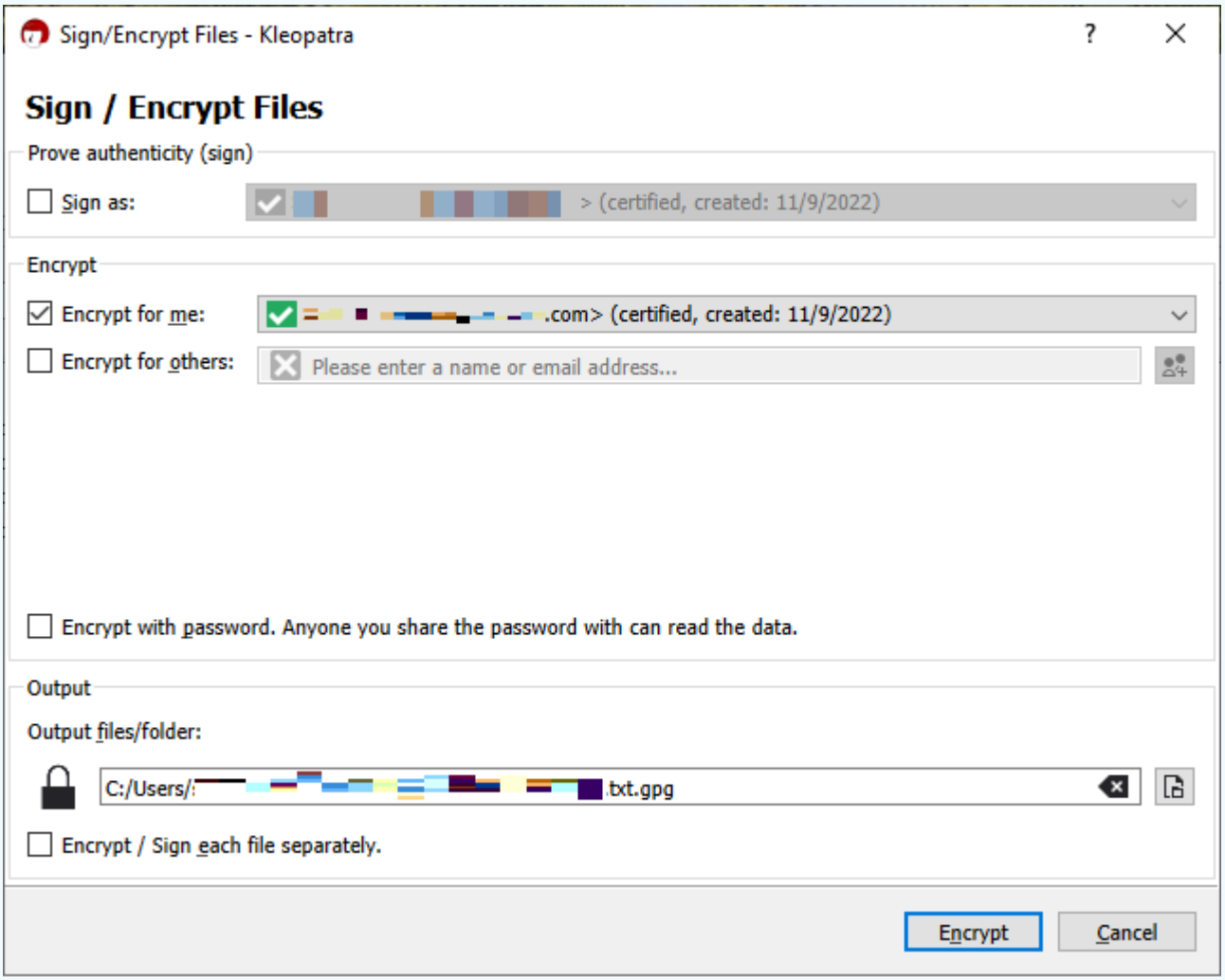


## Unterstützte PGP-Clients

Die folgenden Clients wurden mit Transfer Family getestet und können zum Generieren von PGP-Schlüsseln und zum Verschlüsseln von Dateien verwendet werden, die Sie mit einem Workflow entschlüsseln möchten.

- Gpg4win + Kleopatra.

> **Note**
>
> Wenn Sie Dateien signieren/verschlüsseln auswählen, stellen Sie sicher, dass Sie die Auswahl für Signieren als deaktivieren: Derzeit unterstützen wir das Signieren von verschlüsselten Dateien nicht.




Wenn Sie die verschlüsselte Datei signieren und versuchen, sie mit einem Entschlüsselungsworkflow auf einen Transfer Family Family-Server hochzuladen, wird die folgende Fehlermeldung angezeigt:

```
Encrypted file with signed message unsupported
```

- Hauptversionen von GnuPG: 2.4, 2.3, 2.2, 2.0 und 1.4.

Beachten Sie, dass andere PGP-Clients möglicherweise auch funktionieren, aber nur die hier genannten Clients wurden mit Transfer Family getestet.

# Identitäts- und Zugriffsmanagement für AWS Transfer Family

AWS Identity and Access Management (IAM) hilft einem Administrator AWS-Service , den Zugriff auf Ressourcen sicher zu AWS kontrollieren. IAM-Administratoren kontrollieren, wer authentifiziert (angemeldet) und autorisiert werden kann (über Berechtigungen verfügt), um Ressourcen zu verwenden. AWS Transfer Family IAM ist ein Programm AWS-Service , das Sie ohne zusätzliche Kosten nutzen können.

Themen


- Zielgruppe
- Authentifizierung mit Identitäten
- Verwalten des Zugriffs mit Richtlinien
- Wie AWS Transfer Family funktioniert mit IAM
- AWS Transfer Family Beispiele für identitätsbasierte Richtlinien
- AWS Transfer Family Beispiele für Tag-basierte Richtlinien
- Problembehandlung bei AWS Transfer Family Identität und Zugriff


## Zielgruppe

Die Art und Weise, wie Sie AWS Identity and Access Management (IAM) verwenden, hängt von der Arbeit ab, in der Sie tätig sind. AWS Transfer Family

Dienstbenutzer — Wenn Sie den AWS Transfer Family Dienst für Ihre Arbeit verwenden, stellt Ihnen Ihr Administrator die erforderlichen Anmeldeinformationen und Berechtigungen zur Verfügung. Wenn Sie für Ihre Arbeit mehr AWS Transfer Family Funktionen verwenden, benötigen Sie möglicherweise zusätzliche Berechtigungen. Wenn Sie die Fuktionsweise der Zugriffskontrolle nachvollziehen,

wissen Sie bereits, welche Berechtigungen Sie von Ihrem Administrator anzufordern müssen. Unter Problembehandlung bei AWS Transfer Family Identität und Zugriff finden Sie nützliche Informationen für den Fall, dass Sie keinen Zugriff auf eine Feature in AWS Transfer Family haben.

Serviceadministrator — Wenn Sie in Ihrem Unternehmen für die AWS Transfer Family Ressourcen verantwortlich sind, haben Sie wahrscheinlich vollen Zugriff auf AWS Transfer Family. Es ist Ihre Aufgabe, zu bestimmen, auf welche AWS Transfer Family Funktionen und Ressourcen 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 nutzen kann AWS Transfer Family, finden Sie unterWie AWS Transfer Family funktioniert mit IAM.

IAM-Administrator: Wenn Sie als IAM-Administrator fungieren, sollten Sie Einzelheiten dazu kennen, wie Sie Richtlinien zur Verwaltung des Zugriffs auf AWS Transfer Family verfassen können. Beispiele für AWS Transfer Family identitätsbasierte Richtlinien, die Sie in IAM verwenden können, finden Sie unter. AWS Transfer Family Beispiele für identitätsbasierte Richtlinien

## Authentifizierung mit Identitäten

Authentifizierung ist die Art und Weise, wie Sie sich AWS mit Ihren Identitätsdaten anmelden. Sie müssen als IAM-Benutzer authentifiziert (angemeldet AWS) sein oder eine IAM-Rolle annehmen. Root-Benutzer des AWS-Kontos

Sie können sich AWS als föderierte Identität anmelden, indem Sie Anmeldeinformationen verwenden, die über eine Identitätsquelle bereitgestellt wurden. AWS IAM Identity Center (IAM Identity Center) -Benutzer, die Single Sign-On-Authentifizierung Ihres Unternehmens und Ihre Google- oder Facebook-Anmeldeinformationen sind Beispiele für föderierte Identitäten. Wenn Sie sich als Verbundidentität anmelden, hat der Administrator vorher mithilfe von IAM-Rollen einen Identitätsverbund eingerichtet. Wenn Sie über den Verbund darauf zugreifen AWS , übernehmen Sie indirekt eine Rolle.

Je nachdem, welcher Benutzertyp Sie sind, können Sie sich beim AWS Management Console oder beim AWS Zugangsportal anmelden. Weitere Informationen zur Anmeldung finden Sie AWS unter So melden Sie sich bei Ihrem an AWS-Konto im AWS-Anmeldung Benutzerhandbuch.

Wenn Sie AWS programmgesteuert darauf zugreifen, AWS stellt es ein Software Development Kit (SDK) und eine Befehlszeilenschnittstelle (CLI) bereit, um Ihre Anfragen mithilfe Ihrer Anmeldeinformationen kryptografisch zu signieren. Wenn Sie keine AWS Tools verwenden, müssen

Sie Anfragen selbst signieren. Weitere Informationen zur Verwendung der empfohlenen Methode, um Anfragen selbst zu signieren, finden Sie im IAM-Benutzerhandbuch unter AWS API-Anfragen signieren.

Unabhängig von der verwendeten Authentifizierungsmethode müssen Sie möglicherweise zusätzliche Sicherheitsinformationen angeben. AWS Empfiehlt beispielsweise, die Multi-Faktor-Authentifizierung (MFA) zu verwenden, um die Sicherheit Ihres Kontos zu erhöhen. 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.

### Root-Benutzer des AWS-Kontos

Wenn Sie eine erstellen AWS-Konto, beginnen Sie mit einer Anmeldeidentität, die vollständigen Zugriff auf alle Ressourcen im AWS-Services Konto hat. Diese Identität wird als AWS-Konto Root-Benutzer bezeichnet. Sie können darauf zugreifen, indem Sie sich mit der E-Mail-Adresse und dem Passwort anmelden, mit denen Sie das Konto erstellt haben. 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 sollten menschliche Benutzer, einschließlich Benutzer, die Administratorzugriff benötigen, für den Zugriff AWS-Services mithilfe temporärer Anmeldeinformationen den Verbund mit einem Identitätsanbieter verwenden.

Eine föderierte Identität ist ein Benutzer aus Ihrem Unternehmensbenutzerverzeichnis, einem Web-Identitätsanbieter AWS Directory Service, dem Identity Center-Verzeichnis oder einem beliebigen Benutzer, der mithilfe AWS-Services von Anmeldeinformationen zugreift, die über eine Identitätsquelle bereitgestellt wurden. Wenn föderierte Identitäten darauf zugreifen AWS-Konten, 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 in IAM Identity Center erstellen, oder Sie können eine Verbindung zu einer Gruppe von Benutzern und Gruppen in Ihrer eigenen Identitätsquelle herstellen und diese synchronisieren, um sie in all Ihren AWS-Konten 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 innerhalb Ihres Unternehmens AWS-Konto , die über spezifische Berechtigungen für eine einzelne Person oder Anwendung verfügt. 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 innerhalb Ihres Unternehmens AWS-Konto , die über bestimmte Berechtigungen verfügt. Sie ist einem IAM-Benutzer vergleichbar, ist aber nicht mit einer bestimmten Person verknüpft. Sie können vorübergehend eine IAM-Rolle in der übernehmen, AWS Management Console indem Sie die Rollen wechseln. Sie können eine Rolle übernehmen, 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. Bei einigen können Sie AWS-Services jedoch eine Richtlinie direkt an eine Ressource anhängen (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 Funktionen 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 verwenden, um Aktionen auszuführen AWS, gelten Sie als Principal. 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 aufruft AWS-Service, in Kombination mit der Anfrage, Anfragen an AWS-Service nachgelagerte Dienste zu stellen. FAS-Anfragen werden nur gestellt, wenn ein Dienst eine Anfrage erhält, für deren Abschluss Interaktionen mit anderen AWS-Services oder Ressourcen erforderlich sind. 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.

- Dienstbezogene Rolle — Eine dienstbezogene Rolle ist eine Art von Servicerolle, die mit einer verknüpft ist. AWS-Service Der Service kann die Rolle übernehmen, um eine Aktion in Ihrem Namen auszuführen. Servicebezogene Rollen erscheinen in Ihrem Dienst AWS-Konto und gehören dem Dienst. Ein IAM-Administrator kann die Berechtigungen für Service-verknüpfte Rollen anzeigen, aber nicht bearbeiten.

- Anwendungen, die auf Amazon EC2 ausgeführt werden — Sie können eine IAM-Rolle verwenden, um temporäre Anmeldeinformationen für Anwendungen zu verwalten, die auf einer EC2-Instance ausgeführt werden und API-Anfragen stellen AWS CLI . AWS Das ist eher zu empfehlen, als Zugriffsschlüssel innerhalb der EC2-Instance zu speichern. Um einer EC2-Instance eine AWS Rolle zuzuweisen und sie allen ihren Anwendungen zur Verfügung zu stellen, erstellen Sie ein Instance-Profil, das an die Instance angehängt ist. 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

Sie kontrollieren den Zugriff, AWS indem Sie Richtlinien erstellen und diese an AWS Identitäten oder Ressourcen anhängen. Eine Richtlinie ist ein Objekt, AWS das, wenn es einer Identität oder Ressource zugeordnet ist, deren Berechtigungen definiert. AWS wertet diese Richtlinien aus, wenn ein Prinzipal (Benutzer, Root-Benutzer oder Rollensitzung) eine Anfrage stellt. Berechtigungen in den Richtlinien bestimmen, ob die Anforderung zugelassen oder abgelehnt wird. Die meisten Richtlinien werden 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 angeben, wer auf was Zugriff hat. 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 Rolleninformationen von der AWS Management Console AWS CLI, der oder der 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 System zuordnen können AWS-Konto. Zu den verwalteten Richtlinien gehören AWS verwaltete Richtlinien und vom 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. Zu den Prinzipalen können Konten, Benutzer, Rollen, Verbundbenutzer oder gehören. AWS-Services

Ressourcenbasierte Richtlinien sind Richtlinien innerhalb dieses Diensts. Sie können AWS verwaltete 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 und Amazon VPC sind Beispiele für Services, die ACLs unterstützen. AWS WAF 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 verbreitete 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. Eine explizite Zugriffsverweigerung in einer dieser Richtlinien setzt eine Zugriffserlaubnis außer Kraft. Weitere Informationen über Berechtigungsgrenzen finden Sie unter Berechtigungsgrenzen für IAM-Entitäten im IAM-Benutzerhandbuch.
- Service Control Policies (SCPs) — SCPs sind JSON-Richtlinien, die die maximalen Berechtigungen für eine Organisation oder Organisationseinheit (OU) in festlegen. AWS Organizations AWS Organizations ist ein Dienst zur Gruppierung und zentralen Verwaltung mehrerer Objekte AWS-Konten , die Ihrem Unternehmen gehören. Wenn Sie innerhalb einer Organisation alle Features aktivieren, können Sie Service-Kontrollrichtlinien (SCPs) auf alle oder einzelne Ihrer Konten anwenden. Das SCP schränkt die Berechtigungen für Entitäten in Mitgliedskonten ein, einschließlich der einzelnen Entitäten. Root-Benutzer des AWS-Kontos Weitere Informationen zu Organizations 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 darüber, wie AWS bestimmt wird, ob eine Anfrage zulässig ist, wenn mehrere Richtlinientypen betroffen sind, finden Sie im IAM-Benutzerhandbuch unter Bewertungslogik für Richtlinien.

## Wie AWS Transfer Family funktioniert mit IAM

Bevor Sie AWS Identity and Access Management (IAM) zur Verwaltung des Zugriffs auf verwenden AWS Transfer Family, sollten Sie wissen, mit welchen IAM-Funktionen Sie arbeiten können. AWS Transfer FamilyEinen allgemeinen Überblick darüber, wie AWS Transfer Family und andere AWS Dienste mit IAM funktionieren, finden Sie im AWS IAM-Benutzerhandbuch unter Dienste, die mit IAM funktionieren.

Themen

- Identitätsbasierte AWS Transfer Family -Richtlinien
- Ressourcenbasierte AWS Transfer Family -Richtlinien
- Autorisierung auf der Basis von AWS Transfer Family -Tags
- AWS Transfer Family IAM-Rollen

### Identitätsbasierte AWS Transfer Family -Richtlinien

Mit identitätsbasierten IAM-Richtlinien können Sie angeben, welche Aktionen und Ressourcen erteilt oder abgelehnt werden. Darüber hinaus können Sie die Bedingungen festlegen, unter denen Aktionen zugelassen oder abgelehnt werden. AWS Transfer Family unterstützt bestimmte Aktionen, Ressourcen und Bedingungsschlüssel. Weitere Informationen zu allen Elementen, die Sie in einer JSON-Richtlinie verwenden, finden Sie im Benutzerhandbuch unter Referenz zu den IAM-JSON-Richtlinienelementen.AWS Identity and Access Management

Aktionen

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer auf was Zugriff hat. 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 der zugehörige AWS API-Vorgang. Es gibt einige Ausnahmen, z. B. Aktionen, die nur mit Genehmigung durchgeführt werden können und für die es keinen passenden API-Vorgang 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.

Bei Richtlinienaktionen wird vor der Aktion das folgende Präfix AWS Transfer Family verwendet:`transfer:`. Um beispielsweise jemandem die Erlaubnis zu erteilen, einen Server mit dem Transfer Family `CreateServer` API-Vorgang zu erstellen, nehmen Sie die `transfer:CreateServer` Aktion in seine Richtlinie auf. Richtlinienanweisungen müssen ein `Action`- oder `NotAction`-Element enthalten. AWS Transfer Family definiert seinen eigenen Satz an Aktionen, die Aufgaben beschreiben, die Sie mit diesem Service durchführen können.

Um mehrere -Aktionen in einer einzigen Anweisung anzugeben, trennen Sie sie folgendermaßen durch Kommas.

```
"Action": [
      "transfer:action1",
      "transfer:action2"
```

Sie können auch Platzhalter (*) verwenden, um mehrere Aktionen anzugeben. Beispielsweise können Sie alle Aktionen festlegen, die mit dem Wort `Describe` beginnen, einschließlich der folgenden Aktion:

```
"Action": "transfer:Describe*"
```

Eine Liste der AWS Transfer Family Aktionen finden Sie unter Aktionen definiert von AWS Transfer Family in der Service Authorization Reference.

Ressourcen

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer Zugriff auf was hat. 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": "*"
```

Die Transfer Family Family-Serverressource hat den folgenden ARN.

```
arn:aws:transfer:${Region}:${Account}:server/${ServerId}
```

Um beispielsweise den `s-01234567890abcdef` Transfer Family Family-Server in Ihrer Anweisung anzugeben, verwenden Sie den folgenden ARN.

```
"Resource": "arn:aws:transfer:us-east-1:123456789012:server/s-01234567890abcdef"
```

Weitere Informationen zum Format von ARNs finden Sie unter Amazon Resource Names (ARNs) in der Service Authorization Reference oder unter IAM-ARNs im IAM-Benutzerhandbuch.

Um alle Instances anzugeben, die zu einem bestimmten Konto gehören, verwenden Sie den Platzhalter (*).

```
"Resource": "arn:aws:transfer:us-east-1:123456789012:server/*"
```

Einige AWS Transfer Family Aktionen werden für mehrere Ressourcen ausgeführt, z. B. für Ressourcen, die in IAM-Richtlinien verwendet werden. In diesen Fällen müssen Sie den Platzhalter (*) verwenden.

```
"Resource": "arn:aws:transfer:*:123456789012:server/*"
```

In einigen Fällen müssen Sie mehr als einen Ressourcentyp angeben, z. B. wenn Sie eine Richtlinie erstellen, die den Zugriff auf Server und Benutzer von Transfer Family ermöglicht. Um mehrere Ressourcen in einer einzigen Anweisung anzugeben, trennen Sie die ARNs durch Kommata voneinander.

```
"Resource": [
      "resource1",
      "resource2"
            ]
```

Eine Liste der AWS Transfer Family Ressourcen finden Sie unter Ressourcentypen definiert von AWS Transfer Family in der Service Authorization Reference.

Bedingungsschlüssel

Administratoren können mithilfe von AWS JSON-Richtlinien angeben, wer Zugriff auf was hat. 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, AWS wertet die Bedingung mithilfe 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 dienstspezifische Bedingungsschlüssel. Eine Übersicht aller AWS globalen Bedingungsschlüssel finden Sie unter Kontextschlüssel für AWS globale Bedingungen im IAM-Benutzerhandbuch.

AWS Transfer Family definiert seinen eigenen Satz von Bedingungsschlüsseln und unterstützt auch die Verwendung einiger globaler Bedingungsschlüssel. Eine Liste der AWS Transfer Family Bedingungsschlüssel finden Sie unter Bedingungsschlüssel für AWS Transfer Family in der Service Authorization Reference.

Beispiele

Beispiele für AWS Transfer Family identitätsbasierte Richtlinien finden Sie unter. AWS Transfer Family Beispiele für identitätsbasierte Richtlinien

## Ressourcenbasierte AWS Transfer Family -Richtlinien

Ressourcenbasierte Richtlinien sind JSON-Richtliniendokumente, die angeben, welche Aktionen ein bestimmter Principal unter welchen Bedingungen auf der AWS Transfer Family Ressource ausführen kann. *Amazon S3 unterstützt ressourcenbasierte Berechtigungsrichtlinien für Amazon S3 S3-Buckets.* Ressourcenbasierte Richtlinien ermöglichen die Erteilung von Nutzungsberechtigungen für andere -Konten pro Ressource. *Sie können auch eine ressourcenbasierte Richtlinie verwenden, um einem AWS Service den Zugriff auf Ihre Amazon S3 S3-Buckets zu ermöglichen.*

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 Principal und die Ressource in unterschiedlichen AWS Konten befinden, müssen Sie der Prinzipaleinheit auch die Erlaubnis erteilen, auf die Ressource zuzugreifen. Sie erteilen Berechtigungen, indem Sie der Entität 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 im Benutzerhandbuch unter Unterschiede zwischen IAM-Rollen und ressourcenbasierten Richtlinien.AWS Identity and Access Management

*Der Amazon S3 S3-Service unterstützt nur eine Art von ressourcenbasierter Richtlinie, die als Bucket-Richtlinie bezeichnet wird und an einen Bucket angehängt ist.* Diese Richtlinie definiert, welche Hauptentitäten (Konten, Benutzer, Rollen und Verbundbenutzer) Aktionen für das Objekt ausführen können.

Beispiele

Beispiele für AWS Transfer Family ressourcenbasierte Richtlinien finden Sie unter. AWS Transfer Family Beispiele für Tag-basierte Richtlinien

### Autorisierung auf der Basis von AWS Transfer Family -Tags

Sie können Tags an AWS Transfer Family Ressourcen anhängen oder Tags in einer Anfrage an übergeben. AWS Transfer Family Um den Zugriff auf der Grundlage von Tags zu steuern, geben Sie im Bedingungselement einer Richtlinie Tag-Informationen an, indem Sie die Schlüssel `transfer:ResourceTag/key-name`, `aws:RequestTag/key-name`, oder Bedingung `aws:TagKeys` verwenden. Informationen zur Verwendung von Tags zur Steuerung des Zugriffs auf AWS Transfer Family Ressourcen finden Sie unterAWS Transfer Family Beispiele für Tag-basierte Richtlinien.

### AWS Transfer Family IAM-Rollen

Eine IAM-Rolle ist eine Entität in Ihrem AWS Konto, die über bestimmte Berechtigungen verfügt.

Verwenden temporärer Anmeldeinformationen mit AWS Transfer Family

Sie können temporäre Anmeldeinformationen verwenden, um sich über einen Verbund anzumelden, eine IAM-Rolle anzunehmen oder eine kontenübergreifende Rolle anzunehmen. Sie erhalten temporäre Sicherheitsanmeldedaten, indem Sie AWS STS API-Operationen wie AssumeRoleoder GetFederationToken aufrufen.

AWS Transfer Family unterstützt die Verwendung temporärer Anmeldeinformationen.

## AWS Transfer Family Beispiele für identitätsbasierte Richtlinien

IAM-Benutzer besitzen keine Berechtigungen zum Erstellen oder Ändern von AWS Transfer Family -Ressourcen. Sie können auch keine Aufgaben mit der AWS Management Console AWS CLI, oder AWS API ausführen. Ein IAM-Administrator muss IAM-Richtlinien erstellen, die Benutzern und Rollen die Berechtigung zum Ausführen bestimmter API-Operationen für die angegebenen Ressourcen gewähren, die diese benötigen. Der Administrator muss diese Richtlinien anschließend den IAM-Benutzern oder -Gruppen anfügen, die diese Berechtigungen benötigen.

Informationen zum Erstellen einer identitätsbasierten IAM-Richtlinie mithilfe dieser Beispieldokumente zu JSON-Richtlinien finden Sie im Benutzerhandbuch unter Erstellen von Richtlinien auf der Registerkarte JSON.AWS Identity and Access Management

Themen

- Bewährte Methoden für Richtlinien
- Verwenden der AWS Transfer Family -Konsole
- Gewähren der Berechtigung zur Anzeige der eigenen Berechtigungen für Benutzer

## Bewährte Methoden für Richtlinien

Identitätsbasierte Richtlinien bestimmen, ob jemand Ressourcen in Ihrem Konto erstellen, darauf zugreifen oder sie löschen AWS Transfer Family kann. Dies kann zusätzliche Kosten für Ihr verursachen AWS-Konto. Befolgen Sie beim Erstellen oder Bearbeiten identitätsbasierter Richtlinien die folgenden Anleitungen und Empfehlungen:

- Beginnen Sie mit AWS verwalteten Richtlinien und wechseln Sie zu Berechtigungen mit den geringsten Rechten — Verwenden Sie die AWS verwalteten Richtlinien, die Berechtigungen für viele gängige Anwendungsfälle gewähren, um Ihren Benutzern und Workloads zunächst Berechtigungen zu gewähren. Sie sind in Ihrem verfügbar. AWS-Konto Wir empfehlen Ihnen, die Berechtigungen weiter zu reduzieren, indem Sie vom AWS Kunden verwaltete Richtlinien definieren, die speziell auf Ihre Anwendungsfälle zugeschnitten sind. Weitere Informationen finden Sie unter AWS -verwaltete Richtlinien oder AWS -verwaltete Richtlinien für Auftrags-Funktionen im IAM-Benutzerhandbuch.
- Anwendung von Berechtigungen mit den geringsten Rechten – Wenn Sie mit IAM-Richtlinien Berechtigungen festlegen, gewähren Sie nur die Berechtigungen, die für die Durchführung einer Aufgabe erforderlich sind. Sie tun dies, indem Sie die Aktionen definieren, die für bestimmte Ressourcen unter bestimmten Bedingungen durchgeführt werden können, auch bekannt als die geringsten Berechtigungen. Weitere Informationen zur Verwendung von IAM zum Anwenden von Berechtigungen finden Sie unter Richtlinien und Berechtigungen in IAM im IAM-Benutzerhandbuch.
- Verwenden von Bedingungen in IAM-Richtlinien zur weiteren Einschränkung des Zugriffs – Sie können Ihren Richtlinien eine Bedingung hinzufügen, um den Zugriff auf Aktionen und Ressourcen zu beschränken. Sie können beispielsweise eine Richtlinienbedingung schreiben, um festzulegen, dass alle Anforderungen mithilfe von SSL gesendet werden müssen. Sie können auch Bedingungen verwenden, um Zugriff auf Serviceaktionen zu gewähren, wenn diese für einen bestimmten Zweck verwendet werden AWS-Service, z. AWS CloudFormation B. Weitere Informationen finden Sie unter IAM-JSON-Richtlinienelemente: Bedingung im IAM-Benutzerhandbuch.
- Verwenden von IAM Access Analyzer zur Validierung Ihrer IAM-Richtlinien, um sichere und funktionale Berechtigungen zu gewährleisten – IAM Access Analyzer validiert neue

und vorhandene Richtlinien, damit die Richtlinien der IAM-Richtliniensprache (JSON) und den bewährten IAM-Methoden entsprechen. IAM Access Analyzer stellt mehr als 100 Richtlinienprüfungen und umsetzbare Empfehlungen zur Verfügung, damit Sie sichere und funktionale Richtlinien erstellen können. Weitere Informationen finden Sie unter Richtlinienvalidierung zum IAM Access Analyzer im IAM-Benutzerhandbuch.

- Multi-Faktor-Authentifizierung (MFA) erforderlich — Wenn Sie ein Szenario haben, das IAM-Benutzer oder einen Root-Benutzer in Ihrem System erfordert AWS-Konto, aktivieren Sie MFA für zusätzliche Sicherheit. Um MFA beim Aufrufen von API-Vorgängen anzufordern, fügen Sie Ihren Richtlinien MFA-Bedingungen hinzu. Weitere Informationen finden Sie unter 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 AWS Transfer Family -Konsole

Um auf die AWS Transfer Family Konsole zugreifen zu können, benötigen Sie ein Mindestmaß an Berechtigungen. Diese Berechtigungen müssen es Ihnen ermöglichen, Informationen zu den AWS Transfer Family Ressourcen in Ihrem AWS Konto aufzulisten und einzusehen. Wenn Sie eine identitätsbasierte Richtlinie erstellen, die strenger ist als die mindestens erforderlichen Berechtigungen, funktioniert die Konsole nicht wie vorgesehen für Entitäten (IAM-Benutzer oder -Rollen) mit dieser Richtlinie. Weitere Informationen finden Sie im Benutzerhandbuch unter Hinzufügen von Berechtigungen für einen AWS Identity and Access Management Benutzer.

Sie müssen Benutzern, die nur die API AWS CLI oder die AWS API aufrufen, keine Mindestberechtigungen für die Konsole gewähren. Stattdessen sollten Sie nur Zugriff auf die Aktionen zulassen, die den API-Operation entsprechen, die Sie ausführen möchten.

## Gewähren der Berechtigung zur Anzeige der eigenen Berechtigungen für Benutzer

In diesem Beispiel wird gezeigt, wie Sie eine Richtlinie erstellen, die IAM-Benutzern die Berechtigung zum Anzeigen der eingebundenen Richtlinien und verwalteten Richtlinien gewährt, die ihrer Benutzeridentität angefügt sind. Diese Richtlinie umfasst Berechtigungen zum Ausführen dieser Aktion auf der Konsole oder programmgesteuert mithilfe der API AWS CLI oder AWS .

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
```

```
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

## AWS Transfer Family Beispiele für Tag-basierte Richtlinien

Im Folgenden finden Sie Beispiele dafür, wie Sie den Zugriff auf AWS Transfer Family Ressourcen anhand von Tags steuern können.

### Verwenden von Tags zur Steuerung des Zugriffs auf AWS Transfer Family - Ressourcen

Bedingungen in IAM-Richtlinien sind Teil der Syntax, mit der Sie Berechtigungen für AWS Transfer Family Ressourcen angeben. Sie können den Zugriff auf AWS Transfer Family Ressourcen (wie Benutzer, Server, Rollen und andere Entitäten) anhand von Tags auf diesen Ressourcen steuern. Tags sind Schlüssel-Wert-Paare, Weitere Informationen zum Markieren von Ressourcen finden Sie unter Taggen von AWS Ressourcen in der. Allgemeine AWS-Referenz

In AWS Transfer Family können Ressourcen Tags haben, und einige Aktionen können Tags enthalten. Wenn Sie eine IAM-Richtlinie erstellen, können Sie Markierungs-Bedingungsschlüssel verwenden, um Folgendes zu kontrollieren:

- Welche Benutzer anhand der Tags, die die AWS Transfer Family Ressource besitzt, Aktionen für eine Ressource ausführen können.
- Welche Tags in der Anforderung einer Aktion übergeben werden können.
- Ob bestimmte Tag-Schlüssel in einer Anforderung verwendet werden können.

Durch die Verwendung der tagbasierten Zugriffskontrolle können Sie eine genauere Kontrolle als auf API-Ebene anwenden. Sie können auch eine dynamischere Steuerung anwenden als mit einer ressourcenbasierten Zugriffskontrolle. Sie können IAM-Richtlinien erstellen, die einen Vorgang auf der Grundlage der in der Anfrage angegebenen Tags (Anforderungs-Tags) zulassen oder verweigern. Sie können IAM-Richtlinien auch auf der Grundlage von Tags auf der Ressource erstellen, auf der gearbeitet wird (Ressourcen-Tags). Im Allgemeinen sind Ressourcen-Tags für Tags vorgesehen, die sich bereits auf Ressourcen befinden. Anforderungs-Tags sind für das Hinzufügen oder Entfernen von Tags zu einer Ressource vorgesehen.

Die vollständige Syntax und Semantik von Tag-Bedingungsschlüsseln finden Sie im IAM-Benutzerhandbuch unter Steuern des Zugriffs auf AWS Ressourcen mithilfe von Ressourcen-Tags. Einzelheiten zur Angabe von IAM-Richtlinien mit API Gateway finden Sie unter Steuern des Zugriffs auf eine API mit IAM-Berechtigungen im API Gateway Developer Guide.

Beispiel 1: Verweigern Sie Aktionen, die auf Ressourcen-Tags basieren

Sie können die Ausführung einer Aktion an einer Ressource auf der Grundlage von Tags verweigern. Die folgende Beispielrichtlinie verweigert`TagResource`,,, `UntagResource``StartServer`, und `DescribeUser` Operationen `StopServer``DescribeServer`, wenn die Benutzer- oder Serverressource mit dem Schlüssel `stage` und dem Wert `prod` gekennzeichnet ist.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": [
                "transfer:TagResource",
                "transfer:UntagResource",
                "transfer:StartServer",
```

```
                "transfer:StopServer",
                "transfer:DescribeServer",
                "transfer:DescribeUser
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/stage": "prod"
                }
            }
        }
    ]
}
```

Beispiel 2: Aktionen auf der Grundlage von Ressourcen-Tags zulassen

Sie können zulassen, dass eine Aktion für eine Ressource ausgeführt wird, die auf Tags basiert. Die folgende Beispielrichtlinie ermöglicht`TagResource`,`UntagResource`,, `StartServer``StopServer`, und `DescribeUser` Operationen`DescribeServer`, wenn die Benutzer- oder Serverressource mit dem Schlüssel `stage` und dem Wert gekennzeichnet ist`prod`.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "transfer:TagResource",
                "transfer:UntagResource",
                "transfer:StartServer",
                "transfer:StopServer",
                "transfer:DescribeServer",
                "transfer:DescribeUser
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/stage": "prod"
                }
            }
        }
    ]
```

```
}
```

Beispiel 3: Verweigern Sie die Erstellung eines Benutzers oder Servers auf der Grundlage von Anforderungs-Tags

Die folgende Beispielrichtlinie enthält zwei Anweisungen. Die erste Anweisung verweigert den `CreateServer` Vorgang für alle Ressourcen, wenn der Kostenstellenschlüssel für das Tag keinen Wert hat.

Die zweite Anweisung lehnt den `CreateServer` Vorgang ab, wenn der Kostenstellenschlüssel für das Tag einen anderen Wert als 1, 2 oder 3 enthält.

> **Note**
>
> Diese Richtlinie ermöglicht das Erstellen oder Löschen einer Ressource, die einen Schlüssel namens `costcenter` und den Wert 12, oder 3 enthält.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": [
                "transfer:CreateServer"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "Null":  {
                    "aws:RequestTag/costcenter": "true"
                }
            }
        },
        {
            "Effect": "Deny",
            "Action": "transfer:CreateServer",
            "Resource": [
                "*"
            ],
            "Condition": {
```

```
                "ForAnyValue:StringNotEquals": {
                    "aws:RequestTag/costcenter": [
                        "1",
                        "2",
                        "3"
                    ]
                }
            }
        }
    ]
}
```

## Problembehandlung bei AWS Transfer Family Identität und Zugriff

Verwenden Sie die folgenden Informationen, um häufig auftretende Probleme zu diagnostizieren und zu beheben, die bei der Arbeit mit AWS Transfer Family IAM auftreten können.

Themen

- Ich bin nicht berechtigt, eine Aktion durchzuführen in AWS Transfer Family
- Ich bin nicht berechtigt, iam auszuführen: PassRole
- Ich möchte Personen außerhalb meines AWS Kontos den Zugriff auf meine AWS Transfer Family Ressourcen ermöglichen

### Ich bin nicht berechtigt, eine Aktion durchzuführen in AWS Transfer Family

Wenn Ihnen AWS Management Console mitgeteilt wird, dass Sie nicht berechtigt sind, eine Aktion durchzuführen, müssen Sie sich an Ihren Administrator wenden, um Unterstützung zu erhalten. Ihr Administrator hat Ihnen Ihre Anmeldeinformationen zur Verfügung gestellt.

Der folgende Beispielfehler tritt auf, wenn der `mateojackson`-IAM-Benutzer versucht, die Konsole zum Anzeigen von Details zu einem *`Widget`* zu verwenden, jedoch nicht über `transfer:`*`GetWidget`*-Berechtigungen verfügt.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 transfer:GetWidget on resource: my-example-widget
```

In diesem Fall bittet Mateo seinen Administrator um die Aktualisierung seiner Richtlinien, um unter Verwendung der Aktion *`my-example-widget`* auf die Ressource `transfer;:`*`GetWidget`* zugreifen zu können.

## Ich bin nicht berechtigt, iam auszuführen: PassRole

Wenn Sie die Fehlermeldung erhalten, dass Sie nicht zum Durchführen der `iam:PassRole`-Aktion autorisiert sind, müssen Ihre Richtlinien aktualisiert werden, um eine Rolle an AWS Transfer Familyübergeben zu können.

Einige AWS-Services ermöglichen es Ihnen, eine bestehende Rolle an diesen Dienst zu übergeben, anstatt eine neue Servicerolle oder eine dienstverknüpfte Rolle zu erstellen. Hierzu benötigen Sie Berechtigungen für die Übergabe der Rolle an den Dienst.

Der folgende Beispielfehler tritt auf, wenn ein IAM-Benutzer mit dem Namen `marymajor` versucht, die Konsole zu verwenden, um eine Aktion in AWS Transfer Family auszuführen. Die Aktion erfordert jedoch, dass der Service über Berechtigungen verfügt, die durch eine Servicerolle gewährt werden. Mary besitzt keine Berechtigungen für die Übergabe der Rolle an den Dienst.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

In diesem Fall müssen die Richtlinien von Mary aktualisiert werden, um die Aktion `iam:PassRole` ausführen zu können.

Wenn Sie Hilfe benötigen, wenden Sie sich an Ihren AWS Administrator. Ihr Administrator hat Ihnen Ihre Anmeldeinformationen zur Verfügung gestellt.

Die folgende Beispielrichtlinie enthält die Erlaubnis, eine Rolle an zu übergeben AWS Transfer Family.

```
{
   "Version": "2012-10-17",
   "Statement": [
      { "Action": "iam:PassRole",
        "Resource": "arn:aws::iam::123456789012:role/*",
        "Effect": "Allow"
      }
   ]
}
```

### Ich möchte Personen außerhalb meines AWS Kontos den Zugriff auf meine AWS Transfer Family 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 diese Funktionen AWS Transfer Family unterstützt werden, finden Sie unterWie AWS Transfer Family funktioniert mit IAM.
- Informationen dazu, wie Sie Zugriff auf Ihre Ressourcen gewähren können, AWS-Konten die Ihnen gehören, finden Sie im IAM-Benutzerhandbuch unter Gewähren des Zugriffs auf einen IAM-Benutzer in einem anderen AWS-Konto , den Sie besitzen.
- Informationen dazu, wie Sie Dritten Zugriff auf Ihre Ressourcen gewähren können AWS-Konten, finden Sie AWS-Konten im IAM-Benutzerhandbuch unter Gewähren des Zugriffs für Dritte.
- Informationen dazu, wie Sie über einen Identitätsverbund Zugriff gewähren, finden Sie unter Gewähren von Zugriff für extern authentifizierte Benutzer (Identitätsverbund) im IAM-Benutzerhandbuch.
- Informationen zum Unterschied zwischen der Verwendung von Rollen und ressourcenbasierten Richtlinien für den kontoübergreifenden Zugriff finden Sie unter So unterscheiden sich IAM-Rollen von ressourcenbasierten Richtlinien im IAM-Benutzerhandbuch.

## Konformitätsprüfung für AWS Transfer Family

Externe Prüfer bewerten die Sicherheit und Einhaltung von Vorschriften im AWS Transfer Family Rahmen mehrerer AWS Compliance-Programme. Zu diesen Programmen gehören SOC, PCI, HIPAA und andere. Die vollständige Liste finden Sie unter AWS Services in Scope nach Compliance-Programmen.

Eine Liste der AWS Dienstleistungen im Rahmen bestimmter Compliance-Programme finden Sie unter AWS Dienstleistungen im Umfang der einzelnen Compliance-Programme. Allgemeine Informationen finden Sie unter AWS -Compliance-Programme.

Sie können Prüfberichte von Drittanbietern unter herunterladen AWS Artifact. Weitere Informationen finden Sie unter Berichte herunterladen in AWS Artifact.

Ihre Verantwortung für die Einhaltung der Vorschriften bei der Nutzung AWS Transfer Family hängt von der Vertraulichkeit Ihrer Daten, den Compliance-Zielen Ihres Unternehmens und den geltenden Gesetzen und Vorschriften ab. AWS stellt die folgenden Ressourcen zur Verfügung, die Sie bei der Einhaltung der Vorschriften unterstützen:

- Schnellstartanleitungen zu Sicherheit und Compliance — In diesen Bereitstellungsleitfäden werden architektonische Überlegungen erörtert und Schritte für die Implementierung von Umgebungen beschrieben, auf denen auf Sicherheit und Compliance ausgerichtete Basisumgebungen eingerichtet werden. AWS
- Whitepaper „Architecting for HIPAA“ zu Sicherheit und Compliance — In diesem Whitepaper wird beschrieben, wie Unternehmen HIPAA-konforme Anwendungen entwickeln können. AWS
- AWS -Compliance-Ressourcen – Diese Arbeitsbücher und Leitfäden könnten für Ihre Branche und Ihren Standort interessant sein.
- AWS Config— Dieser AWS Service bewertet, wie gut Ihre Ressourcenkonfigurationen den internen Praktiken, Branchenrichtlinien und Vorschriften entsprechen.
- AWS Security Hub— Dieser AWS Service bietet einen umfassenden Überblick über Ihren Sicherheitsstatus und hilft Ihnen AWS , die Einhaltung der Sicherheitsstandards und bewährten Verfahren der Sicherheitsbranche zu überprüfen.

## Resilienz in AWS Transfer Family

Die AWS globale Infrastruktur basiert auf AWS Regionen und Availability Zones. AWS Regionen bieten mehrere physisch getrennte und isolierte Availability Zones, die über Netzwerke mit niedriger Latenz, hohem Durchsatz und hoher Redundanz miteinander verbunden sind. Mithilfe von Availability Zones können Sie Anwendungen und Datenbanken erstellen und ausführen, die automatisch Failover zwischen Availability Zones ausführen, ohne dass es zu Unterbrechungen kommt. Availability Zones sind besser hoch verfügbar, fehlertoleranter und skalierbarer als herkömmliche Infrastrukturen mit einem oder mehreren Rechenzentren.

AWS Transfer Family unterstützt bis zu 3 Availability Zones und wird durch eine automatisch skalierbare, redundante Flotte für Ihre Verbindungs- und Übertragungsanfragen unterstützt.

Beachten Sie Folgendes:

- Für öffentliche Endpunkte:
  - Verfügbarkeit Redundanz auf Zonenebene ist in den Service integriert
  - Für jede AZ gibt es redundante Flotten.
  - Diese Redundanz wird automatisch bereitgestellt
- Informationen zu Endpunkten in einer Virtual Private Cloud (VPC) finden Sie unter. Erstellen Sie einen Server in einer virtuellen privaten Cloud

Informationen finden Sie auch unter:

- Weitere Informationen zu Availability Zones AWS-Regionen und Availability Zones finden Sie unter AWS Globale Infrastruktur.
- Ein Beispiel dafür, wie Sie mithilfe von latenzbasiertem Routing eine höhere Redundanz erreichen und die Netzwerklatenz minimieren können, finden Sie im Blogbeitrag Minimiere die Netzwerklatenz mit Ihren Servern. AWS Transfer Family

# Sicherheit der Infrastruktur in AWS Transfer Family

Als verwalteter Dienst AWS Transfer Family ist er durch AWS globale Netzwerksicherheit geschützt. Informationen zu AWS Sicherheitsdiensten und zum AWS Schutz der Infrastruktur finden Sie unter AWS Cloud-Sicherheit. Informationen zum Entwerfen Ihrer AWS Umgebung unter Verwendung der bewährten Methoden für die Infrastruktursicherheit finden Sie unter Infrastructure Protection in Security Pillar AWS Well-Architected Framework.

Sie verwenden AWS veröffentlichte API-Aufrufe für den Zugriff AWS Transfer Family über das Netzwerk. 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.

# Fügen Sie eine Firewall für Webanwendungen hinzu

AWS WAF ist eine Firewall für Webanwendungen, die hilft, Webanwendungen und APIs vor Angriffen zu schützen. Sie können damit eine Reihe von Regeln konfigurieren, die als Web Access Control List (Web ACL) bezeichnet werden und Webanfragen auf der Grundlage von anpassbaren Websicherheitsregeln und -bedingungen, die Sie definieren, zulassen, blockieren oder zählen. Weitere Informationen finden Sie unter Verwenden AWS WAF zum Schutz Ihrer APIs.

Um hinzuzufügen AWS WAF

1. Öffnen Sie die API Gateway-Konsole unter https://console.aws.amazon.com/apigateway/.
2. Wählen Sie im API-Navigationsbereich Ihre benutzerdefinierte Identitätsanbieter-Vorlage aus.
3. Wählen Sie Stages.
4. Wählen Sie im Bereich Stages den Namen der Stufe aus.
5. Wählen Sie im Bereich Stage Editor die Registerkarte Settings.
6. Führen Sie eine der folgenden Aktionen aus:

   - Wählen Sie unter Web Application Firewall (WAF) für Web ACL die Web-ACL aus, die Sie dieser Phase zuordnen möchten.
   - Wenn die von Ihnen benötigte Web-ACL nicht existiert, müssen Sie eine erstellen, indem Sie wie folgt vorgehen:
     1. Wählen Sie Web-ACL erstellen.
     2. Wählen Sie auf der Startseite des AWS WAF-Service die Option Web-ACL erstellen aus.
     3. Geben Sie unter Web-ACL-Details für Name den Namen der Web-ACL ein.
     4. Wählen Sie unter Regeln die Option Regeln hinzufügen und anschließend Meine eigenen Regeln und Regelgruppen hinzufügen aus.
     5. Wählen Sie unter Regeltyp die Option IP-Set aus, um eine bestimmte Liste von IP-Adressen zu identifizieren.
     6. Geben Sie unter Regel den Namen der Regel ein.
     7. Wählen Sie für IP-Set einen vorhandenen IP-Satz aus. Informationen zum Erstellen eines IP-Sets finden Sie unter Einen IP-Satz erstellen.
     8. Wenn Sie die IP-Adresse als ursprüngliche Adresse verwenden möchten, wählen Sie im Header die Option IP-Adresse aus.
     9. Geben Sie als Namen des Header-Felds ein`SourceIP`.

10. Wählen Sie für Position im Header die Option Erste IP-Adresse aus.
11. Wählen Sie für Fallback bei fehlender IP-Adresse die Option Match oder No Match aus, je nachdem, wie Sie mit einer ungültigen (oder fehlenden) IP-Adresse im Header umgehen möchten.
12. Wählen Sie unter Aktion die Aktion des IP-Sets aus.
13. Wählen Sie für Standard-Web-ACL-Aktion für Anfragen, die keiner Regel entsprechen, die Option Zulassen oder Blockieren aus und klicken Sie dann auf Weiter.
14. Wählen Sie für die Schritte 4 und 5 Weiter aus.
15. Überprüfen Sie unter Überprüfen und erstellen Ihre Auswahl und wählen Sie dann Web-ACL erstellen aus.

7. Wählen Sie Save Changes.
8. Wählen Sie Resources aus.
9. Wählen Sie für Aktionen die Option Deploy API aus.

Informationen zur Sicherheit AWS Transfer Family mit der AWS Web Application Firewall finden Sie im AWS Storage-Blog unter Absichern AWS Transfer Family mit AWS Anwendungs-Firewall und Amazon API Gateway.

## Serviceübergreifende Confused-Deputy-Prävention

Das Problem des verwirrten Stellvertreters ist ein Sicherheitsproblem, bei dem eine Entität, die keine Berechtigung zur Durchführung einer Aktion hat, eine privilegiertere Entität zur Durchführung der Aktion zwingen kann. Bei AWS dienststellenübergreifendem Identitätswechsel kann es zu einem Problem mit verwirrtem Stellvertreter kommen. Ein serviceübergreifender Identitätswechsel kann auftreten, wenn ein Service (der Anruf-Service) einen anderen Service anruft (den aufgerufenen Service). Der anrufende Service kann so manipuliert werden, dass er seine Berechtigungen nutzt, um auf die Ressourcen eines anderen Kunden so einzuwirken, dass er sonst keine Zugriffsberechtigung hätte. Um dies zu verhindern, bietet AWS Tools, mit denen Sie Ihre Daten für alle Services mit Serviceprinzipalen schützen können, die Zugriff auf Ressourcen in Ihrem Konto erhalten haben. Eine ausführliche Beschreibung dieses Problems finden Sie im IAM-Benutzerhandbuch unter The Confused Deputy Problem.

Wir empfehlen, die Kontextschlüssel `aws:SourceArn`und die `aws:SourceAccount`globalen Bedingungsschlüssel in Ressourcenrichtlinien zu verwenden, um die Berechtigungen einzuschränken, die AWS Transfer Family für die Ressource hat. Wenn Sie beide globalen

Bedingungskontextschlüssel verwenden, müssen der `aws:SourceAccount`-Wert und das Konto im `aws:SourceArn`-Wert dieselbe Konto-ID verwenden, wenn sie in derselben Richtlinienanweisung verwendet werden.

Der effektivste Weg, um sich vor dem Confused-Deputy-Problem zu schützen, ist die Verwendung des exakten Amazon-Ressourcennamens (ARN) der Ressource, die Sie zulassen möchten. Wenn Sie mehrere Ressourcen angeben, verwenden Sie den `aws:SourceArn` globalen Kontextbedingungsschlüssel mit Platzhalterzeichen (`*`) für die unbekannten Teile des ARN. z. B. `arn:aws:transfer::region::account-id:server/*`.

AWS Transfer Family verwendet die folgenden Rollentypen:

- Benutzerrolle — Ermöglicht dienstverwalteten Benutzern den Zugriff auf die erforderlichen Transfer Family Family-Ressourcen. AWS Transfer Family übernimmt diese Rolle im Kontext eines Transfer Family Family-Benutzer-ARN.
- Zugriffsrolle — Ermöglicht den Zugriff nur auf die Amazon S3 S3-Dateien, die übertragen werden. Für eingehende AS2-Übertragungen verwendet die Zugriffsrolle den Amazon-Ressourcennamen (ARN) für die Vereinbarung. Für ausgehende AS2-Übertragungen verwendet die Zugriffsrolle den ARN für den Connector.
- Aufrufrolle — Zur Verwendung mit Amazon API Gateway als benutzerdefiniertem Identitätsanbieter des Servers. Transfer Family übernimmt diese Rolle im Kontext eines Transfer Family Family-Servers ARN.
- Rolle „Protokollierung“ — Wird verwendet, um Einträge bei Amazon zu protokollieren CloudWatch. Transfer Family verwendet diese Rolle, um Erfolgs- und Fehlschlagsdetails sowie Informationen zu Dateiübertragungen zu protokollieren. Transfer Family übernimmt diese Rolle im Kontext eines Transfer Family Family-Servers ARN. Für ausgehende AS2-Übertragungen verwendet die Protokollierungsrolle den Connector-ARN.
- Ausführungsrolle — Ermöglicht es einem Transfer Family Family-Benutzer, Workflows aufzurufen und zu starten. Transfer Family übernimmt diese Rolle im Zusammenhang mit einem Transfer Family Family-Workflow-ARN.

Weitere Informationen finden Sie unter Richtlinien und Berechtigungen in IAM im -IAM-Benutzerhandbuch.

> **Note**
>
> Ersetzen Sie in den folgenden Beispielen alle *Platzhalter für Benutzereingabe* durch Ihre eigenen Informationen.

> **Note**
>
> In unseren Beispielen verwenden wir `ArnLike` sowohl als auch`ArnEquals`. Sie sind funktionell identisch, weshalb Sie beide verwenden können, wenn Sie Ihre Richtlinien erstellen. Die Dokumentation Transfer Family verwendet`ArnLike`, wenn die Bedingung ein Platzhalterzeichen enthält, und `ArnEquals` um eine exakte Übereinstimmungsbedingung anzugeben.

## AWS Transfer Family Benutzerrolle dienstübergreifend verwirrter Stellvertreter Prävention

Die folgende Beispielrichtlinie ermöglicht es jedem Benutzer eines beliebigen Servers im Konto, die Rolle zu übernehmen.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "",
            "Effect": "Allow",
            "Principal": {
                "Service": "transfer.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "account-id"
                },
                "ArnLike": {
                    "aws:SourceArn": "arn:aws:transfer:region:account-id:user/*"
                }
            }
        }
```

```
    ]
}
```

Die folgende Beispielrichtlinie ermöglicht es jedem Benutzer eines bestimmten Servers, die Rolle zu übernehmen.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "",
            "Effect": "Allow",
            "Principal": {
                "Service": "transfer.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "account-id"
                },
                "ArnEquals": {
                    "aws:SourceArn": "arn:aws:transfer:region:account-id:user/server-
id/*"
                }
            }
        }
    ]
}
```

Die folgende Beispielrichtlinie ermöglicht es einem bestimmten Benutzer eines bestimmten Servers, die Rolle zu übernehmen.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "",
            "Effect": "Allow",
            "Principal": {
                "Service": "transfer.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
```

```
            "Condition": {
                "ArnLike": {
                    "aws:SourceArn": "arn:aws:transfer:region:account-id:user/server-
id/user-name"
                }
            }
        }
    ]
}
```

## AWS Transfer Family Arbeitsablauf Rolle dienstübergreifend verwirrt stellvertretend Prävention

Mit der folgenden Beispielrichtlinie kann jeder Workflow im Konto diese Rolle übernehmen.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "",
            "Effect": "Allow",
            "Principal": {
                "Service": "transfer.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "account-id"
                },
                "ArnLike": {
                    "aws:SourceArn": "arn:aws:transfer:region:account-id:workflow/*"
                }
            }
        }
    ]
}
```

Die folgende Beispielrichtlinie ermöglicht es einem bestimmten Workflow, die Rolle zu übernehmen.

```
{
    "Version": "2012-10-17",
    "Statement": [
```

```
        {
            "Sid": "",
            "Effect": "Allow",
            "Principal": {
                "Service": "transfer.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "ArnLike": {
                    "aws:SourceArn": "arn:aws:transfer:region:account-
id:workflow/workflow-id"
                }
            }
        }
    ]
}
```

## AWS Transfer Family Familienprotokollierung und Anrufrolle dienstübergreifend verwirrte Stellvertreterprävention

> **Note**
>
> Die folgenden Beispiele können sowohl für Protokollierungs- als auch für Aufrufrollen verwendet werden.
> In diesen Beispielen können Sie die ARN-Details für einen Workflow entfernen, wenn an Ihren Server keine Workflows angehängt sind.

Das folgende Beispiel für eine Protokollierungs-/Aufrufreichtlinie ermöglicht es jedem Server (und Workflow) im Konto, die Rolle zu übernehmen.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowAllServersWithWorkflowAttached",
            "Effect": "Allow",
            "Principal": {
                "Service": "transfer.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
```

```
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "account-id"
                },
                "ArnLike": {
                   "aws:SourceArn": [
                     "arn:aws:transfer:region:account-id:server/*",
                     "arn:aws:transfer:region:account-id:workflow/*"
                   ]
                }
            }
        }
    ]
}
```

Das folgende Beispiel für eine Protokollierungs-/Aufruf-Richtlinie ermöglicht es einem bestimmten Server (und Workflow), die Rolle zu übernehmen.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowSpecificServerWithWorkflowAttached",
            "Effect": "Allow",
            "Principal": {
                "Service": "transfer.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "aws:SourceAccount": "account-id"
                },
                "ArnEquals": {
                   "aws:SourceArn": [
                       "arn:aws:transfer:region:account-id:server/server-id",
                       "arn:aws:transfer:region:account-id:workflow/workflow-id"
                   ]
                }
            }
        }
    ]
}
```

# AWS verwaltete Richtlinien für AWS Transfer Family

Um Benutzern, Gruppen und Rollen Berechtigungen hinzuzufügen, ist es einfacher, AWS verwaltete Richtlinien zu verwenden, als Richtlinien selbst zu schreiben. Es erfordert Zeit und Fachwissen, kundenverwaltete AWS Identity and Access Management (IAM) Richtlinien zu erstellen, die Ihrem Team nur die Berechtigungen gewähren, die es benötigt. Um schnell loszulegen, können Sie unsere AWS verwalteten Richtlinien verwenden. Diese Richtlinien decken allgemeine Anwendungsfälle ab und sind in Ihrem AWS-Konto verfügbar. Weitere Informationen zu verwalteten AWS -Richtlinien finden Sie unter Verwaltete AWS -Richtlinien im IAM-Leitfaden. Eine detaillierte Liste aller AWS verwalteten Richtlinien finden Sie im Referenzhandbuch für AWS verwaltete Richtlinien.

AWS Dienste verwalten und aktualisieren AWS verwaltete Richtlinien. Sie können die Berechtigungen in AWS verwalteten Richtlinien nicht ändern. Services fügen einer von AWS verwalteten Richtlinien gelegentlich zusätzliche Berechtigungen hinzu, um neue Features zu unterstützen. Diese Art von Update betrifft alle Identitäten (Benutzer, Gruppen und Rollen), an welche die Richtlinie angehängt ist. Services aktualisieren eine von AWS verwaltete Richtlinie am ehesten, ein neues Feature gestartet wird oder neue Vorgänge verfügbar werden. Dienste entfernen keine Berechtigungen aus einer AWS verwalteten Richtlinie, sodass durch Richtlinienaktualisierungen Ihre bestehenden Berechtigungen nicht beeinträchtigt werden.

AWS Unterstützt außerdem verwaltete Richtlinien für Jobfunktionen, die sich über mehrere Dienste erstrecken. Die `ReadOnlyAccess` AWS verwaltete Richtlinie bietet beispielsweise schreibgeschützten Zugriff auf alle AWS Dienste und Ressourcen. Wenn ein Dienst eine neue Funktion startet, werden nur Leseberechtigungen für neue Operationen und Ressourcen AWS hinzugefügt. Eine Liste und Beschreibungen der Richtlinien für Auftragsfunktionen finden Sie in Verwaltete AWS -Richtlinien für Auftragsfunktionen im IAM-Leitfaden.

## AWS verwaltete Richtlinie: AWSTransferConsoleFullAccess

Die `AWSTransferConsoleFullAccess` Richtlinie bietet vollen Zugriff auf Transfer Family über die AWS Management Console.

Details zu Berechtigungen

Diese Richtlinie umfasst die folgenden Berechtigungen.

- `acm:ListCertificates`— Erteilt die Erlaubnis, eine Liste der Amazon Resource Names (ARNs) -Zertifikate und den Domainnamen für jeden ARN abzurufen.

- `ec2:DescribeAddresses`— Erteilt die Erlaubnis, eine oder mehrere Elastic IP-Adressen zu beschreiben.
- `ec2:DescribeAvailabilityZones`— Erteilt die Erlaubnis, eine oder mehrere der Availability Zones zu beschreiben, die Ihnen zur Verfügung stehen.
- `ec2:DescribeNetworkInterfaces`— Erteilt die Erlaubnis, eine oder mehrere elastische Netzwerkschnittstellen zu beschreiben.
- `ec2:DescribeSecurityGroups`— Erteilt die Erlaubnis, eine oder mehrere Sicherheitsgruppen zu beschreiben.
- `ec2:DescribeSubnets`— Erteilt die Erlaubnis, ein oder mehrere Subnetze zu beschreiben.
- `ec2:DescribeVpcs`— Erteilt die Erlaubnis, eine oder mehrere virtuelle private Clouds (VPCs) zu beschreiben.
- `ec2:DescribeVpcEndpoints`— Erteilt die Erlaubnis, einen oder mehrere VPC-Endpunkte zu beschreiben.
- `health:DescribeEventAggregates`— Gibt die Anzahl der Ereignisse für jeden Ereignistyp zurück (Problem, geplante Änderung und Kontobenachrichtigung).
- `iam:GetPolicyVersion`— Erteilt die Berechtigung zum Abrufen von Informationen über eine Version der angegebenen verwalteten Richtlinie, einschließlich des Richtliniendokuments.
- `iam:ListPolicies`— Erteilt die Erlaubnis, alle verwalteten Richtlinien aufzulisten.
- `iam:ListRoles`— Erteilt die Berechtigung, die IAM-Rollen aufzulisten, die das angegebene Pfadpräfix haben.
- `iam:PassRole`— Erteilt die Erlaubnis, eine IAM-Rolle an Transfer Family zu übergeben. Weitere Informationen finden Sie unter Einem Benutzer Berechtigungen zur Übergabe einer Rolle an AWS-Service einen gewähren.
- `route53:ListHostedZones`— Erteilt die Berechtigung, eine Liste der öffentlichen und privaten Hosting-Zonen abzurufen, die der aktuellen Zone zugeordnet sind AWS-Konto.
- `s3:ListAllMyBuckets`— Erteilt die Erlaubnis, alle Buckets aufzulisten, die dem authentifizierten Absender der Anfrage gehören.
- `transfer:*`— Gewährt Zugriff auf die Ressourcen von Transfer Family. Das Sternchen (*) gewährt Zugriff auf alle Ressourcen.

```
{
    "Version": "2012-10-17",
    "Statement": [
```

```
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": "transfer.amazonaws.com"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "acm:ListCertificates",
                "ec2:DescribeAddresses",
                "ec2:DescribeAvailabilityZones",
                "ec2:DescribeNetworkInterfaces",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeSubnets",
                "ec2:DescribeVpcs",
                "ec2:DescribeVpcEndpoints",
                "health:DescribeEventAggregates",
                "iam:GetPolicyVersion",
                "iam:ListPolicies",
                "iam:ListRoles",
                "route53:ListHostedZones",
                "s3:ListAllMyBuckets",
                "transfer:*"
            ],
            "Resource": "*"
        }
    ]
}
```

## AWS verwaltete Richtlinie: AWSTransferFullAccess

Die `AWSTransferFullAccess` Richtlinie bietet vollen Zugriff auf die Dienste von Transfer Family.

Details zu Berechtigungen

Diese Richtlinie umfasst die folgenden Berechtigungen.

- `transfer:*`— Erteilt die Erlaubnis, auf Transfer Family Family-Ressourcen zuzugreifen. Das Sternchen (`*`) gewährt Zugriff auf alle Ressourcen.
- `iam:PassRole`— Erteilt die Erlaubnis, eine IAM-Rolle an Transfer Family zu übergeben. Weitere Informationen finden Sie unter Einem Benutzer Berechtigungen zur Übergabe einer Rolle an AWS-Service einen gewähren.
- `ec2:DescribeAddresses`— Erteilt die Erlaubnis, eine oder mehrere Elastic IP-Adressen zu beschreiben.
- `ec2:DescribeNetworkInterfaces`— Erteilt die Erlaubnis, eine oder mehrere Netzwerkschnittstellen zu beschreiben.
- `ec2:DescribeVpcEndpoints`— Erteilt die Erlaubnis, einen oder mehrere VPC-Endpunkte zu beschreiben.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "transfer:*",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "iam:PassedToService": "transfer.amazonaws.com"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeVpcEndpoints",
                "ec2:DescribeNetworkInterfaces",
                "ec2:DescribeAddresses"
            ],
            "Resource": "*"
        }
```

```
    ]
}
```

## AWS verwaltete Richtlinie: AWSTransferLoggingAccess

Die `AWSTransferLoggingAccess` Richtlinie gewährt AWS Transfer Family vollen Zugriff, um Protokollstreams und Gruppen zu erstellen und Protokollereignisse in Ihrem Konto zu speichern.

Details zu Berechtigungen

Diese Richtlinie umfasst die folgenden Berechtigungen für Amazon CloudWatch Logs.

- `CreateLogStream`— Erteilt Prinzipalen die Erlaubnis, einen Protokollstream zu erstellen.
- `DescribeLogStreams`— Erteilt Prinzipalen die Berechtigung, die Protokollstreams für die Protokollgruppe aufzulisten.
- `CreateLogGroup`— Erteilt Prinzipalen Berechtigungen zum Erstellen von Protokollgruppen.
- `PutLogEvents`— Erteilt Prinzipalen die Berechtigung, einen Stapel von Protokollereignissen in einen Protokollstream hochzuladen.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogStream",
                "logs:DescribeLogStreams",
                "logs:CreateLogGroup",
                "logs:PutLogEvents"
            ],
            "Resource": "*"
        }
    ]
}
```

## AWS verwaltete Richtlinie: AWSTransferReadOnlyAccess

Die `AWSTransferReadOnlyAccess` Richtlinie bietet Lesezugriff auf die Transfer Family Family-Dienste.

Details zu Berechtigungen

Diese Richtlinie beinhaltet die folgenden Berechtigungen für Transfer Family.

- `DescribeUser`— Erteilt Prinzipalen die Erlaubnis, die Beschreibungen für Benutzer einzusehen.
- `DescribeServer`— Erteilt Prinzipalen die Erlaubnis, die Beschreibungen für Server einzusehen.
- `ListUsers`— Erteilt Prinzipalen die Berechtigung, Benutzer für einen Server aufzulisten.
- `ListServers`— Erteilt Prinzipalen die Erlaubnis, die Server für das Konto aufzulisten.
- `TestIdentityProvider`— Erteilt Prinzipalen Berechtigungen, um zu testen, ob der konfigurierte Identitätsanbieter korrekt eingerichtet ist.
- `ListTagsForResource`— Erteilt Prinzipalen die Berechtigung, die Tags für eine Ressource aufzulisten.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "transfer:DescribeUser",
                "transfer:DescribeServer",
                "transfer:ListUsers",
                "transfer:ListServers",
                "transfer:TestIdentityProvider",
                "transfer:ListTagsForResource"
            ],
            "Resource": "*"
        }
    ]
}
```

## AWS Familienupdates auf AWS verwaltete Richtlinien übertragen

Sehen Sie sich Details zu Aktualisierungen der AWS verwalteten Richtlinien für AWS Transfer Family an, seit dieser Dienst begonnen hat, diese Änderungen zu verfolgen. Um automatische Warnungen über Änderungen an dieser Seite zu erhalten, abonnieren Sie den RSS-Feed auf der Dokumenthistorie für AWS Transfer Family-Seite.

| Änderung | Beschreibung | Datum |
| --- | --- | --- |
| Aktualisierung der Dokumentation | Es wurden Abschnitte für jede der von Transfer Family verwalteten Richtlinien hinzugefügt. | 27. Januar 2022 |
| AWSTransferReadOnlyAccess – Aktualisierung auf eine bestehende Richtlinie | AWS Transfer Family hat neue Berechtigungen hinzugefügt, damit die Richtlinie gelesen werden kann AWS Managed Microsoft AD. | 30. September 2021 |
| AWS Transfer Family hat begonnen, Änderungen zu verfolgen | AWS Transfer Family hat damit begonnen, Änderungen an seinen AWS verwalteten Richtlinien nachzuverfolgen. | 15. Juni 2021 |

# Problembehebung AWS Transfer Family

Verwenden Sie die folgenden Informationen, um häufig auftretende Probleme zu diagnostizieren und zu beheben, die bei der Arbeit mit AWS Transfer Family Problemen auftreten können.

Informationen zu Problemen mit IAM in Transfer Family finden Sie unterProblembehandlung bei AWS Transfer Family Identität und Zugriff.

Themen


- Problembehandlung für vom Service verwaltete Benutzer
- Probleme mit Amazon API Gateway beheben
- Richtlinien zur Fehlerbehebung für verschlüsselte Amazon S3 S3-Buckets
- Beheben Sie Probleme mit der Authentifizierung
- Beheben Sie Probleme mit verwalteten Workflows
- Beheben Sie Probleme mit der Workflow-Entschlüsselung
- Probleme mit Amazon EFS beheben
- Beheben Sie Fehler beim Testen Ihres Identitätsanbieters
- Beheben Sie Probleme beim Hinzufügen vertrauenswürdiger Hostschlüssel für Ihren SFTP-Connector
- Beheben Sie Probleme beim Hochladen von Dateien
- Beheben Sie die Ausnahme ResourceNotFound
- Beheben Sie Probleme mit dem SFTP-Connector
- Beheben Sie AS2-Probleme


## Problembehandlung für vom Service verwaltete Benutzer

In diesem Abschnitt werden mögliche Lösungen für die folgenden Probleme beschrieben.

Themen


- Problembehandlung für vom Service verwaltete Amazon EFS-Benutzer
- Die Problembehandlung für den öffentlichen Schlüsseltext dauert zu lange
- Fehler beim Hinzufügen des öffentlichen SSH-Schlüssels zur Fehlerbehebung

## Problembehandlung für vom Service verwaltete Amazon EFS-Benutzer

Beschreibung

Sie führen den `sftp` Befehl aus und die Eingabeaufforderung wird nicht angezeigt. Stattdessen wird die folgende Meldung angezeigt:

```
Couldn't canonicalize: Permission denied
           Need cwd
```

Ursache

Die Rolle Ihres AWS Identity and Access Management (IAM-) Benutzers ist nicht berechtigt, auf Amazon Elastic File System (Amazon EFS) zuzugreifen.

Lösung

Erhöhen Sie die Richtlinienberechtigungen für die Rolle Ihres Benutzers. Sie können eine AWS verwaltete Richtlinie hinzufügen, `AmazonElasticFileSystemClientFullAccess` z.

## Die Problembehandlung für den öffentlichen Schlüsseltext dauert zu lange

Beschreibung

Wenn Sie versuchen, einen vom Dienst verwalteten Benutzer zu erstellen, wird die folgende Fehlermeldung angezeigt:

```
Failed to create user (1 validation error detected:
'sshPublicKeyBody' failed to satisfy constraint: Member must have length less than or
 equal to 2048)
```

Ursache

Möglicherweise geben Sie einen PGP-Schlüssel für den öffentlichen Schlüssel ein. PGP-Schlüssel für vom Dienst verwaltete Benutzer werden AWS Transfer Family nicht unterstützt.

Lösung

Wenn der PGP-Schlüssel RSA-basiert ist, können Sie ihn in das PEM-Format konvertieren. Zum Beispiel stellt Ubuntu hier ein Konvertierungstool zur Verfügung: https://manpages.ubuntu.com/manpages/xenial/man1/openpgp2ssh.1.html

## Fehler beim Hinzufügen des öffentlichen SSH-Schlüssels zur Fehlerbehebung

Beschreibung

Wenn Sie versuchen, einen öffentlichen Schlüssel für einen vom Service verwalteten Benutzer hinzuzufügen, wird die folgende Fehlermeldung angezeigt:

```
Failed to add SSH public key (Unsupported or invalid SSH public key format)
```

Ursache

Möglicherweise versuchen Sie, einen öffentlichen Schlüssel im SSH2-Format zu importieren. Öffentliche Schlüssel im SSH2-Format werden für vom Dienst verwaltete Benutzer AWS Transfer Family nicht unterstützt.

Lösung

Sie müssen den Schlüssel in das OpenSSH-Format konvertieren. Dieser Vorgang wird unter beschrieben. Konvertiert einen öffentlichen SSH2-Schlüssel in das PEM-Format

# Probleme mit Amazon API Gateway beheben

In diesem Abschnitt werden mögliche Lösungen für die folgenden API-Gateway-Probleme beschrieben.

Themen

- Zu viele Authentifizierungsfehler
- Die Verbindung wurde geschlossen

## Zu viele Authentifizierungsfehler

Beschreibung

Wenn Sie versuchen, mithilfe des Secure Shell (SSH) File Transfer Protocol (SFTP) eine Verbindung zu Ihrem Server herzustellen, wird die folgende Fehlermeldung angezeigt:

```
Received disconnect from 3.15.127.197 port 22:2: Too many authentication failures
Authentication failed.
Couldn't read packet: Connection reset by peer
```

Ursache

Möglicherweise haben Sie ein falsches Passwort für Ihren Benutzer eingegeben. Versuchen Sie erneut, das richtige Passwort einzugeben.

Wenn das Passwort korrekt ist, kann das Problem durch eine ungültige Rolle mit dem Amazon Resource Name (ARN) verursacht werden. Um zu bestätigen, dass dies das Problem ist, testen Sie den Identitätsanbieter für Ihren Server. Wenn Sie eine Antwort ähnlich der folgenden sehen, ist der Rollen-ARN nur ein Platzhalter, wie durch den Rollen-ID-Wert aller Nullen angezeigt wird:

```
{
    "Response": "{\"Role\": \"arn:aws:iam::000000000000:role/MyUserS3AccessRole\",
\"HomeDirectory\": \"/\"}",
    "StatusCode": 200,
    "Message": "",
    "Url": "https://api-gateway-ID.execute-api.us-east-1.amazonaws.com/prod/
servers/transfer-server-ID/users/myuser/config"
}
```

Lösung

Ersetzen Sie die Platzhalterrolle ARN durch eine tatsächliche Rolle, die berechtigt ist, auf den Server zuzugreifen.

So aktualisieren Sie die Rolle

1. Öffnen Sie die AWS CloudFormation Konsole unter https://console.aws.amazon.com/cloudformation.
2. Wählen Sie im linken Navigationsbereich Stack aus.
3. Wählen Sie in der Liste „Stacks“ Ihren Stack und anschließend den Tab „Parameter“ aus.
4. Wählen Sie Aktualisieren. Wählen Sie auf der Seite Stack aktualisieren die Option Aktuelle Vorlage verwenden und dann Weiter aus.
5. UserRoleArnErsetzen Sie es durch einen Rollen-ARN, der über ausreichende Berechtigungen für den Zugriff auf Ihren Transfer Family Family-Server verfügt.

> **Note**
>
> Um die erforderlichen Berechtigungen zu gewähren, können Sie die `AmazonAPIGatewayAdministrator` und die `AmazonS3FullAccess` verwalteten Richtlinien zu Ihrer Rolle hinzufügen.

6. Wählen Sie Weiter und dann erneut Weiter. Wählen Sie auf der Seite „***Stack*** überprüfen“ die Option Ich bestätige, dass AWS CloudFormation möglicherweise IAM-Ressourcen erstellt werden, und wählen Sie dann Stack aktualisieren aus.

## Die Verbindung wurde geschlossen

Beschreibung

Wenn Sie versuchen, mithilfe des Secure Shell (SSH) File Transfer Protocol (SFTP) eine Verbindung zu Ihrem Server herzustellen, wird die folgende Fehlermeldung angezeigt:

```
Connection closed
```

Ursache

Eine mögliche Ursache für dieses Problem ist, dass Ihre CloudWatch Amazon-Logging-Rolle kein Vertrauensverhältnis zu Transfer Family hat.

Lösung

Stellen Sie sicher, dass die Protokollierungsrolle für den Server eine Vertrauensbeziehung mit Transfer Family hat. Weitere Informationen finden Sie unter So stellen Sie eine Vertrauensbeziehung her.

# Richtlinien zur Fehlerbehebung für verschlüsselte Amazon S3 S3-Buckets

Beschreibung

Sie haben einen verschlüsselten Amazon S3 S3-Bucket, den Sie als Speicher für Ihren Transfer Family Family-Server verwenden. Wenn Sie versuchen, eine Datei auf den Server hochzuladen, erhalten Sie die Fehlermeldung`Couldn't close file: Permission denied`.

Und wenn Sie sich die Serverprotokolle ansehen, werden Ihnen die folgenden Fehler angezeigt:

```
ERROR Message="Access denied" Operation=CLOSE Path=/bucket/user/test.txt BytesIn=13
ERROR Message="Access denied"
```

Ursache

Die Richtlinie für Ihren IAM-Benutzer ist nicht berechtigt, auf den verschlüsselten Bucket zuzugreifen.

Lösung

Sie müssen in Ihrer Richtlinie zusätzliche Berechtigungen angeben, um die erforderlichen AWS Key Management Service (AWS KMS) Berechtigungen zu gewähren. Details hierzu finden Sie unter Datenverschlüsselung in Amazon S3.

# Beheben Sie Probleme mit der Authentifizierung

In diesem Abschnitt werden mögliche Lösungen für die folgenden Authentifizierungsprobleme beschrieben.

Themen

- Authentifizierungsfehler — SSH/SFTP
- Problem mit nicht übereinstimmenden verwalteten AD-Bereichen
- Verschiedene Probleme bei der Authentifizierung

## Authentifizierungsfehler — SSH/SFTP

Beschreibung

Wenn Sie versuchen, mithilfe des Secure Shell (SSH) File Transfer Protocol (SFTP) eine Verbindung zu Ihrem Server herzustellen, erhalten Sie eine Meldung, die der folgenden ähnelt:

```
Received disconnect from 3.130.115.105 port 22:2: Too many authentication failures
  Authentication failed.
```

> **Note**
>
> Wenn Sie ein API Gateway verwenden und dieser Fehler angezeigt wird, finden Sie weitere Informationen unterZu viele Authentifizierungsfehler.

Ursache

Sie haben kein RSA-Schlüsselpaar für Ihren Benutzer hinzugefügt, daher müssen Sie sich stattdessen mit einem Passwort authentifizieren.

Lösung

Wenn Sie den `sftp` Befehl ausführen, geben Sie die `-o PubkeyAuthentication=no` Option an. Diese Option zwingt das System, Ihr Passwort anzufordern. Beispielsweise:

```
sftp -o PubkeyAuthentication=no sftp-user@server-id.server.transfer.region-id.amazonaws.com
```

## Problem mit nicht übereinstimmenden verwalteten AD-Bereichen

Beschreibung

Der Bereich eines Benutzers und sein Gruppenbereich müssen übereinstimmen. Sie müssen sich beide im Standardbereich oder beide im vertrauenswürdigen Bereich befinden.

Ursache

Wenn ein Benutzer und seine Gruppe nicht übereinstimmen, kann der Benutzer nicht von Transfer Family authentifiziert werden. Wenn Sie den Identitätsanbieter für den Benutzer testen, erhalten Sie die Fehlermeldung Kein zugeordneter Zugriff für Benutzergruppen gefunden.

Lösung

Verweisen Sie auf eine Gruppe im Bereich des Benutzers, die dem Gruppenbereich entspricht (entweder Standard oder vertrauenswürdig).

## Verschiedene Probleme bei der Authentifizierung

Beschreibung

Sie erhalten einen Authentifizierungsfehler und keine der anderen Problembehebungsmaßnahmen funktioniert

Ursache

Möglicherweise haben Sie ein Ziel für ein logisches Verzeichnis angegeben, das einen führenden oder abschließenden Schrägstrich (/) enthält.

Lösung

Aktualisieren Sie Ihr logisches Verzeichnisziel, um sicherzustellen, dass es mit einem Schrägstrich beginnt und keinen abschließenden Schrägstrich enthält. Das ist zum Beispiel akzeptabel, `/DOC-EXAMPLE-BUCKET/images` ist es aber auch nicht`DOC-EXAMPLE-BUCKET/images`. `/DOC-EXAMPLE-BUCKET/images/`

# Beheben Sie Probleme mit verwalteten Workflows

In diesem Abschnitt werden mögliche Lösungen für die folgenden Workflow-Probleme beschrieben.

Themen

- Workflow-bezogene Fehler mithilfe von Amazon beheben CloudWatch
- Beheben Sie Fehler beim Kopieren von Workflows

## Workflow-bezogene Fehler mithilfe von Amazon beheben CloudWatch

Beschreibung

Wenn Sie Probleme mit Ihren Workflows haben, können Sie Amazon verwenden, CloudWatch um die Ursache zu untersuchen.

Ursache

Es kann mehrere Ursachen geben. Verwenden Sie Amazon CloudWatch Logs, um Nachforschungen anzustellen.

Lösung

Transfer Family gibt den Status der Workflow-Ausführung in CloudWatch Logs aus. Die folgenden Arten von Workflow-Fehlern können in CloudWatch Protokollen auftreten:

- `"type": "StepErrored"`
- `"type": "ExecutionErrored"`
- `"type": "ExecutionThrottled"`
- `"Service failure on starting workflow"`

Sie können die Ausführungsprotokolle Ihres Workflows mithilfe einer anderen Filter- und Mustersyntax filtern. Sie können beispielsweise einen Protokollfilter in Ihren CloudWatch Protokollen erstellen, um Workflow-Ausführungsprotokolle zu erfassen, die die ExecutionErroredNachricht enthalten. Einzelheiten finden Sie unter Echtzeitverarbeitung von Protokolldaten mit Abonnements und Filter- und Mustersyntax im Amazon CloudWatch Logs-Benutzerhandbuch.

StepErrored

```
2021-10-29T12:57:26.272-05:00
            {"type":"StepErrored","details":
{"errorType":"BAD_REQUEST","errorMessage":"Cannot
            tag Efs file","stepType":"TAG","stepName":"successful_tag_step"},
            "workflowId":"w-
abcdef01234567890","executionId":"1234abcd-56ef-78gh-90ij-1234klmno567",
            "transferDetails":
{"serverId":"s-1234567890abcdef0","username":"lhr","sessionId":"1234567890abcdef0"}
```

`StepErrored`Zeigt an, dass ein Schritt innerhalb des Workflows einen Fehler generiert hat. In einem einzigen Workflow können Sie mehrere Schritte konfigurieren. Dieser Fehler teilt Ihnen mit, in welchem Schritt der Fehler aufgetreten ist, und gibt eine Fehlermeldung aus. In diesem speziellen Beispiel wurde der Schritt so konfiguriert, dass er eine Datei kennzeichnet. Das Taggen einer Datei in einem Amazon EFS-Dateisystem wird jedoch nicht unterstützt, sodass der Schritt einen Fehler generierte.

ExecutionErrored

```
2021-10-29T12:57:26.618-05:00
            {"type":"ExecutionErrored","details":{},"workflowId":"w-w-
abcdef01234567890",
            "executionId":"1234abcd-56ef-78gh-90ij-1234klmno567","transferDetails":
{"serverId":"s-1234567890abcdef0",
            "username":"lhr","sessionId":"1234567890abcdef0"}}
```

Wenn ein Workflow keinen Schritt ausführen kann, generiert er eine `ExecutionErrored` Meldung. Wenn Sie beispielsweise einen einzelnen Schritt in einem bestimmten Workflow konfiguriert haben und der Schritt nicht ausgeführt werden kann, schlägt der gesamte Workflow fehl.

Die Ausführung wurde gedrosselt

Die Ausführung wird gedrosselt, wenn ein Workflow mit einer Geschwindigkeit ausgelöst wird, die schneller ist, als das System unterstützen kann. Diese Protokollmeldung weist darauf hin, dass Sie die Ausführungsrate für Workflows verlangsamen müssen. Wenn Sie nicht in der Lage sind, Ihre Workflow-Ausführungsrate zu reduzieren, wenden Sie sich AWS Support an Kontakt. AWS

Dienstfehler beim Starten des Workflows

Jedes Mal, wenn Sie einen Workflow von einem Server entfernen und ihn durch einen neuen ersetzen oder die Serverkonfiguration aktualisieren (was sich auf die Ausführungsrolle eines Workflows auswirkt), müssen Sie ungefähr 10 Minuten warten, bevor Sie den neuen Workflow ausführen. Der Transfer Family Family-Server speichert die Workflow-Details im Cache, und es dauert 10 Minuten, bis der Server seinen Cache aktualisiert hat.

Darüber hinaus müssen Sie sich von allen aktiven SFTP-Sitzungen abmelden und sich nach Ablauf der 10-minütigen Wartezeit wieder anmelden, um die Änderungen zu sehen.

## Beheben Sie Fehler beim Kopieren von Workflows

Beschreibung

Wenn Sie einen Workflow ausführen, der einen Schritt zum Kopieren der hochgeladenen Datei enthält, kann der folgende Fehler auftreten:

```
{
  "type": "StepErrored", "details": {
    "errorType": "BAD_REQUEST", "errorMessage": "Bad Request (Service: Amazon S3;
 Status Code: 400; Error Code: 400 Bad Request;
    Request ID: request-ID; S3 Extended Request ID: request-ID Proxy: null)",
 "stepType": "COPY", "stepName": "copy-step-name" },
  "workflowId": "workflow-ID",
  "executionId": "execution-ID",
  "transferDetails": {
     "serverId": "server-ID",
     "username": "user-name",
     "sessionId": "session-ID"
```

```
  }
}
```

Ursache

Die Quelldatei befindet sich in einem Amazon S3 S3-Bucket, der sich in einem anderen AWS-Region als dem Ziel-Bucket befindet.

Lösung

Wenn Sie einen Workflow ausführen, der einen Kopierschritt beinhaltet, stellen Sie sicher, dass sich der Quell- und der Ziel-Bucket in demselben AWS-Region befinden.

# Beheben Sie Probleme mit der Workflow-Entschlüsselung

In diesem Abschnitt werden mögliche Lösungen für die folgenden Probleme mit verschlüsselten Workflows beschrieben.

Themen


- Beheben Sie den Fehler für die signierte Verschlüsselungsdatei
- Beheben Sie einen Fehler für einen FIPS-Algorithmus


## Beheben Sie den Fehler für die signierte Verschlüsselungsdatei

Beschreibung

Ihr Entschlüsselungsworkflow schlägt fehl und Sie erhalten die folgende Fehlermeldung:

```
"Encrypted file with signed message unsupported"
```

Ursache

Transfer Family unterstützt derzeit nicht das Signieren von verschlüsselten Dateien.

Lösung

Wenn es in Ihrem PGP-Client eine Option zum Signieren der verschlüsselten Datei gibt, stellen Sie sicher, dass Sie die Auswahl deaktivieren, da Transfer Family derzeit das Signieren verschlüsselter Dateien nicht unterstützt.

# Beheben Sie einen Fehler für einen FIPS-Algorithmus

## Beschreibung

Ihr Entschlüsselungsworkflow schlägt fehl, und die Protokollnachricht sieht wie folgt aus:

```
{
   "type": "StepErrored",
   "details": {
      "errorType": "BAD_REQUEST",
      "errorMessage": "File encryption algorithm not supported with FIPS mode
 enabled.",
      "stepType": "DECRYPT",
      "stepName": "step-name"
   },
   "workflowId": "workflow-ID",
   "executionId": "execution-ID",
   "transferDetails": {
      "serverId": "server-ID",
      "username": "user-name",
      "sessionId": "session-ID"
   }
}
```

## Ursache

Auf Ihrem Transfer Family Family-Server ist der FIPS-Modus aktiviert und es gibt einen zugehörigen Workflow-Schritt „Entschlüsseln“. Beim Verschlüsseln der Dateien vor dem Hochladen auf Ihren Transfer Family Family-Server generiert der Verschlüsselungsclient möglicherweise verschlüsselte Dateien, die nicht von FIPS zugelassene symmetrische Verschlüsselungsalgorithmen verwenden. In einem solchen Szenario ist der Workflow nicht in der Lage, Dateien zu entschlüsseln. Im folgenden Beispiel verwendet GnuPG Version 2.4.0 OCB (einen Nicht-FIPS-Blockchiffriermodus), um Dateien zu verschlüsseln: Dadurch schlägt der Workflow fehl.

## Lösung

Sie müssen den GPG-Schlüssel, mit dem Sie Ihre Dateien verschlüsselt haben, bearbeiten und sie dann erneut verschlüsseln. Das folgende Verfahren beschreibt die Schritte, die Sie ausführen müssen.

Um Ihre PGP-Schlüssel zu bearbeiten

1. Identifizieren Sie den Schlüssel, den Sie bearbeiten müssen, indem Sie `gpg --list-keys`

   Dies gibt eine Liste von Schlüsseln zurück. Jeder Schlüssel hat Details, die den folgenden ähneln:

   ```
   pub   ed25519 2022-07-07 [SC]
         wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
   uid           [ultimate] Mary Major <marymajor@example.com>
   sub   cv25519 2022-07-07 [E]
   ```

2. Identifizieren Sie den Schlüssel, den Sie bearbeiten möchten. In dem im vorherigen Schritt gezeigten Beispiel lautet die ID`wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY`.
3. Führen Sie `gpg --edit-key wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY`.

   Das System antwortet mit Details über das GnuPG-Programm und den angegebenen Schlüssel.
4. Geben Sie an der gpg> Eingabeaufforderung ein. `showpref` Die folgenden Details werden zurückgegeben:

   ```
   [ultimate] (1). Mary Major <marymajor@example.com>
      Cipher: AES256, AES192, AES, 3DES
      AEAD: OCB
      Digest: SHA512, SHA384, SHA256, SHA224, SHA1
      Compression: ZLIB, BZIP2, ZIP, Uncompressed
      Features: MDC, AEAD, Keyserver no-modify
   ```

   Beachten Sie, dass die bevorzugten Algorithmen, die auf dem Schlüssel gespeichert sind, aufgeführt sind.
5. Wir möchten den Schlüssel bearbeiten, um alle Algorithmen außer OCB beizubehalten. Führen Sie den `setpref` Befehl aus und geben Sie alle Algorithmen an, die beibehalten werden sollen:

   ```
   gpg> setpref AES256, AES192,AES,3DES,SHA512, SHA384, SHA256, SHA224, SHA1,ZLIB,
    BZIP2, ZIP, Uncompressed
   ```

   Dadurch werden die folgenden Details zurückgegeben:

   ```
   Set preference list to:
        Cipher: AES256, AES192, AES, 3DES
   ```

```
     AEAD:
     Digest: SHA512, SHA384, SHA256, SHA224, SHA1
     Compression: ZLIB, BZIP2, ZIP, Uncompressed
     Features: MDC, Keyserver no-modify
Really update the preferences? (y/N)
```

6. Geben Sie ein, y um zu aktualisieren, und geben Sie dann Ihr Passwort ein, wenn Sie aufgefordert werden, die Änderung zu bestätigen.
7. Speichern Sie die Änderungen.

```
gpg> save
```

Bevor Sie Ihren Entschlüsselungsworkflow erneut ausführen, müssen Sie Ihre Dateien mit dem bearbeiteten Schlüssel erneut verschlüsseln.

# Probleme mit Amazon EFS beheben

In diesem Abschnitt werden mögliche Lösungen für die folgenden Amazon EFS-Probleme beschrieben.

Themen

- Beheben Sie das fehlende POSIX-Profil
- Problembehandlung bei logischen Verzeichnissen mit Amazon EFS

## Beheben Sie das fehlende POSIX-Profil

Beschreibung

Wenn Sie Amazon EFS-Speicher für Ihren Server und einen benutzerdefinierten Identitätsanbieter verwenden, müssen Sie für Ihre AWS Lambda Funktion ein POSIX-Profil bereitstellen.

Ursache

Eine mögliche Ursache ist, dass die Vorlagen, die wir für die Erstellung einer Amazon API Gateway AWS Lambda Gateway-Methode bereitstellen, derzeit keine POSIX-Informationen enthalten.

Wenn Sie POSIX-Informationen angegeben haben, wird das Format, das Sie für die Bereitstellung der POSIX-Informationen verwendet haben, von Transfer Family möglicherweise nicht korrekt analysiert.

Lösung

Stellen Sie sicher, dass Sie Transfer Family ein JSON-Element für den `PosixProfile` Parameter bereitstellen.

Wenn Sie beispielsweise Python verwenden, könnten Sie die folgende Zeile hinzufügen, in der Sie den `PosixProfile` Parameter analysieren:

```
if PosixProfile:
        response_data["PosixProfile"] = json.loads(PosixProfile)
```

Oder Sie könnten die folgende Zeile hinzufügen, wobei die *`uid-value`* und Ganzzahlen 0 oder größer *`gid-value`* sind, die jeweils die Benutzer-ID (UID) und die Gruppen-ID (GID) darstellen: JavaScript

```
PosixProfile: {"Uid": uid-value, "Gid": gid-value},
```

In diesen Codebeispielen `PosixProfile` wird der Parameter als JSON-Objekt und nicht als Zeichenfolge an Transfer Family gesendet.

Außerdem müssen Sie den `PosixProfile` Parameter darin AWS Secrets Manager wie folgt speichern. Ersetzen Sie *`your-uid`* und *`your-gid`* durch Ihre tatsächlichen Werte für GID und UID.

```
{"Uid": your-uid, "Gid": your-gid, "SecondaryGids": []}
```

## Problembehandlung bei logischen Verzeichnissen mit Amazon EFS

Beschreibung

Wenn das Home-Verzeichnis des Benutzers nicht existiert und er einen `ls` Befehl ausführt, reagiert das System wie folgt:

```
sftp> ls
remote readdir ("/"): No such file or directory
```

Ursache

Wenn Ihr Transfer Family Family-Server Amazon EFS verwendet, muss das Home-Verzeichnis für den Benutzer mit Lese- und Schreibzugriff erstellt werden, bevor der Benutzer in seinem logischen

Home-Verzeichnis arbeiten kann. Der Benutzer kann dieses Verzeichnis nicht selbst erstellen, da ihm die entsprechenden Berechtigungen für `mkdir` sein logisches Home-Verzeichnis fehlen würden.

Lösung

Ein Benutzer mit Administratorzugriff auf das übergeordnete Verzeichnis muss das logische Basisverzeichnis des Benutzers erstellen.

## Beheben Sie Fehler beim Testen Ihres Identitätsanbieters

Beschreibung

Wenn Sie Ihren Identitätsanbieter mithilfe der Konsole oder des `TestIdentityProvider` API-Aufrufs testen, ist das `Response` Feld leer. Beispielsweise:

```
{
    "Response": "{}",
    "StatusCode": 200,
    "Message": ""
}
```

Ursache

Die wahrscheinlichste Ursache ist, dass die Authentifizierung aufgrund eines falschen Benutzernamens oder Passworts fehlgeschlagen ist.

Lösung

Stellen Sie sicher, dass Sie die richtigen Anmeldeinformationen für Ihren Benutzer verwenden, und aktualisieren Sie gegebenenfalls den Benutzernamen oder das Passwort.

## Beheben Sie Probleme beim Hinzufügen vertrauenswürdiger Hostschlüssel für Ihren SFTP-Connector

Beschreibung

Wenn Sie einen SFTP-Connector erstellen oder bearbeiten und einen vertrauenswürdigen Hostschlüssel hinzufügen, wird die folgende Fehlermeldung angezeigt: `Failed to edit connector details (Invalid host key format.)`

Ursache

Wenn Sie einen korrekten öffentlichen Schlüssel einfügen, liegt das Problem möglicherweise darin, dass Sie den `comment` Teil des Schlüssels eingefügt haben. AWS Transfer Family akzeptiert derzeit den Kommentarteil des Schlüssels nicht.

Lösung

Löscht den Kommentarteil des Schlüssels, wenn Sie ihn in das Textfeld einfügen. Gehen Sie beispielsweise davon aus, dass Ihr Schlüssel wie folgt aussieht:

```
ssh-rsa AAAA...== marymajor@dev-dsk-marymajor-1d-c1234567.us-east-1.amazon.com
```

Entfernen Sie den Text, der auf die == Zeichen folgt, und fügen Sie nur den Teil des Schlüssels ein, der == bis einschließlich

```
ssh-rsa AAAA...==
```

# Beheben Sie Probleme beim Hochladen von Dateien

In diesem Abschnitt werden mögliche Lösungen für die folgenden Probleme beim Hochladen von Dateien beschrieben.

Themen

- Fehler beim Hochladen von Amazon S3 S3-Dateien beheben
- Beheben Sie Probleme mit unlesbaren Dateinamen

## Fehler beim Hochladen von Amazon S3 S3-Dateien beheben

Beschreibung

Wenn Sie versuchen, eine Datei mit Transfer Family in den Amazon S3 S3-Speicher hochzuladen, erhalten Sie die folgende Fehlermeldung: AWS Transfer unterstützt keine Direktzugriffsschreibvorgänge auf S3-Objekte.

Ursache

Wenn Sie Amazon S3 für den Speicher Ihres Servers verwenden, unterstützt Transfer Family nicht mehrere Verbindungen für eine einzelne Übertragung.

Lösung

Wenn Ihr Transfer Family Family-Server Amazon S3 als Speicher verwendet, deaktivieren Sie alle Optionen für Ihre Client-Software, die die Verwendung mehrerer Verbindungen für eine einzelne Übertragung erwähnen.

## Beheben Sie Probleme mit unlesbaren Dateinamen

Beschreibung

In einigen Ihrer hochgeladenen Dateien werden beschädigte Dateinamen angezeigt. Benutzer stoßen manchmal auf Probleme mit FTP- und SFTP-Übertragungen, bei denen bestimmte Zeichen in Dateinamen unleserlich dargestellt werden, z. B. Umlaute, Buchstaben mit Akzenten oder bestimmte Schriften wie Chinesisch oder Arabisch.

Ursache

Die FTP- und SFTP-Protokolle ermöglichen zwar, dass die Zeichenkodierung von Dateinamen von Clients ausgehandelt wird, Amazon S3 und Amazon EFS jedoch nicht. Stattdessen benötigen sie die UTF-8-Zeichenkodierung. Infolgedessen werden bestimmte Zeichen nicht korrekt gerendert.

Lösung

Um dieses Problem zu lösen, überprüfen Sie Ihre Client-Anwendung auf die Zeichenkodierung von Dateinamen und stellen Sie sicher, dass sie auf UTF-8 eingestellt ist.

# Beheben Sie die Ausnahme **`ResourceNotFound`**

Beschreibung

Sie erhalten eine Fehlermeldung, dass die Ressource nicht gefunden werden kann. Wenn Sie beispielsweise ausführen`UpdateServer`, wird möglicherweise die folgende Fehlermeldung angezeigt:

```
An error occurred (ResourceNotFoundException) when calling the UpdateServer operation:
 Unknown server
```

Ursache

Es gibt mehrere Gründe für den Empfang einer ResourceNotFoundExceptionNachricht. In den meisten Fällen ist die Ressource, die Sie in Ihrem API-Befehl angegeben haben, nicht vorhanden.

Wenn Sie eine vorhandene Ressource angegeben haben, ist die wahrscheinlichste Ursache, dass sich Ihre Standardregion von der Region für Ihre Ressource unterscheidet. Wenn Ihre Standardregion beispielsweise us-east-1 ist und sich Ihr Transfer Family Family-Server in us-east-2 befindet, erhalten Sie eine Unbekannte Ressourcenausnahme.

Einzelheiten zum Festlegen einer Standardregion finden Sie unter Schnellkonfiguration mit. `aws configure`

Lösung

Fügen Sie Ihrem API-Befehl einen Regionsparameter hinzu, um explizit anzugeben, wo sich eine bestimmte Ressource befindet.

```
aws transfer -describe-server --server-id server-id --region us-east-2
```

# Beheben Sie Probleme mit dem SFTP-Connector

In diesem Abschnitt werden mögliche Lösungen für die folgenden Probleme mit dem SFTP-Connector beschrieben.

Themen

- Die Schlüsselverhandlung schlägt fehl
- Verschiedene Probleme mit dem SFTP-Connector

## Die Schlüsselverhandlung schlägt fehl

Beschreibung

Sie erhalten eine Fehlermeldung, wenn die Schlüsselaustauschverhandlung fehlschlägt. Beispielsweise:

```
Key exchange negotiation failed due to incompatible host key algorithms.
Client offered: [ecdsa-sha2-nistp256, ecdsa-sha2-nistp384,
ecdsa-sha2-nistp521, rsa-sha2-512, rsa-sha2-256] Server offered: [ssh-rsa]
```

Ursache

Dieser Fehler ist darauf zurückzuführen, dass es keine Überschneidung zwischen den vom Server unterstützten Host-Schlüsselalgorithmen und den vom Connector unterstützten Algorithmen gibt.

Lösung

Stellen Sie sicher, dass der Remoteserver mindestens einen der in der Fehlermeldung aufgeführten Client-Host-Schlüsselalgorithmen unterstützt. Eine Liste der unterstützten Algorithmen finden Sie unterAWS Transfer Family Sicherheitsrichtlinien für SFTP-Konnektoren.

## Verschiedene Probleme mit dem SFTP-Connector

Beschreibung

Sie erhalten nach der Ausführung eine Fehlermeldung`StartFileTransfer`, kennen aber die Ursache des Problems nicht, und nach dem API-Aufruf wird nur die Connector-ID zurückgegeben.

Ursache

Dieser Fehler kann mehrere Ursachen haben. Zur Fehlerbehebung empfehlen wir Ihnen, Ihren Connector zu testen und Ihre CloudWatch Logs zu durchsuchen.

Lösung

- Testen Sie Ihren Connector: SieheTesten Sie einen SFTP-Connector. Wenn der Test fehlschlägt, gibt das System eine Fehlermeldung aus, die auf dem Grund basiert, warum der Test fehlgeschlagen ist. In diesem Abschnitt wird beschrieben, wie Sie Ihren Connector entweder von der Konsole aus oder mithilfe des TestConnectionAPI-Befehls testen.
- CloudWatch Protokolle für Ihren Connector anzeigen: SieheBeispiel für Protokolleinträge für SFTP-Konnektoren. Dieses Thema enthält Beispiele für Protokolleinträge für SFTP-Konnektoren und die Benennungskonvention, die Ihnen bei der Suche nach den entsprechenden Protokollen helfen soll.

# Beheben Sie AS2-Probleme

Fehlermeldungen und Tipps zur Fehlerbehebung für Applicability Statement 2 (AS2) -fähige Server werden hier beschrieben:. AS2-Fehlercodes

# API-Referenz

In den folgenden Abschnitten werden die AWS Transfer Family API-Dienstaufrufe, Datentypen, Parameter und Fehler dokumentiert.

Themen


- Willkommen bei der AWS Transfer Family API
- Aktionen
- Datentypen
- API-Anfragen stellen
- Geläufige Parameter
- Häufige Fehler


## Willkommen bei der AWS Transfer Family API

AWS Transfer Family ist ein sicherer Übertragungsservice, mit dem Sie Dateien über die folgenden Protokolle in den und aus dem Amazon Simple Storage Service (Amazon S3) -Speicher übertragen können:

- Secure Shell (SSH) File Transfer Protocol (SFTP)
- Sicheres Dateiübertragungsprotokoll (FTPS)
- Dateiübertragungsprotokoll (FTP)
- Erklärung zur Anwendbarkeit 2 (AS2)

Dateiübertragungsprotokolle werden in Datenaustausch-Workflows in verschiedenen Branchen wie Finanzdienstleistungen, Gesundheitswesen, Werbung und Einzelhandel verwendet. AWS Transfer Family vereinfacht die Migration von Dateiübertragungsworkflows zu AWS.

Um den AWS Transfer Family Dienst zu nutzen, instanziieren Sie einen Server in der AWS Region Ihrer Wahl. Sie können den Server erstellen, verfügbare Server auflisten und Server aktualisieren und löschen. Der Server ist die Entität, von der Dateioperationen angefordert AWS Transfer Family werden. Server verfügen über eine Reihe wichtiger Eigenschaften. Der Server ist eine benannte Instance, die mit einer vom System zugewiesenen `ServerId` identifiziert wird. Sie können einem

Server optional einen Host-Namen (auch benutzerdefiniert) zuweisen. Der Service berechnet alle instanziierten Server (auch solche`OFFLINE`) und die übertragene Datenmenge.

Benutzer müssen dem Server bekannt sein, der Dateioperationen anfordert. Ein über den Benutzernamen identifizierter Benutzer wird dem Server zugewiesen. Benutzernamen werden zur Authentifizierung von Anfragen verwendet. Ein Server kann nur eine Authentifizierungsmethode haben:`AWS_DIRECTORY_SERVICE`, `SERVICE_MANAGEDAWS_LAMBDA`, oder. `API_GATEWAY`

Sie können jeden der folgenden Identitätsanbietertypen verwenden, um Benutzer zu authentifizieren:

- Denn `SERVICE_MANAGED` ein öffentlicher SSH-Schlüssel wird mit den Eigenschaften des Benutzers auf einem Server gespeichert. Ein Benutzer kann einen oder mehrere öffentliche SSH-Schlüssel für die `SERVICE_MANAGED` Authentifizierungsmethode gespeichert haben. Wenn ein Client eine Dateioperation für eine `SERVICE_MANAGED` Methode anfordert, gibt der Client den Benutzernamen und den privaten SSH-Schlüssel an, der authentifiziert wird, und der Zugriff wird gewährt.
- Sie können die Benutzerauthentifizierung und den Zugriff mit Ihren Microsoft Active Directory-Gruppen verwalten, indem Sie die `AWS_DIRECTORY_SERVICE` Authentifizierungsmethode auswählen.
- Sie können eine Verbindung zu einem benutzerdefinierten Identitätsanbieter herstellen, indem Sie AWS Lambda Wählen Sie die `AWS_LAMBDA` Authentifizierungsmethode.
- Sie können Benutzeranforderungen auch mit einer benutzerdefinierten Authentifizierungsmethode authentifizieren, die Benutzerauthentifizierung und Zugriff bereitstellt. Diese Methode basiert darauf, dass Amazon API Gateway Ihren API-Aufruf von Ihrem Identitätsanbieter verwendet, um Benutzeranfragen zu validieren. Diese Methode wird `API_GATEWAY` in API-Aufrufen als Benutzerdefiniert und in der Konsole als Benutzerdefiniert bezeichnet. Sie können mit dieser benutzerdefinierten Methode Benutzer anhand eines Verzeichnis-Services, eines Datenbankname/Passwort-Paars oder eines anderen Mechanismus authentifizieren.

Benutzern wird eine Richtlinie mit einer Vertrauensbeziehung zwischen ihnen und einem Amazon S3 S3-Bucket zugewiesen. Sie können möglicherweise auf den ganzen Bucket oder auf Teile davon zugreifen. Damit ein Server im Namen eines Benutzers handeln kann, muss der Server die Vertrauensbeziehung vom Benutzer erben. Es wird eine AWS Identity and Access Management (IAM-) Rolle erstellt, die die Vertrauensstellung enthält, und dieser Rolle wird eine `AssumeRole` Aktion zugewiesen. Der Server kann dann Dateioperationen ausführen, als wäre er der Benutzer.

Bei Benutzern, für die eine home Verzeichniseigenschaft festgelegt ist, dient dieses Verzeichnis (oder dieser Ordner) als Ziel und Quelle von Dateioperationen. Wenn kein home-Verzeichnis festgelegt ist, wird das `root`-Verzeichnis des Buckets zum Startverzeichnis.

Server, Benutzer und Rollen werden alle anhand ihres Amazon-Ressourcennamens (ARN) identifiziert. Sie können Entitäten mit einem ARN Tags zuweisen, bei denen es sich um Schlüssel-Wert-Paare handelt. Tags sind Metadaten, die verwendet werden können, um diese Entitäten zu gruppieren oder nach ihnen zu suchen. Tags sind beispielsweise für die Buchhaltung nützlich.

Die folgenden Konventionen werden bei AWS Transfer Family ID-Formaten eingehalten:

- `ServerId`-Werte haben das Format `s-01234567890abcdef`.
- `SshPublicKeyId`-Werte haben das Format `key-01234567890abcdef`.

Die Formate von Amazon Resource Name (ARN) haben die folgende Form:

- Für Server haben ARNs die folgende Form`arn:aws:transfer:`*`region`*`:`*`account-id`*`:server/`*`server-id`*.

  Ein Beispiel für einen Server-ARN lautet `arn:aws:transfer:us-east-1:123456789012:server/s-01234567890abcdef`.
- Für Benutzer haben ARNs das Format `arn:aws:transfer:`*`region`*`:`*`account-id`*`:user/`*`server-id`*`/`*`username`*.

  Ein Beispiel ist `arn:aws:transfer:us-east-1:123456789012:user/s-01234567890abcdef/user1`.

Folgende DNS-Einträge (Endpunkte) werden verwendet:

- API-Endpunkte haben das Format `transfer.`*`region`*`.amazonaws.com`.
- Server-Endpunkte haben das Format `server.transfer.`*`region`*`.amazonaws.com`.

Eine Liste der Transfer Family Family-Endpunkte nach AWS Regionen finden Sie unter AWS Transfer Family Endpunkte und Kontingente in der. Allgemeine AWS-Referenz

Diese API-Schnittstellenreferenz für AWS Transfer Family enthält Dokumentation für eine Programmierschnittstelle, die Sie zur Verwaltung verwenden können. AWS Transfer Family Die Referenzstruktur:

- Eine alphabetische Liste der API-Aktionen finden Sie unter Actions.
- Eine alphabetische Liste der Datentypen finden Sie unter. Data Types
- Eine Liste der häufigen Abfrageparameter finden Sie unter Häufige Parameter.
- Beschreibungen der Fehlercodes finden Sie unter Häufige Fehler.

> **Tip**
>
> Anstatt einen Befehl tatsächlich auszuführen, können Sie den `--generate-cli-skeleton` Parameter mit jedem API-Aufruf verwenden, um eine Parametervorlage zu generieren und anzuzeigen. Anschließend können Sie die generierte Vorlage anpassen und als Eingabe für einen späteren Befehl verwenden. Einzelheiten finden Sie unter Generieren und Verwenden einer Parameter-Skelettdatei.

# Aktionen

Folgende Aktionen werden unterstützt:

- CreateAccess
- CreateAgreement
- CreateConnector
- CreateProfile
- CreateServer
- CreateUser
- CreateWorkflow
- DeleteAccess
- DeleteAgreement
- DeleteCertificate
- DeleteConnector
- DeleteHostKey
- DeleteProfile
- DeleteServer
- DeleteSshPublicKey

- DeleteUser
- DeleteWorkflow
- DescribeAccess
- DescribeAgreement
- DescribeCertificate
- DescribeConnector
- DescribeExecution
- DescribeHostKey
- DescribeProfile
- DescribeSecurityPolicy
- DescribeServer
- DescribeUser
- DescribeWorkflow
- ImportCertificate
- ImportHostKey
- ImportSshPublicKey
- ListAccesses
- ListAgreements
- ListCertificates
- ListConnectors
- ListExecutions
- ListHostKeys
- ListProfiles
- ListSecurityPolicies
- ListServers
- ListTagsForResource
- ListUsers
- ListWorkflows
- SendWorkflowStepState
- StartDirectoryListing

- StartFileTransfer
- StartServer
- StopServer
- TagResource
- TestConnection
- TestIdentityProvider
- UntagResource
- UpdateAccess
- UpdateAgreement
- UpdateCertificate
- UpdateConnector
- UpdateHostKey
- UpdateProfile
- UpdateServer
- UpdateUser

# CreateAccess

Wird von Administratoren verwendet, um auszuwählen, welche Gruppen im Verzeichnis Zugriff auf das Hoch- und Herunterladen von Dateien über die aktivierten Protokolle haben sollen AWS Transfer Family. Ein Microsoft Active Directory kann beispielsweise 50.000 Benutzer enthalten, aber nur ein kleiner Teil benötigt möglicherweise die Fähigkeit, Dateien auf den Server zu übertragen. Ein Administrator kann `CreateAccess` damit den Zugriff auf die richtige Gruppe von Benutzern beschränken, die diese Fähigkeit benötigen.

## Anforderungssyntax

```
{
   "ExternalId": "string",
   "HomeDirectory": "string",
   "HomeDirectoryMappings": [
      {
         "Entry": "string",
         "Target": "string",
         "Type": "string"
      }
   ],
   "HomeDirectoryType": "string",
   "Policy": "string",
   "PosixProfile": {
      "Gid": number,
      "SecondaryGids": [ number ],
      "Uid": number
   },
   "Role": "string",
   "ServerId": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

### ExternalId

Eine eindeutige Kennung, die zur Identifizierung bestimmter Gruppen in Ihrem Verzeichnis erforderlich ist. Die Benutzer der Gruppe, die Sie zuordnen, haben über die aktivierten Protokolle Zugriff auf Ihre Amazon S3- oder Amazon EFS-Ressourcen AWS Transfer Family. Wenn Sie den Gruppennamen kennen, können Sie die SID-Werte anzeigen, indem Sie den folgenden Befehl unter Windows ausführen PowerShell.

```
Get-ADGroup -Filter {samAccountName -like "YourGroupName*"} -Properties
* | Select SamAccountName,ObjectSid
```

Ersetzen Sie diesen Befehl YourGroupNamedurch den Namen Ihrer Active Directory-Gruppe.

Der reguläre Ausdruck, der zur Überprüfung dieses Parameters verwendet wird, ist eine Zeichenfolge, die aus alphanumerischen Groß- und Kleinbuchstaben ohne Leerzeichen besteht. Sie können auch Unterstriche oder eines der folgenden Zeichen verwenden: =, . @: /-

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 256 Zeichen.

Pattern: `S-1-[\d-]+`

Erforderlich: Ja

### HomeDirectory

Das Zielverzeichnis (Ordner) für einen Benutzer bei der Serveranmeldung mithilfe des Client.

Ein Beispiel für `HomeDirectory` ist `/bucket_name/home/mydirectory`.

> **Note**
>
> Der `HomeDirectory`-Parameter wird nur verwendet, wenn `HomeDirectoryType` auf `PATH` gesetzt ist.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 1024 Zeichen.

Pattern: `(|/.*)`

Erforderlich: Nein

HomeDirectoryMappings

Logische Verzeichniszuordnungen, die angeben, welche Amazon S3- oder Amazon EFS-Pfade und -Schlüssel für Ihren Benutzer sichtbar sein sollen und wie Sie sie sichtbar machen möchten. Sie müssen das `Entry` `Target` Und-Paar angeben, das `Entry` zeigt, wie der Pfad sichtbar gemacht `Target` wird und der tatsächliche Amazon S3- oder Amazon EFS-Pfad ist. Wenn Sie nur ein Ziel angeben, wird es unverändert angezeigt. Sie müssen außerdem sicherstellen, dass Ihre AWS Identity and Access Management (IAM-) Rolle Zugriff auf Pfade in `Target` bietet. Dieser Wert kann nur gesetzt werden, wenn er auf LOGICAL gesetzt `HomeDirectoryType` ist.

Im Folgenden finden Sie ein `Entry` Beispiel für ein `Target` Und-Paar.

```
[ { "Entry": "/directory1", "Target": "/bucket_name/home/
mydirectory" } ]
```

In den meisten Fällen können Sie diesen Wert anstelle der Sitzungsrichtlinie verwenden, um Ihren Benutzer auf das angegebene Home-Verzeichnis (“ `chroot` „) zu sperren. Zu diesem Zweck können Sie den `HomeDirectory` Parameterwert `Entry` `Target` auf `/` und auf diesen festlegen.

Im Folgenden finden Sie ein `Entry` Beispiel für ein `Target` Und-Paar`chroot`.

```
[ { "Entry": "/", "Target": "/bucket_name/home/mydirectory" } ]
```

Typ: Array von HomeDirectoryMapEntry-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Maximale Anzahl von 50000 Artikeln.

Erforderlich: Nein

HomeDirectoryType

Die Art des Zielverzeichnisses (Ordners), das das Home-Verzeichnis Ihrer Benutzer sein soll, wenn sie sich beim Server anmelden. Wenn Sie es auf einstellen`PATH`, sieht der Benutzer den absoluten Amazon S3-Bucket- oder Amazon EFS-Pfad so, wie er in seinen File Transfer Protocol-Clients ist. Wenn Sie es auf einstellen`LOGICAL`, müssen Sie Zuordnungen dafür angeben, wie Sie Amazon S3- oder Amazon EFS-Pfade `HomeDirectoryMappings` für Ihre Benutzer sichtbar machen möchten.

> **Note**
>
> `HomeDirectoryType`Ist dies der `LOGICAL` Fall, müssen Sie mithilfe des Parameters Zuordnungen angeben. `HomeDirectoryMappings` Ist dies hingegen der Fall, `HomeDirectoryType` geben Sie mithilfe des Parameters einen absoluten Pfad an`HomeDirectory`. `PATH` Sie können nicht beides `HomeDirectory` und `HomeDirectoryMappings` in Ihrer Vorlage haben.

Typ: Zeichenfolge

Zulässige Werte: `PATH | LOGICAL`

Erforderlich: Nein

Policy

Eine Sitzungsrichtlinie für Ihren Benutzer, sodass Sie dieselbe AWS Identity and Access Management (IAM-) Rolle für mehrere Benutzer verwenden können. Diese Richtlinie beschränkt den Zugriff eines Benutzers auf Teile seines Amazon S3 S3-Buckets. Variablen, die Sie in dieser Richtlinie verwenden können: `${Transfer:UserName}`, `${Transfer:HomeDirectory}` und `${Transfer:HomeBucket}`.

> **Note**
>
> Diese Richtlinie gilt nur, wenn die Domain von Amazon S3 `ServerId` ist. Amazon EFS verwendet keine Sitzungsrichtlinien.
> AWS Transfer Family Speichert für Sitzungsrichtlinien die Richtlinie als JSON-Blob und nicht als Amazon-Ressourcennamen (ARN) der Richtlinie. Speichern Sie die Richtlinie als ein JSON-Blob und geben Sie sie in das `Policy`-Argument ein.
> Ein Beispiel für eine Sitzungsrichtlinien finden Sie unter Sitzungsrichtlinien.
> Weitere Informationen finden Sie AssumeRolein der AWS Security Token Service API-Referenz.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 2048 Zeichen.

Erforderlich: Nein

### PosixProfile

Die vollständige POSIX-Identität, einschließlich Benutzer-ID (`Uid`), Gruppen-ID (`Gid`) und sekundärer Gruppen-IDs (`SecondaryGids`), die den Zugriff Ihrer Benutzer auf Ihre Amazon EFS-Dateisysteme (Elastic File System) steuert. Die POSIX-Berechtigungen, die für Dateien und Verzeichnisse in Ihrem Dateisystem festgelegt sind, bestimmen die Zugriffsebene, die Ihre Benutzer beim Übertragen von Dateien in und aus Ihren Amazon EFS-Dateisystemen erhalten.

Typ: PosixProfile Objekt

Erforderlich: Nein

### Role

Der Amazon-Ressourcenname (ARN) der Rolle AWS Identity and Access Management (IAM), die den Zugriff Ihrer Benutzer auf Ihren Amazon S3-Bucket oder Ihr Amazon EFS-Dateisystem steuert. Die mit dieser Rolle verbundenen Richtlinien bestimmen die Zugriffsebene, die Sie Ihren Benutzern beim Übertragen von Dateien in und aus Ihrem Amazon-S3-Bucket oder Amazon-EFS-Dateisystem bereitstellen möchten. Die IAM-Rolle sollte außerdem eine Vertrauensstellung enthalten, mit der der Server Zugriff auf Ihre Ressourcen erhält, wenn er die Übertragungsanfragen Ihres Benutzers bearbeitet.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 2048 Zeichen.

Pattern: `arn:.*role/\S+`

Erforderlich: Ja

### ServerId

Eine vom System zugewiesene eindeutige ID für eine Server-Instance. Dies ist der spezifische Server, dem Sie Ihren Benutzer hinzugefügt haben.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Erforderlich: Ja

## Antwortsyntax

```
{
   "ExternalId": "string",
   "ServerId": "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.

ExternalId

Die externe Kennung der Gruppe, deren Benutzer über die aktivierten Protokolle Zugriff auf Ihre Amazon S3- oder Amazon EFS-Ressourcen haben AWS Transfer Family.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 256 Zeichen.

Pattern: `S-1-[\d-]+`

ServerId

Die Kennung des Servers, mit dem der Benutzer verbunden ist.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceExistsException

Die angeforderte Ressource ist nicht vorhanden oder befindet sich in einer anderen Region als der für den Befehl angegebenen.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python

- AWS SDK for Ruby V3

# CreateAgreement

Erstellt eine Vereinbarung. Eine Vereinbarung ist eine bilaterale Handelspartnervereinbarung oder Partnerschaft zwischen einem AWS Transfer Family Server und einem AS2-Prozess. Die Vereinbarung definiert die Beziehung bei der Datei- und Nachrichtenübertragung zwischen dem Server und dem AS2-Prozess. Zum Definieren einer Vereinbarung kombiniert Transfer Family einen Server, ein lokales Profil, ein Partnerprofil, ein Zertifikat und andere Attribute.

Der Partner wird mit der `PartnerProfileId` und der AS2-Prozess wird mit `LocalProfileId` identifiziert.

## Anforderungssyntax

```
{
   "AccessRole": "string",
   "BaseDirectory": "string",
   "Description": "string",
   "LocalProfileId": "string",
   "PartnerProfileId": "string",
   "ServerId": "string",
   "Status": "string",
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ]
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

### AccessRole

Konnektoren werden verwendet, um Dateien entweder über das AS2- oder das SFTP-Protokoll zu senden. Geben Sie für die Zugriffsrolle den Amazon-Ressourcennamen (ARN) der zu AWS Identity and Access Management verwendenden Rolle an.

Für AS2-Konnektoren

Mit AS2 senden Sie Dateien, indem Sie `StartFileTransfer` aufrufen und die Dateipfade im Anforderungsparameter `SendFilePaths` angeben. Mit dem übergeordneten Verzeichnis der Datei (Beispiel: das übergeordnetes Verzeichnis für `--send-file-paths /bucket/dir/file.txt` ist `/bucket/dir/`) speichern wir eine verarbeitete AS2-Nachrichtendatei vorübergehend, speichern die MDN, wenn wir sie vom Partner erhalten, und schreiben eine endgültige JSON-Datei, die relevante Metadaten der Übertragung enthält. Daher muss `AccessRole` Lese- und Schreibzugriff auf das übergeordnete Verzeichnis des in der `StartFileTransfer`-Anforderung verwendeten Dateispeicherorts gewähren. Darüber hinaus müssen Sie Lese- und Schreibzugriff für das übergeordnete Verzeichnis der Dateien gewähren, die Sie mit `StartFileTransfer` senden möchten.

Wenn Sie die Standardauthentifizierung für Ihren AS2-Connector verwenden, erfordert die Zugriffsrolle die `secretsmanager:GetSecretValue` Erlaubnis für den geheimen Schlüssel. Wenn das Geheimnis mit einem vom Kunden verwalteten Schlüssel anstelle des AWS verwalteten Schlüssels in Secrets Manager verschlüsselt wird, benötigt die Rolle auch die `kms:Decrypt` Erlaubnis für diesen Schlüssel.

Für SFTP-Konnektoren

Stellen Sie sicher, dass die Zugriffsrolle Lese- und Schreibzugriff auf das übergeordnete Verzeichnis des Dateispeicherorts bietet, der in der `StartFileTransfer` Anfrage verwendet wird. Stellen Sie außerdem sicher, dass die Rolle die `secretsmanager:GetSecretValue` Berechtigung dazu bietet AWS Secrets Manager.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 2048 Zeichen.

Pattern: `arn:.*role/\S+`

Erforderlich: Ja

## BaseDirectory

Das Zielverzeichnis (Ordner) für Dateien, die mithilfe des AS2-Protokolls übertragen wurden.

Ein Beispiel für `BaseDirectory` ist `/DOC-EXAMPLE-BUCKET/home/mydirectory`.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 1024 Zeichen.

Pattern: `(|/.*)`

Erforderlich: Ja

### Description

Ein Name oder eine kurze Beschreibung zur Identifizierung der Vereinbarung.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 1. Höchstlänge = 200 Zeichen.

Pattern: `[\p{Graph}]+`

Erforderlich: Nein

### LocalProfileId

Eine eindeutige Kennung für das lokale AS2-Profil.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `p-([0-9a-f]{17})`

Erforderlich: Ja

### PartnerProfileId

Eine eindeutige Kennung für das in der Vereinbarung verwendete Partnerprofil.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `p-([0-9a-f]{17})`

Erforderlich: Ja

### ServerId

Eine vom System zugewiesene eindeutige ID für eine Server-Instance. Dies ist der spezifische Server, den die Vereinbarung verwendet.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Erforderlich: Ja

### Status

Der Status der Vereinbarung. Die Vereinbarung kann entweder `ACTIVE` oder sein`INACTIVE`.

Typ: Zeichenfolge

Zulässige Werte: `ACTIVE | INACTIVE`

Erforderlich: Nein

### Tags

Schlüssel-Wert-Paare, die zur Gruppierung und Suche von Vereinbarungen verwendet werden können.

Typ: Array von Tag-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Die maximale Anzahl beträgt 50 Elemente.

Erforderlich: Nein

## Antwortsyntax

```
{
   "AgreementId": "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.

### AgreementId

Die eindeutige Kennung für die Vereinbarung. Verwenden Sie diese ID zum Löschen oder Aktualisieren einer Vereinbarung sowie für alle anderen API-Aufrufe, bei denen Sie die Vereinbarungs-ID angeben müssen.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `a-([0-9a-f]{17})`

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceExistsException

Die angeforderte Ressource ist nicht vorhanden oder befindet sich in einer anderen Region als der für den Befehl angegebenen.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

ThrottlingException

Die Anforderung wurde aufgrund der Drosselung von Anforderungen abgelehnt.

HTTP Status Code: 400

## Beispiele

### Beispiel

Im folgenden Beispiel wird eine Vereinbarung erstellt und die Vereinbarungs-ID zurückgegeben.

```
aws transfer create-agreement --server-id s-021345abcdef6789 --local-profile-
id p-1234567890abcdef0 --partner-profile-id p-abcdef01234567890 --base-folder /DOC-
EXAMPLE-BUCKET/AS2-files --access-role arn:aws:iam::111122223333:role/AS2-role
```

### Beispielantwort

Der API-Aufruf gibt die Vereinbarungs-ID für die neue Vereinbarung zurück.

```
{
    "AgreementId": "a-11112222333344444"
}
```

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# CreateConnector

Erzeugt den Connector, der die Parameter für eine Verbindung für das AS2- oder SFTP-Protokoll erfasst. Für AS2 ist der Connector erforderlich, um Dateien an einen extern gehosteten AS2-Server zu senden. Für SFTP ist der Connector erforderlich, wenn Dateien an einen SFTP-Server gesendet oder Dateien von einem SFTP-Server empfangen werden. Weitere Informationen zu Konnektoren finden Sie unter AS2-Konnektoren konfigurieren und SFTP-Konnektoren erstellen.

> **Note**
>
> Sie müssen genau ein Konfigurationsobjekt angeben: entweder für AS2 (`As2Config`) oder SFTP (). `SftpConfig`

## Anforderungssyntax

```
{
   "AccessRole": "string",
   "As2Config": {
      "BasicAuthSecretId": "string",
      "Compression": "string",
      "EncryptionAlgorithm": "string",
      "LocalProfileId": "string",
      "MdnResponse": "string",
      "MdnSigningAlgorithm": "string",
      "MessageSubject": "string",
      "PartnerProfileId": "string",
      "SigningAlgorithm": "string"
   },
   "LoggingRole": "string",
   "SecurityPolicyName": "string",
   "SftpConfig": {
      "TrustedHostKeys": [ "string" ],
      "UserSecretId": "string"
   },
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ],
```

```
    "Url": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

AccessRole

Konnektoren werden verwendet, um Dateien entweder über das AS2- oder das SFTP-Protokoll zu senden. Geben Sie für die Zugriffsrolle den Amazon-Ressourcennamen (ARN) der zu AWS Identity and Access Management verwendenden Rolle an.

Für AS2-Konnektoren

Mit AS2 senden Sie Dateien, indem Sie `StartFileTransfer` aufrufen und die Dateipfade im Anforderungsparameter `SendFilePaths` angeben. Mit dem übergeordneten Verzeichnis der Datei (Beispiel: das übergeordnetes Verzeichnis für `--send-file-paths /bucket/dir/file.txt` ist `/bucket/dir/`) speichern wir eine verarbeitete AS2-Nachrichtendatei vorübergehend, speichern die MDN, wenn wir sie vom Partner erhalten, und schreiben eine endgültige JSON-Datei, die relevante Metadaten der Übertragung enthält. Daher muss `AccessRole` Lese- und Schreibzugriff auf das übergeordnete Verzeichnis des in der `StartFileTransfer`-Anforderung verwendeten Dateispeicherorts gewähren. Darüber hinaus müssen Sie Lese- und Schreibzugriff für das übergeordnete Verzeichnis der Dateien gewähren, die Sie mit `StartFileTransfer` senden möchten.

Wenn Sie die Standardauthentifizierung für Ihren AS2-Connector verwenden, erfordert die Zugriffsrolle die `secretsmanager:GetSecretValue` Erlaubnis für den geheimen Schlüssel. Wenn das Geheimnis mit einem vom Kunden verwalteten Schlüssel anstelle des AWS verwalteten Schlüssels in Secrets Manager verschlüsselt wird, benötigt die Rolle auch die `kms:Decrypt` Erlaubnis für diesen Schlüssel.

Für SFTP-Konnektoren

Stellen Sie sicher, dass die Zugriffsrolle Lese- und Schreibzugriff auf das übergeordnete Verzeichnis des Dateispeicherorts bietet, der in der `StartFileTransfer` Anfrage verwendet

wird. Stellen Sie außerdem sicher, dass die Rolle die `secretsmanager:GetSecretValue` Berechtigung dazu bietet AWS Secrets Manager.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 2048 Zeichen.

Pattern: `arn:.*role/\S+`

Erforderlich: Ja

As2Config

Eine Struktur, die die Parameter für ein AS2-Connector-Objekt enthält.

Typ: As2ConnectorConfig Objekt

Erforderlich: Nein

LoggingRole

Der Amazon-Ressourcenname (ARN) der Rolle AWS Identity and Access Management (IAM), der es einem Connector ermöglicht, die CloudWatch Protokollierung für Amazon S3-Ereignisse zu aktivieren. Wenn diese Option aktiviert ist, können Sie die Connector-Aktivität in Ihren CloudWatch Protokollen einsehen.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 2048 Zeichen.

Pattern: `arn:.*role/\S+`

Erforderlich: Nein

SecurityPolicyName

Gibt den Namen der Sicherheitsrichtlinie für den Connector an.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 100 Zeichen.

Pattern: `TransferSFTPConnectorSecurityPolicy-[A-Za-z0-9-]+`

Erforderlich: Nein

SftpConfig

Eine Struktur, die die Parameter für ein SFTP-Connector-Objekt enthält.

Typ: SftpConnectorConfig Objekt

Erforderlich: Nein

Tags

Schlüssel-Wert-Paare, die zur Gruppierung und Suche von Connectors verwendet werden. Tags sind Metadaten, die für jeden Zweck an Konnektoren angehängt werden.

Typ: Array von Tag-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Die maximale Anzahl beträgt 50 Elemente.

Erforderlich: Nein

Url

Die URL des AS2- oder SFTP-Endpunkts des Partners.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 255 Zeichen.

Erforderlich: Ja

## Antwortsyntax

```
{
   "ConnectorId": "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.

ConnectorId

Die eindeutige Kennung für den Connector, die nach erfolgreichem API-Aufruf zurückgegeben wird.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `c-([0-9a-f]{17})`

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceExistsException

Die angeforderte Ressource ist nicht vorhanden oder befindet sich in einer anderen Region als der für den Befehl angegebenen.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

ThrottlingException

Die Anforderung wurde aufgrund der Drosselung von Anforderungen abgelehnt.

HTTP Status Code: 400

## Beispiele

### Beispiel

Im folgenden Beispiel wird ein AS2-Connector erstellt. Ersetzen Sie im Befehl die Elemente wie folgt:

- `url`: Geben Sie die URL für den AS2-Server des Handelspartners ein.
- `your-IAM-role-for-bucket-access`: eine IAM-Rolle, die Zugriff auf den Amazon S3 S3-Bucket hat, den Sie zum Speichern Ihrer Dateien verwenden.
- Verwenden Sie den ARN für Ihre Logging-Rolle, der Ihre AWS-Konto ID enthält.
- Geben Sie einen Pfad zu einer Datei an, die die AS2-Connector-Konfigurationsparameter enthält. Das AS2-Connector-Konfigurationsobjekt wird in As2 beschrieben. ConnectorConfig

```
// Listing for testAs2Config.json
{
  "LocalProfileId": "your-profile-id",
  "PartnerProfileId": "partner-profile-id",
  "MdnResponse": "SYNC",
  "Compression": "ZLIB",
  "EncryptionAlgorithm": "AES256_CBC",
  "SigningAlgorithm": "SHA256",
  "MdnSigningAlgorithm": "DEFAULT",
  "MessageSubject": "Your Message Subject"
}
```

```
aws transfer create-connector --url "http://partner-as2-server-url" \
          --access-role your-IAM-role-for-bucket-access \
          --logging-role arn:aws:iam::your-account-id:role/service-role/
AWSTransferLoggingAccess \
```

```
          --as2-config file://path/to/testAS2Config.json
```

Beispiel

Im folgenden Beispiel wird ein SFTP-Connector erstellt. Ersetzen Sie im Befehl die Elemente wie folgt:

- `sftp-server-url`: Geben Sie die URL für den SFTP-Server an, mit dem Sie Dateien austauschen.
- `your-IAM-role-for-bucket-access`: eine IAM-Rolle, die Zugriff auf den Amazon S3 S3-Bucket hat, den Sie zum Speichern Ihrer Dateien verwenden.
- Verwenden Sie den ARN für Ihre Logging-Rolle, der Ihre AWS-Konto ID enthält.
- Geben Sie einen Pfad zu einer Datei an, die die Konfigurationsparameter des SFTP-Connectors enthält. Das Konfigurationsobjekt für den SFTP-Connector wird in SftpConnectorConfig beschrieben.

```
// Listing for testSFTPConfig.json
{
   "UserSecretId": "arn:aws:secretsmanager:us-east-2:123456789012:secret:aws/transfer/
example-username-key",
   "TrustedHostKeys": [
      "sftp.example.com ssh-rsa AAAAbbbb...EEEE="
   ]
}
```

```
aws transfer create-connector --url "sftp://sftp-server-url" \
--access-role your-IAM-role-for-bucket-access \
--logging-role arn:aws:iam::your-account-id:role/service-role/AWSTransferLoggingAccess
 \
--sftp-config file://path/to/testSFTPConfig.json
```

Beispiel

Der API-Aufruf gibt die Connector-ID für den neuen Connector zurück.

Beispielantwort

```
{
```

```
    "ConnectorId": "a-11112222333344444"
}
```

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# CreateProfile

Erstellt das lokale Profil oder Partnerprofil, das für AS2-Übertragungen verwendet werden soll.

## Anforderungssyntax

```
{
   "As2Id": "string",
   "CertificateIds": [ "string" ],
   "ProfileType": "string",
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ]
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

### As2Id

Die `As2Id` ist der AS2-Name, wie in RFC 4130 definiert. Bei eingehenden Übertragungen ist dies der `AS2-From`-Header für die vom Partner gesendeten AS2-Nachrichten. Bei ausgehenden Connectors ist dies der `AS2-To`-Header für die AS2-Nachrichten, die mithilfe der `StartFileTransfer`-API-Operation an den Partner gesendet werden. Diese ID darf keine Leerzeichen enthalten.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 128 Zeichen.

Pattern: `[\p{Print}\s]*`

Erforderlich: Ja

CertificateIds

Ein Array von Kennungen für die importierten Zertifikate. Sie verwenden diese Kennung für die Arbeit mit Profilen und Partnerprofilen.

Typ: Zeichenfolgen-Array

Längenbeschränkungen: Feste Länge von 22.

Pattern: `cert-([0-9a-f]{17})`

Erforderlich: Nein

ProfileType

Bestimmt den Typ des zu erstellenden Profils:

- Geben Sie `LOCAL` an, ob ein lokales Profil erstellt werden soll. Ein lokales Profil stellt die AS2-fähige Transfer Family Family-Serverorganisation oder -partei dar.
- Geben Sie `PARTNER` an, ob Sie ein Partnerprofil erstellen möchten. Ein Partnerprofil steht für eine Remote-Organisation außerhalb von Transfer Family.

Typ: Zeichenfolge

Zulässige Werte: `LOCAL | PARTNER`

Erforderlich: Ja

Tags

Schlüssel-Wert-Paare, die zur Gruppierung und Suche nach AS2-Profilen verwendet werden können.

Typ: Array von Tag-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Die maximale Anzahl beträgt 50 Elemente.

Erforderlich: Nein

## Antwortsyntax

```
{
   "ProfileId": "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.

ProfileId

Die eindeutige Kennung für das AS2-Profil, die nach erfolgreichem API-Aufruf zurückgegeben wird.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `p-([0-9a-f]{17})`

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

ThrottlingException

Die Anforderung wurde aufgrund der Drosselung von Anforderungen abgelehnt.

HTTP Status Code: 400

## Beispiele

### Beispiel

Im folgenden Beispiel wird ein Profil erstellt und die Profil-ID zurückgegeben.

Die Zertifikat-IDs werden bei der Ausführung erstellt`import-certificate`, eine für das Signaturzertifikat und eine für das Verschlüsselungszertifikat.

```
aws transfer create-profile --as2-id MYCORP --certificate-ids c-abcdefg123456hijk
               c-987654aaaa321bbbb
```

### Beispielantwort

Der API-Aufruf gibt die Profil-ID für das neue Profil zurück.

```
{
    "ProfileId": "p-11112222333344444"
}
```

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++

- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# CreateServer

Instanziiert einen virtuellen Server mit automatischer Skalierung basierend auf dem ausgewählten Dateiübertragungs-Protokoll in AWS. Wenn Sie Ihren Dateiübertragungs-Protokoll-fähigen Server aktualisieren oder mit Benutzern arbeiten, verwenden Sie die vom Service generierte `ServerId`-Eigenschaft, die dem neu erstellten Server zugewiesen ist.

## Anforderungssyntax

```
{
   "Certificate": "string",
   "Domain": "string",
   "EndpointDetails": {
      "AddressAllocationIds": [ "string" ],
      "SecurityGroupIds": [ "string" ],
      "SubnetIds": [ "string" ],
      "VpcEndpointId": "string",
      "VpcId": "string"
   },
   "EndpointType": "string",
   "HostKey": "string",
   "IdentityProviderDetails": {
      "DirectoryId": "string",
      "Function": "string",
      "InvocationRole": "string",
      "SftpAuthenticationMethods": "string",
      "Url": "string"
   },
   "IdentityProviderType": "string",
   "LoggingRole": "string",
   "PostAuthenticationLoginBanner": "string",
   "PreAuthenticationLoginBanner": "string",
   "ProtocolDetails": {
      "As2Transports": [ "string" ],
      "PassiveIp": "string",
      "SetStatOption": "string",
      "TlsSessionResumptionMode": "string"
   },
   "Protocols": [ "string" ],
   "S3StorageOptions": {
      "DirectoryListingOptimization": "string"
   },
   "SecurityPolicyName": "string",
```

```
   "StructuredLogDestinations": [ "string" ],
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ],
   "WorkflowDetails": {
      "OnPartialUpload": [
         {
            "ExecutionRole": "string",
            "WorkflowId": "string"
         }
      ],
      "OnUpload": [
         {
            "ExecutionRole": "string",
            "WorkflowId": "string"
         }
      ]
   }
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

Certificate

Der Amazon-Ressourcenname (ARN) des AWS Certificate Manager (ACM)-Zertifikats. Erforderlich, wenn `Protocols` auf FTPS eingestellt ist.

Informationen zum Anfordern eines neuen öffentlichen Zertifikats finden Sie unter Anfordern eines öffentlichen Zertifikats im AWS Certificate Manager Benutzerhandbuch.

Informationen zum Importieren eines vorhandenen Zertifikats in ACM finden Sie unter Zertifikate in ACM importieren im AWS Certificate Manager Benutzerhandbuch.

Informationen zum Anfordern eines privaten Zertifikats für die Verwendung von FTPS über private IP-Adressen finden Sie unter Anfordern eines privaten Zertifikats im AWS Certificate Manager Benutzerhandbuch.

Zertifikate mit den folgenden kryptografischen Algorithmen und Schlüsselgrößen werden unterstützt:

- 2048-Bit-RSA (RSA_2048)
- 4096-Bit-RSA (RSA_4096)
- Elliptic Prime Curve 256-Bit (EC_prime256v1)
- Elliptic Prime Curve 384-Bit (EC_secp384r1)
- Elliptic Prime Curve 521-Bit (EC_secp521r1)

> **Note**
>
> Das Zertifikat muss ein gültiges SSL/TLS X.509 Version 3-Zertifikat mit FQDN oder IP-Adresse und Informationen über den Aussteller sein.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 1600 Zeichen.

Erforderlich: Nein

### Domain

Die Domäne des Speichersystems, das für Dateiübertragungen verwendet wird. Es sind zwei Domänen verfügbar: Amazon Simple Storage Service (Amazon S3) und Amazon Elastic File System (Amazon EFS). Der Standardwert ist S3.

> **Note**
>
> Nachdem der Server erstellt wurde, kann die Domain nicht mehr geändert werden.

Typ: Zeichenfolge

Zulässige Werte: `S3 | EFS`

Erforderlich: Nein

EndpointDetails

Die Virtual Private Cloud (VPC)-Endpunkt-Einstellungen, die für Ihren Server konfiguriert sind. Wenn Sie Ihren Endpunkt in Ihrer VPC hosten, können Sie ihn nur für Ressourcen innerhalb Ihrer VPC zugänglich machen, oder Sie können elastische IP-Adressen anfügen und ihn für Clients über das Internet zugänglich machen. Die Standard-Sicherheitsgruppen Ihrer VPC werden Ihrem Endpunkt automatisch zugewiesen.

Typ: EndpointDetails Objekt

Erforderlich: Nein

EndpointType

Der Typ des Endpunkts, den Ihr Server verwenden soll. Sie können den Endpunkt Ihres Servers öffentlich zugänglich machen (PUBLIC) oder ihn in Ihrer VPC hosten. Mit einem Endpunkt, der in einer VPC gehostet wird, können Sie den Zugriff auf Ihren Server und Ressourcen nur innerhalb Ihrer VPC einschränken oder sich für das Internet entscheiden, indem Sie Elastic-IP-Adressen direkt an ihn anfügen.

> **Note**
>
> Nach dem 19. Mai 2021 können Sie mit `EndpointType=VPC_ENDPOINT` Ihrem keinen Server mehr erstellen, AWS-Konto sofern Ihr Konto dies nicht bereits vor dem 19. Mai 2021 getan hat. Wenn du AWS-Konto am oder vor dem 19. Mai 2021 bereits Server mit `EndpointType=VPC_ENDPOINT` in deinem erstellt hast, bist du davon nicht betroffen. Verwenden Sie nach diesem Datum `EndpointType =VPC`.
> Weitere Informationen finden Sie unter Einstellung der Verwendung von VPC_ENDPOINT.
> Es wird empfohlen, dass Sie `VPC` als `EndpointType` verwenden. Bei diesem Endpunkttyp haben Sie die Möglichkeit, bis zu drei Elastic IPv4-Adressen (inklusive BYO IP) direkt mit dem Endpunkt Ihres Servers zu verknüpfen und VPC-Sicherheitsgruppen zu verwenden, um den Datenverkehr durch die öffentliche IP-Adresse des Clients zu beschränken. Dies ist nicht möglich, wenn `EndpointType` auf `VPC_ENDPOINT` gesetzt ist.

Typ: Zeichenfolge

Zulässige Werte: `PUBLIC | VPC | VPC_ENDPOINT`

Erforderlich: Nein

HostKey

Der private RSA-, ECDSA- oder ED25519-Schlüssel, der für Ihren SFTP-fähigen Server verwendet werden soll. Sie können mehrere Hostschlüssel hinzufügen, falls Sie Schlüssel rotieren möchten oder über einen Satz aktiver Schlüssel verfügen, die unterschiedliche Algorithmen verwenden.

Verwenden Sie den folgenden Befehl, um einen RSA 2048-Bit-Schlüssel ohne Passphrase zu generieren:

```
ssh-keygen -t rsa -b 2048 -N "" -m PEM -f my-new-server-key.
```

Verwenden Sie einen Mindestwert von 2048 für die Option. `-b` Sie können einen stärkeren Schlüssel erstellen, indem Sie 3072 oder 4096 verwenden.

Verwenden Sie den folgenden Befehl, um einen 256-Bit-ECDSA-Schlüssel ohne Passphrase zu generieren:

```
ssh-keygen -t ecdsa -b 256 -N "" -m PEM -f my-new-server-key.
```

Gültige Werte für die `-b` Option für ECDSA sind 256, 384 und 521.

Verwenden Sie den folgenden Befehl, um einen ED25519-Schlüssel ohne Passphrase zu generieren:

```
ssh-keygen -t ed25519 -N "" -f my-new-server-key.
```

Alle diese Befehle können Sie durch eine Zeichenfolge Ihrer Wahl my-new-server-keyersetzen.

> **Important**
>
> Wenn Sie nicht vorhaben, bestehende Benutzer von einem vorhandenen SFTP-fähigen Server auf einen neuen Server zu migrieren, aktualisieren Sie den Hostschlüssel nicht. Das versehentliche Ändern des Host-Schlüssels eines Servers kann zu Unterbrechungen führen.

Weitere Informationen finden Sie im Benutzerhandbuch unter Hostschlüssel für Ihren SFTP-fähigen Server aktualisieren. AWS Transfer Family

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge von 4096.

Erforderlich: Nein

IdentityProviderDetails

Erforderlich, wenn auf `IdentityProviderType` oder gesetzt ist. `AWS_DIRECTORY_SERVICE` `AWS_LAMBDA` `API_GATEWAY` Akzeptiert ein Array mit allen Informationen, die erforderlich sind, um ein Verzeichnis in `AWS_DIRECTORY_SERVICE` zu verwenden oder eine vom Kunden bereitgestellte Authentifizierungs-API aufzurufen, einschließlich der API-Gateway-URL. Erforderlich, wenn `IdentityProviderType` auf `SERVICE_MANAGED` gesetzt ist.

Typ: IdentityProviderDetails Objekt

Erforderlich: Nein

IdentityProviderType

Das Authentifizierungsverfahren für einen Server. Der Standardwert ist`SERVICE_MANAGED`, mit dem Sie Benutzeranmeldeinformationen innerhalb des AWS Transfer Family Dienstes speichern und darauf zugreifen können.

Wird verwendet`AWS_DIRECTORY_SERVICE`, um Zugriff auf Active Directory-Gruppen in AWS Directory Service for Microsoft Active Directory oder Microsoft Active Directory in Ihrer lokalen Umgebung oder bei der AWS Verwendung von AD Connector bereitzustellen. Für diese Option ist es auch erforderlich, dass Sie mithilfe des Parameters `IdentityProviderDetails` eine Directory-ID angeben.

Verwenden Sie den `API_GATEWAY`-Wert für die Integration eines Identitätsanbieters Ihrer Wahl. Die `API_GATEWAY`-Einstellung erfordert, dass Sie mithilfe des Parameters `IdentityProviderDetails` die URL eines Amazon-API-Gateway-Endpunkts angeben, der zur Authentifizierung aufgerufen wird.

Verwenden Sie den `AWS_LAMBDA` Wert, um eine AWS Lambda Funktion direkt als Identitätsanbieter zu verwenden. Wenn Sie diesen Wert wählen, müssen Sie den ARN für die Lambda-Funktion im `Function` Parameter für den `IdentityProviderDetails` Datentyp angeben.

Typ: Zeichenfolge

Zulässige Werte: `SERVICE_MANAGED | API_GATEWAY | AWS_DIRECTORY_SERVICE | AWS_LAMBDA`

Erforderlich: Nein

LoggingRole

Der Amazon-Ressourcenname (ARN) der AWS Identity and Access Management (IAM) -Rolle, der es einem Server ermöglicht, die CloudWatch Amazon-Protokollierung für Amazon S3 oder Amazon EFSEvents zu aktivieren. Wenn diese Option aktiviert ist, können Sie Benutzeraktivitäten in Ihren Protokollen einsehen. CloudWatch

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 2048 Zeichen.

Pattern: `(|arn:.*role/\S+)`

Erforderlich: Nein

PostAuthenticationLoginBanner

Gibt eine Zeichenfolge an, die angezeigt wird, wenn Benutzer sich mit einem Server verbinden. Diese Zeichenfolge wird nach Authentifizierung des Benutzers angezeigt.

> **Note**
>
> Das SFTP-Protokoll unterstützt keine Anzeige-Banner nach der Authentifizierung.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge von 4096.

Pattern: `[\x09-\x0D\x20-\x7E]*`

Erforderlich: Nein

PreAuthenticationLoginBanner

Gibt eine Zeichenfolge an, die angezeigt wird, wenn Benutzer sich mit einem Server verbinden. Diese Zeichenfolge wird angezeigt, bevor sich der Benutzer authentifiziert. Das folgende Banner zeigt beispielsweise Details zur Verwendung des Systems an:

```
This system is for the use of authorized users only. Individuals using
this computer system without authority, or in excess of their authority,
```

```
are subject to having all of their activities on this system monitored
and recorded by system personnel.
```

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge von 4096.

Pattern: `[\x09-\x0D\x20-\x7E]*`

Erforderlich: Nein

## ProtocolDetails

Protokolleinstellungen, die für Ihren Server konfiguriert sind.

- Verwenden Sie den Parameter `PassiveIp` zur Angabe des passiven Modus (für FTP- und FTPS-Protokolle). Geben Sie eine einzelne gepunktete IPv4-Adresse ein, z. B. die externe IP-Adresse einer Firewall, eines Routers oder eines Load Balancers.
- Verwenden Sie den Parameter `SetStatOption`, um den Fehler zu ignorieren, der generiert wird, wenn der Client versucht, den Befehl `SETSTAT` für eine Datei zu verwenden, die Sie in einen Amazon-S3-Bucket hochladen. Damit der AWS Transfer Family Server den `SETSTAT` Befehl ignoriert und Dateien hochlädt, ohne Änderungen an Ihrem SFTP-Client vornehmen zu müssen, setzen Sie den Wert auf. `ENABLE_NO_OP` Wenn Sie den `SetStatOption` Parameter auf setzen`ENABLE_NO_OP`, generiert Transfer Family einen Protokolleintrag in Amazon CloudWatch Logs, sodass Sie feststellen können, wann der Client einen `SETSTAT` Anruf tätigt.
- Verwenden Sie den `TlsSessionResumptionMode` Parameter, um festzustellen, ob Ihr AWS Transfer Family Server die letzten, ausgehandelten Sitzungen über eine eindeutige Sitzungs-ID wieder aufnimmt.
- `As2Transports` gibt an, wie AS2-Nachrichten transportiert werden sollen. Derzeit wird nur HTTP unterstützt.

Typ: ProtocolDetails Objekt

Erforderlich: Nein

## Protocols

Gibt das/die Dateiübertragungsprotokoll(e) an, über das/die der Dateiübertragungsprotokoll-Client eine Verbindung zum Endpunkt des Servers herstellen kann. Die verfügbaren Protokolle sind:

- `SFTP` (Secure Shell (SSH) File Transfer Protocol): Dateiübertragung über SSH
- `FTPS` (File Transfer Protocol Secure): Dateiübertragung mit TLS-Verschlüsselung

- FTP (File Transfer Protocol): Unverschlüsselte Dateiübertragung
- AS2(Anwendbarkeitserklärung 2): Wird für den Transport strukturierter Daten verwendet business-to-business

> **Note**
>
> - Wenn Sie wählenFTPS, müssen Sie ein in AWS Certificate Manager (ACM) gespeichertes Zertifikat wählen, das zur Identifizierung Ihres Servers verwendet wird, wenn Clients über FTPS eine Verbindung zu ihm herstellen.
> - Wenn `Protocol` entweder `FTP` oder `FTPS` enthält, muss `EndpointType` `VPC` lauten und `IdentityProviderType` muss `AWS_DIRECTORY_SERVICE`, `AWS_LAMBDA` oder `API_GATEWAY` lauten.
> - Wenn `Protocol` `FTP` enthält, können `AddressAllocationIds` nicht zugeordnet werden.
> - Wenn `Protocol` nur auf `SFTP` gesetzt ist, kann `EndpointType` auf `PUBLIC` gesetzt werden und `IdentityProviderType` kann auf einen der unterstützten Identitätstypen gesetzt werden: `SERVICE_MANAGED`, `AWS_DIRECTORY_SERVICE`, `AWS_LAMBDA` oder `API_GATEWAY`.
> - Wenn das `Protocol` `AS2` enthält, dann muss der `EndpointType` `VPC` lauten und die Domain muss Amazon S3 sein.

Typ: Zeichenfolgen-Array

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Maximale Anzahl von 4 Elementen.

Zulässige Werte: `SFTP | FTP | FTPS | AS2`

Erforderlich: Nein

S3StorageOptions

Gibt an, ob die Leistung für Ihre Amazon S3 S3-Verzeichnisse optimiert ist oder nicht. Diese ist standardmäßig deaktiviert.

Standardmäßig haben Zuordnungen von Home-Verzeichnissen einen `TYPE` Wert von. `DIRECTORY` Wenn Sie diese Option aktivieren, müssten Sie den Wert dann explizit auf setzen `HomeDirectoryMapEntryType`, `FILE` wenn eine Zuordnung ein Dateiziel haben soll.

Typ: S3StorageOptions Objekt

Erforderlich: Nein

SecurityPolicyName

Gibt den Namen der Sicherheitsrichtlinie für den Server an.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 100 Zeichen.

Pattern: `Transfer[A-Za-z0-9]*SecurityPolicy-[A-Za-z0-9-]+`

Erforderlich: Nein

StructuredLogDestinations

Gibt die Protokollgruppen an, an die Ihre Serverprotokolle gesendet werden.

Um eine Protokollgruppe anzugeben, müssen Sie den ARN für eine bestehende Protokollgruppe angeben. In diesem Fall lautet das Format der Protokollgruppe wie folgt:

`arn:aws:logs:region-name:amazon-account-id:log-group:log-group-name:*`

Beispiel: `arn:aws:logs:us-east-1:111122223333:log-group:mytestgroup:*`

Wenn Sie zuvor eine Protokollgruppe für einen Server angegeben haben, können Sie diese löschen und somit die strukturierte Protokollierung deaktivieren, indem Sie in einem `update-server` Aufruf einen leeren Wert für diesen Parameter angeben. Beispielsweise:

```
update-server --server-id s-1234567890abcdef0 --structured-log-destinations
```

Typ: Zeichenfolge-Array

Array-Mitglieder: Die Mindestanzahl beträgt 0 Elemente. Die maximale Anzahl beträgt 1 Element.

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 1600 Zeichen.

Pattern: `arn:\S+`

Erforderlich: Nein

Tags

Schlüssel-Wert-Paare, die zur Gruppierung und Suche von Servern verwendet werden können.

Typ: Array von Tag-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Die maximale Anzahl beträgt 50 Elemente.

Erforderlich: Nein

WorkflowDetails

Gibt die Workflow-ID für den zuzuweisenden Workflow und die für die Ausführung des Workflows verwendete Ausführungsrolle an.

Zusätzlich zu einem Workflow, der ausgeführt wird, wenn eine Datei vollständig hochgeladen wurde, kann `WorkflowDetails` auch eine Workflow-ID (und Ausführungsrolle) für einen Workflow enthalten, der beim teilweisen Upload ausgeführt werden soll. Ein teilweiser Upload erfolgt, wenn die Serversitzung unterbrochen wird, während die Datei noch hochgeladen wird.

Typ: WorkflowDetails Objekt

Erforderlich: Nein

## Antwortsyntax

```
{
   "ServerId": "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.

ServerId

Die vom Dienst zugewiesene ID des Servers, der erstellt wird.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

AccessDeniedException

Sie haben keinen ausreichenden Zugriff zum Durchführen dieser Aktion.

HTTP Status Code: 400

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceExistsException

Die angeforderte Ressource ist nicht vorhanden oder befindet sich in einer anderen Region als der für den Befehl angegebenen.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

ThrottlingException

Die Anforderung wurde aufgrund der Drosselung von Anforderungen abgelehnt.

HTTP Status Code: 400

## Beispiele

### Beispiel

Im folgenden Beispiel wird ein neuer Server mit einem erstelltVPC_ENDPOINT.

### Beispielanforderung

```
{
   "EndpointType": "VPC",
   "EndpointDetails":...,
   "HostKey": "Your RSA private key",
   "IdentityProviderDetails": "IdentityProvider",
   "IdentityProviderType": "SERVICE_MANAGED",
   "LoggingRole": "CloudWatchLoggingRole",
   "Tags": [
      {
         "Key": "Name",
         "Value": "MyServer"
      }
   ]
}
```

### Beispiel

Dies ist eine Beispielantwort für diesen API-Aufruf.

### Beispielantwort

```
{
   "ServerId": "s-01234567890abcdef"
}
```

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle

- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# CreateUser

Erstellt einen Benutzer und ordnet ihn einem vorhandenen Server mit aktiviertem Dateiübertragungsprotokoll zu. Sie können nur Benutzer mit Servern anlegen und zuordnen, bei denen `IdentityProviderType` auf `SERVICE_MANAGED` gesetzt ist. Mithilfe von Parametern für `CreateUser` können Sie den Benutzernamen angeben, das Basisverzeichnis einrichten, den öffentlichen Schlüssel des Benutzers speichern und dem Benutzer die Rolle AWS Identity and Access Management (IAM) zuweisen. Optional können Sie auch eine Sitzungssrichtlinie hinzufügen und Metadaten Tags zuweisen, die zur Gruppierung und Suche von Benutzern verwendet werden.

## Anforderungssyntax

```
{
   "HomeDirectory": "string",
   "HomeDirectoryMappings": [
      {
         "Entry": "string",
         "Target": "string",
         "Type": "string"
      }
   ],
   "HomeDirectoryType": "string",
   "Policy": "string",
   "PosixProfile": {
      "Gid": number,
      "SecondaryGids": [ number ],
      "Uid": number
   },
   "Role": "string",
   "ServerId": "string",
   "SshPublicKeyBody": "string",
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ],
   "UserName": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

### HomeDirectory

Das Zielverzeichnis (Ordner) für einen Benutzer bei der Serveranmeldung mithilfe des Client.

Ein Beispiel für `HomeDirectory` ist `/bucket_name/home/mydirectory`.

> **Note**
>
> Der `HomeDirectory`-Parameter wird nur verwendet, wenn `HomeDirectoryType` auf `PATH` gesetzt ist.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 1024 Zeichen.

Pattern: `(|/.*)`

Erforderlich: Nein

### HomeDirectoryMappings

Logische Verzeichniszuordnungen, die angeben, welche Amazon S3- oder Amazon EFS-Pfade und -Schlüssel für Ihren Benutzer sichtbar sein sollen und wie Sie sie sichtbar machen möchten. Sie müssen das `Entry` `Target` Und-Paar angeben, das `Entry` zeigt, wie der Pfad sichtbar gemacht `Target` wird und der tatsächliche Amazon S3- oder Amazon EFS-Pfad ist. Wenn Sie nur ein Ziel angeben, wird es unverändert angezeigt. Sie müssen außerdem sicherstellen, dass Ihre AWS Identity and Access Management (IAM-) Rolle Zugriff auf Pfade in `Target` bietet. Dieser Wert kann nur gesetzt werden, wenn er auf LOGICAL gesetzt `HomeDirectoryType` ist.

Das Folgende ist ein `Entry` Beispiel für ein `Target` Und-Paar.

```
[ { "Entry": "/directory1", "Target": "/bucket_name/home/
mydirectory" } ]
```

In den meisten Fällen können Sie diesen Wert anstelle der Sitzungsrichtlinie verwenden, um Ihren Benutzer auf das angegebene Home-Verzeichnis (“ `chroot` „) zu beschränken. Zu diesem Zweck können Sie den Wert `Entry` auf `/` und `Target` auf den Wert setzen, den der Benutzer bei der Anmeldung für sein Home-Verzeichnis sehen soll.

Im Folgenden finden Sie ein `Entry` Beispiel für ein `Target` Und-Paar`chroot`.

```
[ { "Entry": "/", "Target": "/bucket_name/home/mydirectory" } ]
```

Typ: Array von HomeDirectoryMapEntry-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Maximale Anzahl von 50000 Artikeln.

Erforderlich: Nein

HomeDirectoryType

Die Art des Zielverzeichnisses (Ordners), das das Home-Verzeichnis Ihrer Benutzer sein soll, wenn sie sich beim Server anmelden. Wenn Sie es auf einstellen`PATH`, sieht der Benutzer den absoluten Amazon S3-Bucket- oder Amazon EFS-Pfad so, wie er in seinen File Transfer Protocol-Clients ist. Wenn Sie es auf einstellen`LOGICAL`, müssen Sie Zuordnungen dafür angeben, wie Sie Amazon S3- oder Amazon EFS-Pfade `HomeDirectoryMappings` für Ihre Benutzer sichtbar machen möchten.

> **Note**
>
> `HomeDirectoryType`Ist dies der `LOGICAL` Fall, müssen Sie mithilfe des Parameters Zuordnungen angeben. `HomeDirectoryMappings` Ist dies hingegen der Fall, `HomeDirectoryType` geben Sie mithilfe des Parameters einen absoluten Pfad an`HomeDirectory`. `PATH` Sie können nicht beides `HomeDirectory` und `HomeDirectoryMappings` in Ihrer Vorlage haben.

Typ: Zeichenfolge

Zulässige Werte: `PATH | LOGICAL`

Erforderlich: Nein

Policy

Eine Sitzungsrichtlinie für Ihren Benutzer, sodass Sie dieselbe AWS Identity and Access Management (IAM-) Rolle für mehrere Benutzer verwenden können. Diese Richtlinie beschränkt

den Zugriff eines Benutzers auf Teile seines Amazon S3 S3-Buckets. Variablen, die Sie in dieser Richtlinie verwenden können: `${Transfer:UserName}`, `${Transfer:HomeDirectory}` und `${Transfer:HomeBucket}`.

> **Note**
>
> Diese Richtlinie gilt nur, wenn die Domain von Amazon S3 `ServerId` ist. Amazon EFS verwendet keine Sitzungsrichtlinien.
> AWS Transfer Family Speichert für Sitzungsrichtlinien die Richtlinie als JSON-Blob und nicht als Amazon-Ressourcennamen (ARN) der Richtlinie. Speichern Sie die Richtlinie als ein JSON-Blob und geben Sie sie in das `Policy`-Argument ein.
> Ein Beispiel für eine Sitzungsrichtlinien finden Sie unter Sitzungsrichtlinien.
> Weitere Informationen finden Sie AssumeRolein der AWS Security Token Service API-Referenz.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 2048 Zeichen.

Erforderlich: Nein

PosixProfile

Gibt die vollständige POSIX-Identität an, einschließlich Benutzer-ID (`Uid`), Gruppen-ID (`Gid`) und aller sekundären Gruppen-IDs (`SecondaryGids`), die den Zugriff Ihrer Benutzer auf Ihre Amazon EFS-Dateisysteme steuert. Die POSIX-Berechtigungen, die für Dateien und Verzeichnisse in Amazon EFS festgelegt sind, bestimmen die Zugriffsebene, die Ihre Benutzer erhalten, wenn sie Dateien in und aus Ihren Amazon EFS-Dateisystemen übertragen.

Typ: PosixProfile Objekt

Erforderlich: Nein

Role

Der Amazon-Ressourcenname (ARN) der Rolle AWS Identity and Access Management (IAM), die den Zugriff Ihrer Benutzer auf Ihren Amazon S3-Bucket oder Ihr Amazon EFS-Dateisystem steuert. Die mit dieser Rolle verbundenen Richtlinien bestimmen die Zugriffsebene, die Sie Ihren Benutzern beim Übertragen von Dateien in und aus Ihrem Amazon-S3-Bucket oder Amazon-EFS-Dateisystem bereitstellen möchten. Die IAM-Rolle sollte außerdem eine

Vertrauensstellung enthalten, mit der der Server Zugriff auf Ihre Ressourcen erhält, wenn er die Übertragungsanfragen Ihres Benutzers bearbeitet.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 2048 Zeichen.

Pattern: `arn:.*role/\S+`

Erforderlich: Ja

ServerId

Eine vom System zugewiesene eindeutige ID für eine Server-Instance. Dies ist der spezifische Server, dem Sie Ihren Benutzer hinzugefügt haben.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Erforderlich: Ja

SshPublicKeyBody

Der öffentliche Teil des Secure Shell (SSH) -Schlüssels, der zur Authentifizierung des Benutzers gegenüber dem Server verwendet wird.

Bei den drei Standardelementen im SSH-Format für öffentliche Schlüssel handelt es sich `<key type>` um ein optionales `<comment>` Element mit Leerzeichen zwischen den einzelnen Elementen. `<body base64>`

AWS Transfer Family akzeptiert RSA-, ECDSA- und ED25519-Schlüssel.

- Für RSA-Schlüssel ist der Schlüsseltyp. `ssh-rsa`
- Für ED25519-Schlüssel ist der Schlüsseltyp. `ssh-ed25519`
- Bei ECSA-Schlüsseln ist der Schlüsseltyp entweder, oder `ecdsa-sha2-nistp256` `ecdsa-sha2-nistp384ecdsa-sha2-nistp521`, abhängig von der Größe des von Ihnen generierten Schlüssels.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 2048 Zeichen.

Erforderlich: Nein

Tags

Schlüssel-Wert-Paare, die zur Gruppierung und Suche von Benutzern verwendet werden können. Tags sind Benutzern angehängte Metadaten für einen bestimmten Zweck.

Typ: Array von Tag-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Die maximale Anzahl beträgt 50 Elemente.

Erforderlich: Nein

UserName

Eine eindeutige Zeichenfolge, die einen Benutzer identifiziert und `ServerId` zugeordnet ist. Dieser Benutzername muss mindestens 3 und maximal 100 Zeichen enthalten. Folgende Zeichen sind gültig: a-z, A-Z, 0-9, Unterstrich '_', Bindestrich '-', Punkt '.'und beim Zeichen '@'. Der Benutzername kann nicht mit einem Bindestrich, einem Punkt oder einem At beginnen.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 3. Maximale Länge beträgt 100 Zeichen.

Pattern: `[\w][\w@.-]{2,99}`

Erforderlich: Ja

## Antwortsyntax

```
{
   "ServerId": "string",
   "UserName": "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.

ServerId

Die ID des Servers, mit dem der Benutzer verbunden ist.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

UserName

Eine eindeutige Zeichenfolge, die einen Transfer Family Family-Benutzer identifiziert.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 3. Maximale Länge beträgt 100 Zeichen.

Pattern: `[\w][\w@.-]{2,99}`

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceExistsException

Die angeforderte Ressource ist nicht vorhanden oder befindet sich in einer anderen Region als der für den Befehl angegebenen.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Beispiele

Beispiel

Um einen Benutzer zu erstellen, können Sie zunächst die Parameter beispielsweise `createUserParameters` in einer JSON-Datei speichern und dann den API-Befehl create-user ausführen.

```
{
    "HomeDirectory": "/DOC-EXAMPLE-BUCKET",
    "HomeDirectoryType": "PATH",
    "Role": "arn:aws:iam::111122223333:role/bob-role",
    "ServerId": "s-1111aaaa2222bbbb3",
    "SshPublicKeyBody": "ecdsa-sha2-nistp521 AAAAE2VjZHNhLXNoYTItbmlzdHA...
 bobusa@mycomputer.us-east-1.amazon.com",
    "UserName": "bobusa-API"
}
```

Beispielanforderung

```
aws transfer create-user --cli-input-json file://createUserParameters
```

Beispielantwort

```
{
    "ServerId": ""s-1111aaaa2222bbbb3",
    "UserName": "bobusa-API"
}
```

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# CreateWorkflow

Ermöglicht es Ihnen, einen Workflow mit bestimmten Schritten und Schrittdetails zu erstellen, die der Workflow nach Abschluss der Dateiübertragung aufruft. Nachdem Sie einen Workflow erstellt haben, können Sie den erstellten Workflow beliebigen Transferservern zuordnen, indem Sie das Feld `workflow-details` in `CreateServer`- und `UpdateServer`-Vorgängen angeben.

## Anforderungssyntax

```
{
   "Description": "string",
   "OnExceptionSteps": [
      {
         "CopyStepDetails": {
            "DestinationFileLocation": {
               "EfsFileLocation": {
                  "FileSystemId": "string",
                  "Path": "string"
               },
               "S3FileLocation": {
                  "Bucket": "string",
                  "Key": "string"
               }
            },
            "Name": "string",
            "OverwriteExisting": "string",
            "SourceFileLocation": "string"
         },
         "CustomStepDetails": {
            "Name": "string",
            "SourceFileLocation": "string",
            "Target": "string",
            "TimeoutSeconds": number
         },
         "DecryptStepDetails": {
            "DestinationFileLocation": {
               "EfsFileLocation": {
                  "FileSystemId": "string",
                  "Path": "string"
               },
               "S3FileLocation": {
                  "Bucket": "string",
                  "Key": "string"
```

```
                }
            },
            "Name": "string",
            "OverwriteExisting": "string",
            "SourceFileLocation": "string",
            "Type": "string"
        },
        "DeleteStepDetails": {
            "Name": "string",
            "SourceFileLocation": "string"
        },
        "TagStepDetails": {
            "Name": "string",
            "SourceFileLocation": "string",
            "Tags": [
                {
                    "Key": "string",
                    "Value": "string"
                }
            ]
        },
        "Type": "string"
    }
],
"Steps": [
    {
        "CopyStepDetails": {
            "DestinationFileLocation": {
                "EfsFileLocation": {
                    "FileSystemId": "string",
                    "Path": "string"
                },
                "S3FileLocation": {
                    "Bucket": "string",
                    "Key": "string"
                }
            },
            "Name": "string",
            "OverwriteExisting": "string",
            "SourceFileLocation": "string"
        },
        "CustomStepDetails": {
            "Name": "string",
            "SourceFileLocation": "string",
```

```
            "Target": "string",
            "TimeoutSeconds": number
         },
         "DecryptStepDetails": {
            "DestinationFileLocation": {
               "EfsFileLocation": {
                  "FileSystemId": "string",
                  "Path": "string"
               },
               "S3FileLocation": {
                  "Bucket": "string",
                  "Key": "string"
               }
            },
            "Name": "string",
            "OverwriteExisting": "string",
            "SourceFileLocation": "string",
            "Type": "string"
         },
         "DeleteStepDetails": {
            "Name": "string",
            "SourceFileLocation": "string"
         },
         "TagStepDetails": {
            "Name": "string",
            "SourceFileLocation": "string",
            "Tags": [
               {
                  "Key": "string",
                  "Value": "string"
               }
            ]
         },
         "Type": "string"
      }
   ],
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ]
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

### Description

Eine Textbeschreibung für den Workflow.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 256 Zeichen.

Pattern: `[\w- ]*`

Erforderlich: Nein

### OnExceptionSteps

Gibt die Schritte (Aktionen) an, die ausgeführt werden sollen, wenn bei der Ausführung des Workflows Fehler auftreten.

> **Note**
>
> Bei benutzerdefinierten Schritten muss die Lambda-Funktion an die Callback-API sendenFAILURE, um die Ausnahmeschritte auszulösen. Wenn das Lambda SUCCESS vor dem Timeout nicht sendet, werden außerdem die Ausnahmeschritte ausgeführt.

Typ: Array von WorkflowStep-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 0 Elemente. Maximale Anzahl von 8 Elementen.

Erforderlich: Nein

### Steps

Gibt die Details für die Schritte an, die im angegebenen Workflow enthalten sind.

Das TYPE gibt an, welche der folgenden Aktionen für diesen Schritt ergriffen werden.

- **COPY** – Die Datei an einen anderen Ort kopieren.

- **CUSTOM**- Führt einen benutzerdefinierten Schritt mit einem AWS Lambda Funktionsziel aus.
- **DECRYPT** – Eine Datei entschlüsseln, die vor dem Hochladen verschlüsselt wurde.
- **DELETE** – Die Datei löschen.
- **TAG** – Der Datei ein Tag hinzufügen.

> **Note**
>
> Derzeit werden Kopieren und Taggen nur auf S3 unterstützt.

Als Dateispeicherort geben Sie entweder den Amazon S3 S3-Bucket und -Schlüssel oder die Amazon EFS-Dateisystem-ID und den Pfad an.

Typ: Array von WorkflowStep-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 0 Elemente. Maximale Anzahl von 8 Artikeln.

Erforderlich: Ja

Tags

Schlüssel-Wert-Paare, die zum Gruppieren und Suchen nach Workflows verwendet werden können. Tags sind Metadaten, die zu beliebigen Zwecken an Workflows angefügt werden.

Typ: Array von Tag-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Die maximale Anzahl beträgt 50 Elemente.

Erforderlich: Nein

## Antwortsyntax

```
{
   "WorkflowId": "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.

WorkflowId

Eine eindeutige Kennung für den Workflow.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `w-([a-z0-9]{17})`

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

AccessDeniedException

Sie haben keinen ausreichenden Zugriff zum Durchführen dieser Aktion.

HTTP Status Code: 400

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceExistsException

Die angeforderte Ressource ist nicht vorhanden oder befindet sich in einer anderen Region als der für den Befehl angegebenen.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, weil der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

ThrottlingException

Die Anforderung wurde aufgrund der Drosselung von Anforderungen abgelehnt.

HTTP Status Code: 400

## Beispiele

### Beispiel

Sie können Workflow-Schrittinformationen in einer Textdatei speichern und diese Datei dann verwenden, um einen Workflow zu erstellen, wie im folgenden Beispiel gezeigt. Das folgende Beispiel geht davon aus, dass Sie Ihre Workflow-Schritte in *example-file.json* gespeichert haben (in demselben Ordner, in dem Sie den Befehl ausführen) und dass Sie den Workflow in der Region Nord-Virginia (us-east-1) erstellen möchten.

```
aws transfer create-workflow --description "example workflow from a file" --steps
 file://example-file.json --region us-east-1
```

```
// Example file containing workflow steps
[
   {
     "Type": "TAG",
     "TagStepDetails": {
        "Name": "TagStep",
        "Tags": [
          {
            "Key": "name",
            "Value": "testTag"
          }
        ]
     }
   },
   {
     "Type": "COPY",
     "CopyStepDetails": {
        "Name": "CopyStep",
        "DestinationFileLocation": {
        "S3FileLocation": {
           "Bucket": "DOC-EXAMPLE-BUCKET",
           "Key": "DOC-EXAMPLE-KEY/"
```

```
                    }
            },
            "OverwriteExisting": "TRUE",
            "SourceFileLocation": "${original.file}"
        }
    },
    {
        "Type": "DELETE",
        "DeleteStepDetails":{
        "Name":"DeleteStep",
        "SourceFileLocation": "${original.file}"
        }
    }
]
```

Beispiel

Der `CreateWorkflow` Aufruf gibt die Workflow-ID für den neuen Workflow zurück.

Beispielantwort

```
{
    "WorkflowId": "w-1234abcd5678efghi"
}
```

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# DeleteAccess

Ermöglicht das Löschen des in den `ExternalID` Parametern `ServerID` und angegebenen Zugriffs.

## Anforderungssyntax

```
{
   "ExternalId": "string",
   "ServerId": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

ExternalId

Eine eindeutige Kennung, die zur Identifizierung bestimmter Gruppen in Ihrem Verzeichnis erforderlich ist. Die Benutzer der Gruppe, die Sie zuordnen, haben über die aktivierten Protokolle Zugriff auf Ihre Amazon S3- oder Amazon EFS-Ressourcen AWS Transfer Family. Wenn Sie den Gruppennamen kennen, können Sie die SID-Werte anzeigen, indem Sie den folgenden Befehl unter Windows ausführen PowerShell.

```
Get-ADGroup -Filter {samAccountName -like "YourGroupName*"} -Properties
* | Select SamAccountName,ObjectSid
```

Ersetzen Sie diesen Befehl YourGroupNamedurch den Namen Ihrer Active Directory-Gruppe.

Der reguläre Ausdruck, der zur Überprüfung dieses Parameters verwendet wird, ist eine Zeichenfolge, die aus alphanumerischen Groß- und Kleinbuchstaben ohne Leerzeichen besteht. Sie können auch Unterstriche oder eines der folgenden Zeichen verwenden: =, . @: /-

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 256 Zeichen.

Pattern: `S-1-[\d-]+`

Erforderlich: Ja

ServerId

Ein vom System zugewiesener eindeutiger Bezeichner für einen Server, dem dieser Benutzer zugewiesen wurde.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Erforderlich: Ja

## Antwortelemente

Wenn die Aktion erfolgreich ist, gibt der Dienst eine HTTP 200-Antwort mit leerem HTTP-Textinhalt zurück.

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# DeleteAgreement

Löschen Sie die Vereinbarung, die in der bereitgestellten Vereinbarung angegeben ist`AgreementId`.

## Anforderungssyntax

```
{
   "AgreementId": "string",
   "ServerId": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

AgreementId

Eine eindeutige Kennung für die Vereinbarung. Diese Kennung wird zurückgegeben, wenn Sie eine Vereinbarung erstellen.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `a-([0-9a-f]{17})`

Erforderlich: Ja

ServerId

Die Server-ID, die mit der Vereinbarung verknüpft ist, die Sie löschen.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Erforderlich: Ja

## Antwortelemente

Wenn die Aktion erfolgreich ist, gibt der Dienst eine HTTP 200-Antwort mit leerem HTTP-Textinhalt zurück.

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++

- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# DeleteCertificate

Löscht das Zertifikat, das im `CertificateId` Parameter angegeben ist.

## Anforderungssyntax

```
{
   "CertificateId": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

CertificateId

Die ID des Zertifikatsobjekts, das Sie löschen.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 22.

Pattern: `cert-([0-9a-f]{17})`

Erforderlich: Ja

## Antwortelemente

Wenn die Aktion erfolgreich ist, gibt der Dienst eine HTTP 200-Antwort mit leerem HTTP-Textinhalt zurück.

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# DeleteConnector

Löscht den Connector, der in der bereitgestellten `ConnectorId` Datei angegeben ist.

## Anforderungssyntax

```
{
   "ConnectorId": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

ConnectorId

Der eindeutige Bezeichner für den Konnektor.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `c-([0-9a-f]{17})`

Erforderlich: Ja

## Antwortelemente

Wenn die Aktion erfolgreich ist, gibt der Dienst eine HTTP 200-Antwort mit leerem HTTP-Textinhalt zurück.

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# DeleteHostKey

Löscht den Hostschlüssel, der im `HostKeyId` Parameter angegeben ist.

## Anforderungssyntax

```
{
   "HostKeyId": "string",
   "ServerId": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

HostKeyId

Der Bezeichner des Hostschlüssels, den Sie löschen.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 25.

Pattern: `hostkey-[0-9a-f]{17}`

Erforderlich: Ja

ServerId

Die ID des Servers, der den Hostschlüssel enthält, den Sie löschen.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Erforderlich: Ja

## Antwortelemente

Wenn die Aktion erfolgreich ist, gibt der Dienst eine HTTP 200-Antwort mit leerem HTTP-Textinhalt zurück.

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

ThrottlingException

Die Anforderung wurde aufgrund der Drosselung von Anforderungen abgelehnt.

HTTP Status Code: 400

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# DeleteProfile

Löscht das Profil, das im `ProfileId` Parameter angegeben ist.

## Anforderungssyntax

```
{
   "ProfileId": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

ProfileId

Die ID des Profils, das Sie löschen.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `p-([0-9a-f]{17})`

Erforderlich: Ja

## Antwortelemente

Wenn die Aktion erfolgreich ist, gibt der Dienst eine HTTP 200-Antwort mit leerem HTTP-Textinhalt zurück.

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# DeleteServer

Löscht den von Ihnen angegebenen Server mit aktiviertem Dateiübertragungsprotokoll.

Von diesem Vorgang wird keine Antwort zurückgegeben.

## Anforderungssyntax

```
{
   "ServerId": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

ServerId

Ein eindeutiger, vom System zugewiesener Bezeichner für eine Serverinstanz.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Erforderlich: Ja

## Antwortelemente

Wenn die Aktion erfolgreich ist, gibt der Dienst eine HTTP 200-Antwort mit leerem HTTP-Textinhalt zurück.

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

AccessDeniedException

Sie haben keinen ausreichenden Zugriff zum Durchführen dieser Aktion.

HTTP Status Code: 400

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Beispiele

### Beispiel

Im folgenden Beispiel wird ein Server gelöscht.

Beispielanforderung

```
{
   "ServerId": "s-01234567890abcdef"
}
```

### Beispiel

Bei Erfolg wird nichts zurückgegeben.

### Beispielantwort

```
{
}
```

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# DeleteSshPublicKey

Löscht den öffentlichen Secure Shell (SSH) -Schlüssel eines Benutzers.

## Anforderungssyntax

```
{
   "ServerId": "string",
   "SshPublicKeyId": "string",
   "UserName": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

ServerId

Ein vom System zugewiesener eindeutiger Bezeichner für eine Serverinstanz mit aktiviertem File Transfer Protocol, der der Benutzer zugewiesen wurde.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Erforderlich: Ja

SshPublicKeyId

Eine eindeutige Kennung, die verwendet wird, um auf den spezifischen SSH-Schlüssel Ihres Benutzers zu verweisen.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 21.

Pattern: `key-[0-9a-f]{17}`

Erforderlich: Ja

UserName

Eine eindeutige Zeichenfolge, die einen Benutzer identifiziert, dessen öffentlicher Schlüssel gelöscht wird.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 3. Maximale Länge beträgt 100 Zeichen.

Pattern: `[\w][\w@.-]{2,99}`

Erforderlich: Ja

## Antwortelemente

Wenn die Aktion erfolgreich ist, gibt der Dienst eine HTTP 200-Antwort mit leerem HTTP-Textinhalt zurück.

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

ThrottlingException

Die Anforderung wurde aufgrund der Drosselung von Anforderungen abgelehnt.

HTTP Status Code: 400

## Beispiele

Beispiel

Im folgenden Beispiel wird der öffentliche SSH-Schlüssel eines Benutzers gelöscht.

Beispielanforderung

```
{
   "ServerId": "s-01234567890abcdef",
   "SshPublicKeyId": "MyPublicKey",
   "UserName": "my_user"
}
```

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python

- AWS SDK for Ruby V3

# DeleteUser

Löscht den Benutzer, der zu einem von Ihnen angegebenen Server gehört, für den das Dateiübertragungsprotokoll aktiviert ist.

Von diesem Vorgang wird keine Antwort zurückgegeben.

> **Note**
>
> Wenn Sie einen Benutzer von einem Server löschen, gehen die Benutzerinformationen verloren.

## Anforderungssyntax

```
{
   "ServerId": "string",
   "UserName": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

**ServerId**

Ein vom System zugewiesener eindeutiger Bezeichner für eine Serverinstanz, der der Benutzer zugewiesen wurde.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Erforderlich: Ja

**UserName**

Eine eindeutige Zeichenfolge, die einen Benutzer identifiziert, der von einem Server gelöscht wird.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 3. Maximale Länge beträgt 100 Zeichen.

Pattern: `[\w][\w@.-]{2,99}`

Erforderlich: Ja

## Antwortelemente

Wenn die Aktion erfolgreich ist, gibt der Dienst eine HTTP 200-Antwort mit leerem HTTP-Textinhalt zurück.

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Beispiele

### Beispiel

Im folgenden Beispiel wird ein Transfer Family Family-Benutzer gelöscht.

### Beispielanforderung

```
{
   "ServerId": "s-01234567890abcdef",
   "UserNames": "my_user"
}
```

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# DeleteWorkflow

Löscht den angegebenen Workflow.

## Anforderungssyntax

```
{
   "WorkflowId": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

WorkflowId

Eine eindeutige Kennung für den Workflow.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `w-([a-z0-9]{17})`

Erforderlich: Ja

## Antwortelemente

Wenn die Aktion erfolgreich ist, gibt der Dienst eine HTTP 200-Antwort mit leerem HTTP-Textinhalt zurück.

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

AccessDeniedException

Sie haben keinen ausreichenden Zugriff zum Durchführen dieser Aktion.

HTTP Status Code: 400

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# DescribeAccess

Beschreibt den Zugriff, der dem spezifischen Server mit aktiviertem Dateiübertragungsprotokoll zugewiesen ist, wie er durch seine `ServerId` Eigenschaft und seine identifiziert wird. `ExternalId`

Die Antwort auf diesen Aufruf gibt die Eigenschaften des Zugriffs zurück, der dem angegebenen `ServerId` Wert zugeordnet ist.

## Anforderungssyntax

```
{
   "ExternalId": "string",
   "ServerId": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

ExternalId

Eine eindeutige Kennung, die zur Identifizierung bestimmter Gruppen in Ihrem Verzeichnis erforderlich ist. Die Benutzer der Gruppe, die Sie zuordnen, haben über die aktivierten Protokolle Zugriff auf Ihre Amazon S3- oder Amazon EFS-Ressourcen AWS Transfer Family. Wenn Sie den Gruppennamen kennen, können Sie die SID-Werte anzeigen, indem Sie den folgenden Befehl unter Windows ausführen PowerShell.

```
Get-ADGroup -Filter {samAccountName -like "YourGroupName*"} -Properties
* | Select SamAccountName,ObjectSid
```

Ersetzen Sie diesen Befehl YourGroupNamedurch den Namen Ihrer Active Directory-Gruppe.

Der reguläre Ausdruck, der zur Überprüfung dieses Parameters verwendet wird, ist eine Zeichenfolge, die aus alphanumerischen Groß- und Kleinbuchstaben ohne Leerzeichen besteht. Sie können auch Unterstriche oder eines der folgenden Zeichen verwenden: =, . @: /-

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 256 Zeichen.

Pattern: `S-1-[\d-]+`

Erforderlich: Ja

ServerId

Ein vom System zugewiesener eindeutiger Bezeichner für einen Server, dem dieser Zugriff zugewiesen wurde.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Erforderlich: Ja

## Antwortsyntax

```
{
   "Access": {
      "ExternalId": "string",
      "HomeDirectory": "string",
      "HomeDirectoryMappings": [
         {
            "Entry": "string",
            "Target": "string",
            "Type": "string"
         }
      ],
      "HomeDirectoryType": "string",
      "Policy": "string",
      "PosixProfile": {
         "Gid": number,
         "SecondaryGids": [ number ],
         "Uid": number
      },
      "Role": "string"
   },
```

```
   "ServerId": "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.

Access

Die externe Kennung des Servers, mit dem der Zugriff verbunden ist.

Typ: DescribedAccess Objekt

ServerId

Eine vom System zugewiesene eindeutige Kennung für einen Server, dem dieser Zugriff zugewiesen wurde.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# DescribeAgreement

Beschreibt die Vereinbarung, die durch den identifiziert wird`AgreementId`.

## Anforderungssyntax

```
{
   "AgreementId": "string",
   "ServerId": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

AgreementId

Eine eindeutige Kennung für die Vereinbarung. Diese Kennung wird zurückgegeben, wenn Sie eine Vereinbarung erstellen.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `a-([0-9a-f]{17})`

Erforderlich: Ja

ServerId

Die Server-ID, die der Vereinbarung zugeordnet ist.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Erforderlich: Ja

## Antwortsyntax

```
{
   "Agreement": {
      "AccessRole": "string",
      "AgreementId": "string",
      "Arn": "string",
      "BaseDirectory": "string",
      "Description": "string",
      "LocalProfileId": "string",
      "PartnerProfileId": "string",
      "ServerId": "string",
      "Status": "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.

Agreement

Die Details für die angegebene Vereinbarung, die als `DescribedAgreement` Objekt zurückgegeben wurden.

Typ: DescribedAgreement Objekt

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# DescribeCertificate

Beschreibt das Zertifikat, das durch den identifiziert wird`CertificateId`.

## Anforderungssyntax

```
{
   "CertificateId": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

### CertificateId

Ein Array von Kennungen für die importierten Zertifikate. Sie verwenden diese Kennung für die Arbeit mit Profilen und Partnerprofilen.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 22.

Pattern: `cert-([0-9a-f]{17})`

Erforderlich: Ja

## Antwortsyntax

```
{
   "Certificate": {
      "ActiveDate": number,
      "Arn": "string",
      "Certificate": "string",
      "CertificateChain": "string",
      "CertificateId": "string",
      "Description": "string",
      "InactiveDate": number,
```

```
      "NotAfterDate": number,
      "NotBeforeDate": number,
      "Serial": "string",
      "Status": "string",
      "Tags": [
         {
            "Key": "string",
            "Value": "string"
         }
      ],
      "Type": "string",
      "Usage": "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.

Certificate

Die Details für das angegebene Zertifikat, die als Objekt zurückgegeben wurden.

Typ: DescribedCertificate Objekt

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# DescribeConnector

Beschreibt den Konnektor, der identifiziert wird durch `ConnectorId`.

## Anforderungssyntax

```
{
   "ConnectorId": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

### ConnectorId

Der eindeutige Bezeichner für den Konnektor.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `c-([0-9a-f]{17})`

Erforderlich: Ja

## Antwortsyntax

```
{
   "Connector": {
      "AccessRole": "string",
      "Arn": "string",
      "As2Config": {
         "BasicAuthSecretId": "string",
         "Compression": "string",
         "EncryptionAlgorithm": "string",
         "LocalProfileId": "string",
         "MdnResponse": "string",
         "MdnSigningAlgorithm": "string",
```

```
         "MessageSubject": "string",
         "PartnerProfileId": "string",
         "SigningAlgorithm": "string"
      },
      "ConnectorId": "string",
      "LoggingRole": "string",
      "SecurityPolicyName": "string",
      "ServiceManagedEgressIpAddresses": [ "string" ],
      "SftpConfig": {
         "TrustedHostKeys": [ "string" ],
         "UserSecretId": "string"
      },
      "Tags": [
         {
            "Key": "string",
            "Value": "string"
         }
      ],
      "Url": "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.

Connector

Die Struktur, die die Details des Konnektors enthält.

Typ: DescribedConnector Objekt

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# DescribeExecution

Sie können verwenden`DescribeExecution`, um die Details der Ausführung des angegebenen Workflows zu überprüfen.

> **Note**
>
> Dieser API-Aufruf gibt nur Details für Workflows zurück, die gerade bearbeitet werden. Wenn Sie eine ID für eine Ausführung angeben, die nicht in Bearbeitung ist, oder wenn die Ausführung nicht mit der angegebenen Workflow-ID übereinstimmt, erhalten Sie eine `ResourceNotFound` Ausnahme.

## Anforderungssyntax

```
{
   "ExecutionId": "string",
   "WorkflowId": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

ExecutionId

Eine eindeutige Kennung für die Ausführung eines Workflows.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 36.

Pattern: `[0-9a-fA-F]{8}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{12}`

Erforderlich: Ja

### WorkflowId

Eine eindeutige Kennung für den Workflow.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `w-([a-z0-9]{17})`

Erforderlich: Ja

## Antwortsyntax

```
{
   "Execution": {
      "ExecutionId": "string",
      "ExecutionRole": "string",
      "InitialFileLocation": {
         "EfsFileLocation": {
            "FileSystemId": "string",
            "Path": "string"
         },
         "S3FileLocation": {
            "Bucket": "string",
            "Etag": "string",
            "Key": "string",
            "VersionId": "string"
         }
      },
      "LoggingConfiguration": {
         "LoggingRole": "string",
         "LogGroupName": "string"
      },
      "PosixProfile": {
         "Gid": number,
         "SecondaryGids": [ number ],
         "Uid": number
      },
      "Results": {
         "OnExceptionSteps": [
            {
               "Error": {
```

```
                  "Message": "string",
                  "Type": "string"
               },
               "Outputs": "string",
               "StepType": "string"
            }
         ],
         "Steps": [
            {
               "Error": {
                  "Message": "string",
                  "Type": "string"
               },
               "Outputs": "string",
               "StepType": "string"
            }
         ]
      },
      "ServiceMetadata": {
         "UserDetails": {
            "ServerId": "string",
            "SessionId": "string",
            "UserName": "string"
         }
      },
      "Status": "string"
   },
   "WorkflowId": "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.

### Execution

Die Struktur, die die Details der Workflow-Ausführung enthält.

Typ: DescribedExecution Objekt

### WorkflowId

Eine eindeutige Kennung für den Workflow.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `w-([a-z0-9]{17})`

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET

- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# DescribeHostKey

Gibt die Details des Host-Schlüssels zurück, der durch `HostKeyId` und angegeben ist`ServerId`.

## Anforderungssyntax

```
{
   "HostKeyId": "string",
   "ServerId": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

HostKeyId

Der Bezeichner des Hostschlüssels, den Sie beschreiben möchten.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 25.

Pattern: `hostkey-[0-9a-f]{17}`

Erforderlich: Ja

ServerId

Die ID des Servers, der den Hostschlüssel enthält, den Sie beschreiben möchten.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Erforderlich: Ja

## Antwortsyntax

```
{
   "HostKey": {
      "Arn": "string",
      "DateImported": number,
      "Description": "string",
      "HostKeyFingerprint": "string",
      "HostKeyId": "string",
      "Tags": [
         {
            "Key": "string",
            "Value": "string"
         }
      ],
      "Type": "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.

HostKey

Gibt die Details für den angegebenen Hostschlüssel zurück.

Typ: DescribedHostKey Objekt

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# DescribeProfile

Gibt die Details des Profils zurück, das durch den angegeben ist`ProfileId`.

## Anforderungssyntax

```
{
   "ProfileId": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

ProfileId

Die ID des Profils, das Sie beschreiben möchten.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `p-([0-9a-f]{17})`

Erforderlich: Ja

## Antwortsyntax

```
{
   "Profile": {
      "Arn": "string",
      "As2Id": "string",
      "CertificateIds": [ "string" ],
      "ProfileId": "string",
      "ProfileType": "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.

Profile

Die Details des angegebenen Profils, die als Objekt zurückgegeben werden.

Typ: DescribedProfile Objekt

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# DescribeSecurityPolicy

Beschreibt die Sicherheitsrichtlinie, die an Ihren Server oder SFTP-Connector angehängt ist. Die Antwort enthält eine Beschreibung der Eigenschaften der Sicherheitsrichtlinie. Weitere Informationen zu Sicherheitsrichtlinien finden Sie unter Arbeiten mit Sicherheitsrichtlinien für Server oder Arbeiten mit Sicherheitsrichtlinien für SFTP-Connectors.

## Anforderungssyntax

```
{
   "SecurityPolicyName": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

SecurityPolicyName

Geben Sie den Textnamen der Sicherheitsrichtlinie an, für die Sie die Details abrufen möchten.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 100 Zeichen.

Pattern: `Transfer[A-Za-z0-9]*SecurityPolicy-[A-Za-z0-9-]+`

Erforderlich: Ja

## Antwortsyntax

```
{
   "SecurityPolicy": {
      "Fips": boolean,
      "Protocols": [ "string" ],
      "SecurityPolicyName": "string",
      "SshCiphers": [ "string" ],
```

```
        "SshHostKeyAlgorithms": [ "string" ],
        "SshKexs": [ "string" ],
        "SshMacs": [ "string" ],
        "TlsCiphers": [ "string" ],
        "Type": "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.

SecurityPolicy

Ein Array, das die Eigenschaften der Sicherheitsrichtlinie enthält.

Typ: DescribedSecurityPolicy Objekt

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Beispiele

### Beispiel

Der folgende Beispielbefehl verwendet den Namen der Sicherheitsrichtlinie als Argument und gibt die Algorithmen für die angegebene Sicherheitsrichtlinie zurück.

Beispielanforderung

```
aws transfer describe-security-policy --security-policy-name "TransferSecurityPolicy-
FIPS-2023-05"
```

Beispielantwort

```
{
    "SecurityPolicy": {
        "Fips": true,
        "SecurityPolicyName": "TransferSecurityPolicy-FIPS-2023-05",
        "SshCiphers": [
            "aes256-gcm@openssh.com",
            "aes128-gcm@openssh.com",
            "aes256-ctr",
            "aes192-ctr"
        ],
        "SshKexs": [
            "diffie-hellman-group16-sha512",
            "diffie-hellman-group18-sha512",
            "diffie-hellman-group-exchange-sha256"
        ],
        "SshMacs": [
            "hmac-sha2-256-etm@openssh.com",
            "hmac-sha2-512-etm@openssh.com"
        ],
        "TlsCiphers": [
            "TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256",
            "TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256",
```

```
            "TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256",
            "TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256",
            "TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384",
            "TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384",
            "TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384",
            "TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384"
        ]
    }
}
```

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# DescribeServer

Beschreibt einen Server, für den das Dateiübertragungsprotokoll aktiviert ist, den Sie angeben, indem Sie den Parameter übergeben. `ServerId`

Die Antwort enthält eine Beschreibung der Eigenschaften eines Servers. Wenn Sie `EndpointType` auf VPC setzen, enthält die Antwort die`EndpointDetails`.

## Anforderungssyntax

```
{
   "ServerId": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

ServerId

Ein vom System zugewiesener eindeutiger Bezeichner für einen Server.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Erforderlich: Ja

## Antwortsyntax

```
{
   "Server": {
      "Arn": "string",
      "As2ServiceManagedEgressIpAddresses": [ "string" ],
      "Certificate": "string",
      "Domain": "string",
      "EndpointDetails": {
```

```
      "AddressAllocationIds": [ "string" ],
      "SecurityGroupIds": [ "string" ],
      "SubnetIds": [ "string" ],
      "VpcEndpointId": "string",
      "VpcId": "string"
   },
   "EndpointType": "string",
   "HostKeyFingerprint": "string",
   "IdentityProviderDetails": {
      "DirectoryId": "string",
      "Function": "string",
      "InvocationRole": "string",
      "SftpAuthenticationMethods": "string",
      "Url": "string"
   },
   "IdentityProviderType": "string",
   "LoggingRole": "string",
   "PostAuthenticationLoginBanner": "string",
   "PreAuthenticationLoginBanner": "string",
   "ProtocolDetails": {
      "As2Transports": [ "string" ],
      "PassiveIp": "string",
      "SetStatOption": "string",
      "TlsSessionResumptionMode": "string"
   },
   "Protocols": [ "string" ],
   "S3StorageOptions": {
      "DirectoryListingOptimization": "string"
   },
   "SecurityPolicyName": "string",
   "ServerId": "string",
   "State": "string",
   "StructuredLogDestinations": [ "string" ],
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ],
   "UserCount": number,
   "WorkflowDetails": {
      "OnPartialUpload": [
         {
            "ExecutionRole": "string",
```

```
                "WorkflowId": "string"
            }
         ],
         "OnUpload": [
            {
               "ExecutionRole": "string",
               "WorkflowId": "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.

Server

Ein Array, das die Eigenschaften eines Servers mit den von `ServerID` Ihnen angegebenen Eigenschaften enthält.

Typ: DescribedServer Objekt

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Beispiele

Beispiel

Im folgenden Beispiel werden die einem Server zugewiesenen Eigenschaften zurückgegeben.

Beispielanforderung

```
{
   "ServerId": "s-01234567890abcdef"
}
```

Beispiel

Dieses Beispiel veranschaulicht eine Verwendung von DescribeServer.

Beispielantwort

```
{
    "Server": {
        "Arn": "arn:aws:transfer:us-east-1:176354371281:server/s-01234567890abcdef",
        "EndpointDetails": {
            "AddressAllocationIds": [
                "eipalloc-01a2eabe3c04d5678",
                "eipalloc-102345be"
            ],
            "SubnetIds": [
                "subnet-047eaa7f0187a7cde",
```

```
                "subnet-0a2d0f474daffde18"
            ],
            "VpcEndpointId": "vpce-03fe0080e7cb008b8",
            "VpcId": "vpc-09047a51f1c8e1634"
        },
        "EndpointType": "VPC",
        "HostKeyFingerprint": "your host key,
        "IdentityProviderType": "SERVICE_MANAGED",
        "ServerId": "s-01234567890abcdef",
        "State": "ONLINE",
        "Tags": [],
        "UserCount": 0
    }
}
```

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# DescribeUser

Beschreibt den Benutzer, der dem spezifischen Server mit aktiviertem Dateiübertragungsprotokoll zugewiesen ist, wie durch seine Eigenschaft identifiziert. `ServerId`

Die Antwort auf diesen Aufruf gibt die Eigenschaften des Benutzers zurück, die dem angegebenen `ServerId` Wert zugeordnet sind.

## Anforderungssyntax

```
{
   "ServerId": "string",
   "UserName": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

ServerId

Ein vom System zugewiesener eindeutiger Bezeichner für einen Server, dem dieser Benutzer zugewiesen wurde.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Erforderlich: Ja

UserName

Der Name des Benutzers, der einem oder mehreren Servern zugewiesen ist. Benutzernamen sind Teil der Anmeldeinformationen für die Nutzung des AWS Transfer Family Dienstes und die Ausführung von Dateiübertragungsaufgaben.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 3. Maximale Länge beträgt 100 Zeichen.

Pattern: `[\w][\w@.-]{2,99}`

Erforderlich: Ja

## Antwortsyntax

```
{
   "ServerId": "string",
   "User": {
      "Arn": "string",
      "HomeDirectory": "string",
      "HomeDirectoryMappings": [
         {
            "Entry": "string",
            "Target": "string",
            "Type": "string"
         }
      ],
      "HomeDirectoryType": "string",
      "Policy": "string",
      "PosixProfile": {
         "Gid": number,
         "SecondaryGids": [ number ],
         "Uid": number
      },
      "Role": "string",
      "SshPublicKeys": [
         {
            "DateImported": number,
            "SshPublicKeyBody": "string",
            "SshPublicKeyId": "string"
         }
      ],
      "Tags": [
         {
            "Key": "string",
            "Value": "string"
         }
      ],
      "UserName": "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.

ServerId

Eine vom System zugewiesene eindeutige Kennung für einen Server, dem dieser Benutzer zugewiesen wurde.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

User

Ein Array, das die Eigenschaften des Transfer Family Family-Benutzers für den von Ihnen angegebenen `ServerID` Wert enthält.

Typ: DescribedUser Objekt

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, weil der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Beispiele

### Beispiel

Das folgende Beispiel zeigt die Details für einen vorhandenen Benutzer.

Beispielanforderung

```
aws transfer describe-user --server-id s-1111aaaa2222bbbb3 --user-name bob-test
```

Beispielantwort

```
{
    "ServerId": "s-1111aaaa2222bbbb3",
    "User": {
        "Arn": "arn:aws:transfer:us-east-1:111122223333:user/s-1111aaaa2222bbbb3/bob-
test",
        "HomeDirectory": "/DOC-EXAMPLE-BUCKET",
        "HomeDirectoryType": "PATH",
        "Role": "arn:aws:iam::111122223333:role/bob-role",
        "SshPublicKeys": [
            {
                "DateImported": "2022-03-31T12:27:52.614000-04:00",
                "SshPublicKeyBody": "ssh-rsa AAAAB3NzaC1yc..... bobusa@mycomputer.us-
east-1.amaazon.com",
                "SshPublicKeyId": "key-abcde12345fghik67"
            }
        ],
        "Tags": [],
        "UserName": "bob-test"
```

```
    }
}
```

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# DescribeWorkflow

Beschreibt den angegebenen Workflow.

## Anforderungssyntax

```
{
   "WorkflowId": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

### WorkflowId

Eine eindeutige Kennung für den Workflow.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `w-([a-z0-9]{17})`

Erforderlich: Ja

## Antwortsyntax

```
{
   "Workflow": {
      "Arn": "string",
      "Description": "string",
      "OnExceptionSteps": [
         {
            "CopyStepDetails": {
               "DestinationFileLocation": {
                  "EfsFileLocation": {
                     "FileSystemId": "string",
                     "Path": "string"
                  },
```

```
            "S3FileLocation": {
               "Bucket": "string",
               "Key": "string"
            }
         },
         "Name": "string",
         "OverwriteExisting": "string",
         "SourceFileLocation": "string"
      },
      "CustomStepDetails": {
         "Name": "string",
         "SourceFileLocation": "string",
         "Target": "string",
         "TimeoutSeconds": number
      },
      "DecryptStepDetails": {
         "DestinationFileLocation": {
            "EfsFileLocation": {
               "FileSystemId": "string",
               "Path": "string"
            },
            "S3FileLocation": {
               "Bucket": "string",
               "Key": "string"
            }
         },
         "Name": "string",
         "OverwriteExisting": "string",
         "SourceFileLocation": "string",
         "Type": "string"
      },
      "DeleteStepDetails": {
         "Name": "string",
         "SourceFileLocation": "string"
      },
      "TagStepDetails": {
         "Name": "string",
         "SourceFileLocation": "string",
         "Tags": [
            {
               "Key": "string",
               "Value": "string"
            }
         ]
```

```
            },
            "Type": "string"
         }
      ],
      "Steps": [
         {
            "CopyStepDetails": {
               "DestinationFileLocation": {
                  "EfsFileLocation": {
                     "FileSystemId": "string",
                     "Path": "string"
                  },
                  "S3FileLocation": {
                     "Bucket": "string",
                     "Key": "string"
                  }
               },
               "Name": "string",
               "OverwriteExisting": "string",
               "SourceFileLocation": "string"
            },
            "CustomStepDetails": {
               "Name": "string",
               "SourceFileLocation": "string",
               "Target": "string",
               "TimeoutSeconds": number
            },
            "DecryptStepDetails": {
               "DestinationFileLocation": {
                  "EfsFileLocation": {
                     "FileSystemId": "string",
                     "Path": "string"
                  },
                  "S3FileLocation": {
                     "Bucket": "string",
                     "Key": "string"
                  }
               },
               "Name": "string",
               "OverwriteExisting": "string",
               "SourceFileLocation": "string",
               "Type": "string"
            },
            "DeleteStepDetails": {
```

```
                "Name": "string",
                "SourceFileLocation": "string"
            },
            "TagStepDetails": {
                "Name": "string",
                "SourceFileLocation": "string",
                "Tags": [
                    {
                        "Key": "string",
                        "Value": "string"
                    }
                ]
            },
            "Type": "string"
        }
    ],
    "Tags": [
        {
            "Key": "string",
            "Value": "string"
        }
    ],
    "WorkflowId": "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.

Workflow

Die Struktur, die die Details des Workflows enthält.

Typ: DescribedWorkflow Objekt

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# ImportCertificate

Importiert die Signier- und Verschlüsselungszertifikate, die Sie zum Erstellen lokaler Profile (AS2-Profile) und Partnerprofile benötigen.

## Anforderungssyntax

```
{
   "ActiveDate": number,
   "Certificate": "string",
   "CertificateChain": "string",
   "Description": "string",
   "InactiveDate": number,
   "PrivateKey": "string",
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ],
   "Usage": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

### ActiveDate

Ein optionales Datum, das angibt, wann das Zertifikat aktiv wird.

Typ: Zeitstempel

Erforderlich: Nein

### Certificate

- Geben Sie für die CLI einen Dateipfad für ein Zertifikat im URI-Format an. z. B. `--certificate file://encryption-cert.pem`. Alternativ können Sie den Rohinhalt bereitstellen.

- Geben Sie für das SDK den Rohinhalt einer Zertifikatsdatei an. Beispiel, `--certificate "`cat encryption-cert.pem`"`.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 16384 Zeichen.

Pattern: `[\u0009\u000A\u000D\u0020-\u00FF]*`

Erforderlich: Ja

## CertificateChain

Eine optionale Liste von Zertifikaten, die die Kette für das importierte Zertifikat bilden.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Die maximale Länge beträgt 2097152.

Pattern: `[\u0009\u000A\u000D\u0020-\u00FF]*`

Erforderlich: Nein

## Description

Eine kurze Beschreibung, die bei der Identifizierung des Zertifikats hilft.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 1. Höchstlänge = 200 Zeichen.

Pattern: `[\p{Graph}]+`

Erforderlich: Nein

## InactiveDate

Ein optionales Datum, das angibt, wann das Zertifikat inaktiv wird.

Typ: Zeitstempel

Erforderlich: Nein

PrivateKey

- Geben Sie für die CLI einen Dateipfad für einen privaten Schlüssel im URI-Format an. Zum Beispiel. `--private-key file://encryption-key.pem` Alternativ können Sie den Rohinhalt der Datei mit dem privaten Schlüssel angeben.
- Geben Sie für das SDK den Rohinhalt einer privaten Schlüsseldatei an. Beispiel: `--private-key "`cat encryption-key.pem`"`

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 16384 Zeichen.

Pattern: `[\u0009\u000A\u000D\u0020-\u00FF]*`

Erforderlich: Nein

Tags

Schlüssel-Wert-Paare, die zur Gruppierung und Suche von Zertifikaten verwendet werden.

Typ: Array von Tag-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Die maximale Anzahl beträgt 50 Elemente.

Erforderlich: Nein

Usage

Gibt an, wie dieses Zertifikat verwendet wird. Es kann auf folgende Weise verwendet werden:

- `SIGNING`: Zum Signieren von AS2-Nachrichten
- `ENCRYPTION`: Zum Verschlüsseln von AS2-Nachrichten
- `TLS`: Zur Sicherung von AS2-Kommunikation, die über HTTPS gesendet wird

Typ: Zeichenfolge

Zulässige Werte: `SIGNING | ENCRYPTION`

Erforderlich: Ja

## Antwortsyntax

```
{
```

```
    "CertificateId": "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.

CertificateId

Ein Array von Kennungen für die importierten Zertifikate. Sie verwenden diese Kennung für die Arbeit mit Profilen und Partnerprofilen.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 22.

Pattern: `cert-([0-9a-f]{17})`

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Beispiele

Beispiel

Im folgenden Beispiel wird ein Zertifikat importiert, das für die Verschlüsselung verwendet werden soll. Im ersten Befehl geben wir den Inhalt der Zertifikat- und Zertifikatskettendateien an. Verwenden Sie dieses Format für SDK-Befehle.

```
aws transfer import-certificate --usage ENCRYPTION --certificate "`cat encryption-
cert.pem`" \
    --private-key "`cat encryption-key.pem`" --certificate-chain "`cat root-ca.pem`"
```

Beispiel

Das folgende Beispiel ist identisch mit dem vorherigen Befehl, außer dass wir die Dateispeicherorte für die Dateien mit dem privaten Schlüssel, dem Zertifikat und der Zertifikatskette angeben. Diese Version des Befehls funktioniert nicht, wenn Sie ein SDK verwenden.

```
aws transfer import-certificate --usage ENCRYPTION --certificate file://encryption-
cert.pem \
    --private-key file://encryption-key.pem --certificate-chain file://root-ca.pem
```

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2

- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# ImportHostKey

Fügt dem Server, der durch den `ServerId` Parameter angegeben ist, einen Hostschlüssel hinzu.

## Anforderungssyntax

```
{
   "Description": "string",
   "HostKeyBody": "string",
   "ServerId": "string",
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ]
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

### Description

Die Textbeschreibung, die diesen Hostschlüssel identifiziert.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Höchstlänge = 200 Zeichen.

Pattern: `[\p{Print}]*`

Erforderlich: Nein

### HostKeyBody

Der private Schlüsselteil eines SSH-Schlüsselpaars.

AWS Transfer Family akzeptiert RSA-, ECDSA- und ED25519-Schlüssel.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge von 4096.

Erforderlich: Ja

ServerId

Die ID des Servers, der den Hostschlüssel enthält, den Sie importieren.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Erforderlich: Ja

Tags

Schlüssel-Wert-Paare, die zur Gruppierung und Suche nach Hostschlüsseln verwendet werden können.

Typ: Array von Tag-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Die maximale Anzahl beträgt 50 Elemente.

Erforderlich: Nein

## Antwortsyntax

```
{
   "HostKeyId": "string",
   "ServerId": "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.

HostKeyId

Gibt den Host-Schlüsselbezeichner für den importierten Schlüssel zurück.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 25.

Pattern: `hostkey-[0-9a-f]{17}`

ServerId

Gibt die Server-ID zurück, die den importierten Schlüssel enthält.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceExistsException

Die angeforderte Ressource ist nicht vorhanden oder befindet sich in einer anderen Region als der für den Befehl angegebenen.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

ThrottlingException

Die Anforderung wurde aufgrund der Drosselung von Anforderungen abgelehnt.

HTTP Status Code: 400

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# ImportSshPublicKey

Fügt einem Transfer Family-Benutzer, der durch einen `UserName` Wert identifiziert wird, der dem spezifischen Server mit aktiviertem File Transfer Protocol zugewiesen ist, einen öffentlichen Secure Shell (SSH) -Schlüssel hinzu, identifiziert durch. `ServerId`

Die Antwort gibt den `UserName` Wert, den `ServerId` Wert und den Namen von zurück. `SshPublicKeyId`

## Anforderungssyntax

```
{
   "ServerId": "string",
   "SshPublicKeyBody": "string",
   "UserName": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

ServerId

Ein vom System zugewiesener eindeutiger Bezeichner für einen Server.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Erforderlich: Ja

SshPublicKeyBody

Der öffentliche Schlüsselteil eines SSH-Schlüsselpaars.

AWS Transfer Family akzeptiert RSA-, ECDSA- und ED25519-Schlüssel.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 2048 Zeichen.

Erforderlich: Ja

UserName

Der Name des Transfer Family Family-Benutzers, der einem oder mehreren Servern zugewiesen ist.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 3. Maximale Länge beträgt 100 Zeichen.

Pattern: `[\w][\w@.-]{2,99}`

Erforderlich: Ja

## Antwortsyntax

```
{
   "ServerId": "string",
   "SshPublicKeyId": "string",
   "UserName": "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.

ServerId

Ein vom System zugewiesener eindeutiger Bezeichner für einen Server.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

SshPublicKeyId

Der Name, den das importierte System einem öffentlichen Schlüssel gegeben hat.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 21.

Pattern: `key-[0-9a-f]{17}`

UserName

Ein Benutzername, der dem von Ihnen angegebenen `ServerID` Wert zugewiesen wurde.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 3. Maximale Länge beträgt 100 Zeichen.

Pattern: `[\w][\w@.-]{2,99}`

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceExistsException

Die angeforderte Ressource ist nicht vorhanden oder befindet sich in einer anderen Region als der für den Befehl angegebenen.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

ThrottlingException

Die Anforderung wurde aufgrund der Drosselung von Anforderungen abgelehnt.

HTTP Status Code: 400

## Beispiele

### Beispiel

Mit diesem Befehl wird ein in der Datei gespeicherter ECDSA-Schlüssel importiert. `id_ecdsa.pub`

```
aws transfer import-ssh-public-key --server-id s-021345abcdef6789 --ssh-public-key-body
 file://id_ecdsa.pub --user-name jane-doe
```

### Beispiel

Wenn Sie den vorherigen Befehl ausführen, gibt das System die folgenden Informationen zurück.

```
{
    "ServerId": "s-021345abcdef6789",
   "SshPublicKeyId": "key-1234567890abcdef0",
   "UserName": "jane-doe"
}
```

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++

- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# ListAccesses

Listet die Details für alle Zugriffe auf, die Sie auf Ihrem Server haben.

## Anforderungssyntax

```
{
   "MaxResults": number,
   "NextToken": "string",
   "ServerId": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

MaxResults

Gibt die maximale Anzahl der zurückzugebenden Zugriffs-SIDs an.

Typ: Ganzzahl

Gültiger Bereich: Mindestwert 1. Maximaler Wert von 1 000.

Erforderlich: Nein

NextToken

Wenn Sie mit dem `ListAccesses` Aufruf zusätzliche Ergebnisse erzielen können, wird in der Ausgabe ein `NextToken` Parameter zurückgegeben. Sie können dann einen nachfolgenden Befehl an den `NextToken` Parameter übergeben, um weitere Zugriffe aufzulisten.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge von 6144.

Erforderlich: Nein

ServerId

Ein vom System zugewiesener eindeutiger Bezeichner für einen Server, dem Benutzer zugewiesen sind.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Erforderlich: Ja

## Antwortsyntax

```
{
   "Accesses": [
      {
         "ExternalId": "string",
         "HomeDirectory": "string",
         "HomeDirectoryType": "string",
         "Role": "string"
      }
   ],
   "NextToken": "string",
   "ServerId": "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.

### Accesses

Gibt die Zugriffe und ihre Eigenschaften für den von Ihnen angegebenen `ServerId` Wert zurück.

Typ: Array von ListedAccess-Objekten

### NextToken

Wenn Sie mit dem `ListAccesses` Aufruf zusätzliche Ergebnisse erzielen können, wird in der Ausgabe ein `NextToken` Parameter zurückgegeben. Sie können dann einen nachfolgenden Befehl an den `NextToken` Parameter übergeben, um weitere Zugriffe aufzulisten.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge von 6144.

ServerId

Ein vom System zugewiesener eindeutiger Bezeichner für einen Server, dem Benutzer zugewiesen sind.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidNextTokenException

Der übergebene `NextToken` Parameter ist ungültig.

HTTP Status Code: 400

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# ListAgreements

Gibt eine Liste der Vereinbarungen für den Server zurück, der durch den `ServerId` von Ihnen angegebenen Server identifiziert wird. Wenn Sie die Ergebnisse auf eine bestimmte Zahl beschränken möchten, geben Sie einen Wert für den `MaxResults` Parameter an. Wenn Sie den Befehl zuvor ausgeführt haben und einen Wert für erhalten haben`NextToken`, können Sie diesen Wert angeben, um die Liste der Vereinbarungen dort fortzusetzen, wo Sie aufgehört haben.

## Anforderungssyntax

```
{
   "MaxResults": number,
   "NextToken": "string",
   "ServerId": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

MaxResults

Die maximale Anzahl von Vereinbarungen, die zurückgegeben werden sollen.

Typ: Ganzzahl

Gültiger Bereich: Mindestwert 1. Maximaler Wert von 1 000.

Erforderlich: Nein

NextToken

Wenn Sie mit dem `ListAgreements` Aufruf zusätzliche Ergebnisse erzielen können, wird in der Ausgabe ein `NextToken` Parameter zurückgegeben. Sie können dann einen nachfolgenden Befehl an den `NextToken` Parameter übergeben, um weitere Vereinbarungen aufzulisten.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge von 6144.

Erforderlich: Nein

ServerId

Die ID des Servers, für den Sie eine Liste von Vereinbarungen wünschen.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Erforderlich: Ja

## Antwortsyntax

```
{
   "Agreements": [
      {
         "AgreementId": "string",
         "Arn": "string",
         "Description": "string",
         "LocalProfileId": "string",
         "PartnerProfileId": "string",
         "ServerId": "string",
         "Status": "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.

Agreements

Gibt ein Array zurück, in dem jedes Element die Details einer Vereinbarung enthält.

Typ: Array von ListedAgreement-Objekten

NextToken

Gibt ein Token zurück, das Sie verwenden können, um `ListAgreements` erneut aufzurufen und zusätzliche Ergebnisse zu erhalten, falls vorhanden.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge von 6144.

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidNextTokenException

Der übergebene `NextToken` Parameter ist ungültig.

HTTP Status Code: 400

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# ListCertificates

Gibt eine Liste der aktuellen Zertifikate zurück, in die importiert wurden AWS Transfer Family. Wenn Sie die Ergebnisse auf eine bestimmte Zahl beschränken möchten, geben Sie einen Wert für den `MaxResults` Parameter an. Wenn Sie den Befehl zuvor ausgeführt und einen Wert für den `NextToken` Parameter erhalten haben, können Sie diesen Wert angeben, um die Liste der Zertifikate dort fortzusetzen, wo Sie aufgehört haben.

## Anforderungssyntax

```
{
   "MaxResults": number,
   "NextToken": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

### MaxResults

Die maximale Anzahl von Zertifikaten, die zurückgegeben werden sollen.

Typ: Ganzzahl

Gültiger Bereich: Mindestwert 1. Maximaler Wert von 1 000.

Erforderlich: Nein

### NextToken

Wenn Sie mit dem `ListCertificates` Aufruf zusätzliche Ergebnisse erzielen können, wird in der Ausgabe ein `NextToken` Parameter zurückgegeben. Sie können dann einen nachfolgenden Befehl an den `NextToken` Parameter übergeben, um weitere Zertifikate aufzulisten.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge von 6144.

Erforderlich: Nein

## Antwortsyntax

```
{
   "Certificates": [
      {
         "ActiveDate": number,
         "Arn": "string",
         "CertificateId": "string",
         "Description": "string",
         "InactiveDate": number,
         "Status": "string",
         "Type": "string",
         "Usage": "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.

### Certificates

Gibt ein Array der Zertifikate zurück, die im `ListCertificates` Aufruf angegeben wurden.

Typ: Array von ListedCertificate-Objekten

### NextToken

Gibt das nächste Token zurück, mit dem Sie das nächste Zertifikat auflisten können.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge von 6144.

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidNextTokenException

Der übergebene `NextToken` Parameter ist ungültig.

HTTP Status Code: 400

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3

- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# ListConnectors

Listet die Konnektoren für die angegebene Region auf.

## Anforderungssyntax

```
{
   "MaxResults": number,
   "NextToken": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

MaxResults

Die maximale Anzahl von Anschlüssen, die zurückgegeben werden sollen.

Typ: Ganzzahl

Gültiger Bereich: Mindestwert 1. Maximaler Wert von 1 000.

Erforderlich: Nein

NextToken

Wenn Sie mit dem `ListConnectors` Aufruf zusätzliche Ergebnisse erzielen können, wird in der Ausgabe ein `NextToken` Parameter zurückgegeben. Sie können dann einen nachfolgenden Befehl an den `NextToken` Parameter übergeben, um weitere Konnektoren aufzulisten.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge von 6144.

Erforderlich: Nein

## Antwortsyntax

```
{
```

```
   "Connectors": [
      {
         "Arn": "string",
         "ConnectorId": "string",
         "Url": "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.

Connectors

Gibt ein Array zurück, in dem jedes Element die Details eines Verbinders enthält.

Typ: Array von ListedConnector-Objekten

NextToken

Gibt ein Token zurück, das Sie verwenden können, um `ListConnectors` erneut aufzurufen und zusätzliche Ergebnisse zu erhalten, falls vorhanden.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge von 6144.

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidNextTokenException

Der übergebene `NextToken` Parameter ist ungültig.

HTTP Status Code: 400

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, weil der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# ListExecutions

Listet alle laufenden Ausführungen für den angegebenen Workflow auf.

> **Note**
>
> Wenn die angegebene Workflow-ID nicht gefunden werden kann, wird eine `ListExecutions` `ResourceNotFound` Ausnahme zurückgegeben.

## Anforderungssyntax

```
{
   "MaxResults": number,
   "NextToken": "string",
   "WorkflowId": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

MaxResults

Gibt die maximale Anzahl zurückzugebender Ausführungen an.

Typ: Ganzzahl

Gültiger Bereich: Mindestwert 1. Maximaler Wert von 1 000.

Erforderlich: Nein

NextToken

`ListExecutions`gibt den `NextToken` Parameter in der Ausgabe zurück. Sie können den `NextToken` Parameter dann in einem nachfolgenden Befehl übergeben, um weitere Ausführungen aufzulisten.

Dies ist beispielsweise für die Paginierung nützlich. Wenn Sie 100 Ausführungen für einen Workflow haben, möchten Sie vielleicht nur die ersten 10 auflisten. Wenn ja, rufen Sie die API auf, indem Sie Folgendes `max-results` angeben:

```
aws transfer list-executions --max-results 10
```

Dadurch werden Details für die ersten 10 Ausführungen sowie der Zeiger (`NextToken`) auf die elfte Ausführung zurückgegeben. Sie können die API jetzt erneut aufrufen und den `NextToken` Wert angeben, den Sie erhalten haben:

```
aws transfer list-executions --max-results 10 --next-token
$somePointerReturnedFromPreviousListResult
```

Dieser Aufruf gibt die nächsten 10 Ausführungen zurück, von der 11. bis zur 20. Sie können den Anruf dann wiederholen, bis die Details für alle 100 Ausführungen zurückgegeben wurden.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge von 6144.

Erforderlich: Nein

WorkflowId

Eine eindeutige Kennung für den Workflow.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `w-([a-z0-9]{17})`

Erforderlich: Ja

## Antwortsyntax

```
{
   "Executions": [
      {
         "ExecutionId": "string",
         "InitialFileLocation": {
            "EfsFileLocation": {
               "FileSystemId": "string",
```

```
            "Path": "string"
         },
         "S3FileLocation": {
            "Bucket": "string",
            "Etag": "string",
            "Key": "string",
            "VersionId": "string"
         }
      },
      "ServiceMetadata": {
         "UserDetails": {
            "ServerId": "string",
            "SessionId": "string",
            "UserName": "string"
         }
      },
      "Status": "string"
   }
],
"NextToken": "string",
"WorkflowId": "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.

### Executions

Gibt die Details für jede Ausführung in einem `ListedExecution` Array zurück.

Typ: Array von ListedExecution-Objekten

### NextToken

`ListExecutions`gibt den `NextToken` Parameter in der Ausgabe zurück. Sie können den `NextToken` Parameter dann in einem nachfolgenden Befehl übergeben, um weitere Ausführungen aufzulisten.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge von 6144.

WorkflowId

Eine eindeutige Kennung für den Workflow.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `w-([a-z0-9]{17})`

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidNextTokenException

Der übergebene `NextToken` Parameter ist ungültig.

HTTP Status Code: 400

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# ListHostKeys

Gibt eine Liste von Hostschlüsseln für den Server zurück, der durch den `ServerId` Parameter angegeben ist.

## Anforderungssyntax

```
{
   "MaxResults": number,
   "NextToken": "string",
   "ServerId": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

### MaxResults

Die maximale Anzahl von Hostschlüsseln, die zurückgegeben werden sollen.

Typ: Ganzzahl

Gültiger Bereich: Mindestwert 1. Maximaler Wert von 1 000.

Erforderlich: Nein

### NextToken

Wenn es zusätzliche Ergebnisse gibt, die nicht zurückgegeben wurden, wird ein `NextToken` Parameter zurückgegeben. Sie können diesen Wert für einen nachfolgenden Aufruf verwenden, `ListHostKeys` um mit der Auflistung der Ergebnisse fortzufahren.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge von 6144.

Erforderlich: Nein

### ServerId

Die ID des Servers, der die Hostschlüssel enthält, die Sie anzeigen möchten.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Erforderlich: Ja

## Antwortsyntax

```
{
   "HostKeys": [
      {
         "Arn": "string",
         "DateImported": number,
         "Description": "string",
         "Fingerprint": "string",
         "HostKeyId": "string",
         "Type": "string"
      }
   ],
   "NextToken": "string",
   "ServerId": "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.

### HostKeys

Gibt ein Array zurück, in dem jedes Element die Details eines Hostschlüssels enthält.

Typ: Array von ListedHostKey-Objekten

### NextToken

Gibt ein Token zurück, das Sie verwenden können, um `ListHostKeys` erneut aufzurufen und zusätzliche Ergebnisse zu erhalten, falls vorhanden.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge von 6144.

ServerId

Gibt den Serverbezeichner zurück, der die aufgelisteten Hostschlüssel enthält.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidNextTokenException

Der übergebene `NextToken` Parameter ist ungültig.

HTTP Status Code: 400

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# ListProfiles

Gibt eine Liste der Profile für Ihr System zurück. Wenn Sie die Ergebnisse auf eine bestimmte Zahl beschränken möchten, geben Sie einen Wert für den `MaxResults` Parameter an. Wenn Sie den Befehl zuvor ausgeführt haben und einen Wert für erhalten haben`NextToken`, können Sie diesen Wert angeben, um die Profile dort aufzulisten, wo Sie aufgehört haben.

## Anforderungssyntax

```
{
   "MaxResults": number,
   "NextToken": "string",
   "ProfileType": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

MaxResults

Die maximale Anzahl von Profilen, die zurückgegeben werden sollen.

Typ: Ganzzahl

Gültiger Bereich: Mindestwert 1. Maximaler Wert von 1 000.

Erforderlich: Nein

NextToken

Wenn es zusätzliche Ergebnisse gibt, die nicht zurückgegeben wurden, wird ein `NextToken` Parameter zurückgegeben. Sie können diesen Wert für einen nachfolgenden Aufruf verwenden, `ListProfiles` um mit der Auflistung der Ergebnisse fortzufahren.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge von 6144.

Erforderlich: Nein

ProfileType

Gibt an, ob nur LOCAL- oder nur PARTNER-Typprofile aufgelistet werden sollen. Sind diese nicht in der Anforderung nicht, listet der Befehl alle Profilarten auf.

Typ: Zeichenfolge

Zulässige Werte: LOCAL | PARTNER

Erforderlich: Nein

## Antwortsyntax

```
{
   "NextToken": "string",
   "Profiles": [
      {
         "Arn": "string",
         "As2Id": "string",
         "ProfileId": "string",
         "ProfileType": "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

Gibt ein Token zurück, das Sie verwenden können, um ListProfiles erneut aufzurufen und weitere Ergebnisse zu erhalten, falls es welche gibt.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge von 6144.

Profiles

Gibt ein Array zurück, in dem jedes Element die Details eines Profils enthält.

Typ: Array von ListedProfile-Objekten

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidNextTokenException

Der übergebene `NextToken` Parameter ist ungültig.

HTTP Status Code: 400

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle

- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# ListSecurityPolicies

Listet die Sicherheitsrichtlinien auf, die an Ihre Server und SFTP-Connectors angehängt sind. Weitere Informationen zu Sicherheitsrichtlinien finden Sie unter Arbeiten mit Sicherheitsrichtlinien für Server oder Arbeiten mit Sicherheitsrichtlinien für SFTP-Connectors.

## Anforderungssyntax

```
{
   "MaxResults": number,
   "NextToken": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

MaxResults

Gibt die Anzahl der Sicherheitsrichtlinien an, die als Antwort auf die `ListSecurityPolicies` Abfrage zurückgegeben werden sollen.

Typ: Ganzzahl

Gültiger Bereich: Mindestwert 1. Maximaler Wert von 1 000.

Erforderlich: Nein

NextToken

Wenn mit dem `ListSecurityPolicies` Befehl zusätzliche Ergebnisse erzielt werden, wird in der Ausgabe ein `NextToken` Parameter zurückgegeben. Sie können den `NextToken` Parameter dann in einem nachfolgenden Befehl übergeben, um weitere Sicherheitsrichtlinien aufzulisten.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge von 6144.

Erforderlich: Nein

## Antwortsyntax

```
{
   "NextToken": "string",
   "SecurityPolicyNames": [ "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

Wenn Sie zusätzliche Ergebnisse aus dem `ListSecurityPolicies` Vorgang erhalten können, wird in der Ausgabe ein `NextToken` Parameter zurückgegeben. In einem folgenden Befehl können Sie den `NextToken` Parameter übergeben, um mit der Auflistung der Sicherheitsrichtlinien fortzufahren.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge von 6144.

SecurityPolicyNames

Eine Reihe von Sicherheitsrichtlinien, die aufgelistet wurden.

Typ: Zeichenfolgen-Array

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 100 Zeichen.

Pattern: `Transfer[A-Za-z0-9]*SecurityPolicy-[A-Za-z0-9-]+`

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidNextTokenException

Der übergebene `NextToken` Parameter ist ungültig.

HTTP Status Code: 400

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Beispiele

### Beispiel

Im folgenden Beispiel werden die Namen aller verfügbaren Sicherheitsrichtlinien aufgeführt.

Beispielanforderung

```
aws transfer list-security-policies
```

Beispielantwort

```
{
    "SecurityPolicyNames": [
        "TransferSecurityPolicy-2023-05",
        "TransferSecurityPolicy-2022-03",
        "TransferSecurityPolicy-FIPS-2024-01",
        "TransferSecurityPolicy-2024-01",
        "TransferSecurityPolicy-PQ-SSH-FIPS-Experimental-2023-04",
        "TransferSecurityPolicy-PQ-SSH-Experimental-2023-04",
        "TransferSecurityPolicy-FIPS-2020-06",
        "TransferSecurityPolicy-2020-06",
        "TransferSecurityPolicy-2018-11",
        "TransferSecurityPolicy-FIPS-2023-05"
```

```
    ]
}
```

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# ListServers

Führt die Server auf, die das Dateiübertragungsprotokoll aktivieren und Ihrem Konto zugeordnet sind. AWS

## Anforderungssyntax

```
{
   "MaxResults": number,
   "NextToken": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

MaxResults

Gibt die Anzahl der Server an, die als Antwort auf die Abfrage zurückgegeben werden sollen. `ListServers`

Typ: Ganzzahl

Gültiger Bereich: Mindestwert 1. Maximaler Wert von 1 000.

Erforderlich: Nein

NextToken

Wenn mit dem `ListServers` Befehl zusätzliche Ergebnisse erzielt werden, wird in der Ausgabe ein `NextToken` Parameter zurückgegeben. Sie können den `NextToken` Parameter dann in einem nachfolgenden Befehl übergeben, um weitere Server aufzulisten.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge von 6144.

Erforderlich: Nein

## Antwortsyntax

```
{
   "NextToken": "string",
   "Servers": [
      {
         "Arn": "string",
         "Domain": "string",
         "EndpointType": "string",
         "IdentityProviderType": "string",
         "LoggingRole": "string",
         "ServerId": "string",
         "State": "string",
         "UserCount": number
      }
   ]
}
```

## 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

Wenn Sie zusätzliche Ergebnisse aus dem `ListServers` Vorgang erhalten können, wird in der Ausgabe ein `NextToken` Parameter zurückgegeben. In einem folgenden Befehl können Sie den `NextToken` Parameter übergeben, um weitere Server aufzulisten.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge von 6144.

### Servers

Eine Reihe von Servern, die aufgelistet wurden.

Typ: Array von ListedServer-Objekten

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidNextTokenException

Der übergebene `NextToken` Parameter ist ungültig.

HTTP Status Code: 400

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Beispiele

### Beispiel

Das folgende Beispiel listet die Server auf, die in Ihrem vorhanden sind AWS-Konto.

Beachten Sie, dass die `NextToken` Beispielwerte nicht real sind: Sie sollen angeben, wie der Parameter verwendet wird.

### Beispielanforderung

```
{
   "MaxResults": 1,
   "NextToken": "token-from-previous-API-call"
}
```

Beispielantwort

```
{
   "NextToken": "another-token-to-continue-listing",
   "Servers": [
      {
         "Arn": "arn:aws:transfer:us-east-1:111112222222:server/s-01234567890abcdef",
         "Domain": "S3",
         "IdentityProviderType": "SERVICE_MANAGED",
         "EndpointType": "PUBLIC",
         "LoggingRole": "arn:aws:iam::111112222222:role/my-role",
         "ServerId": "s-01234567890abcdef",
         "State": "ONLINE",
         "UserCount": 3
      }
   ]
}
```

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# ListTagsForResource

Führt alle Tags auf, die mit dem von Ihnen angegebenen Amazon-Ressourcennamen (ARN) verknüpft sind. Bei der Ressource kann es sich um einen Benutzer, einen Server oder eine Rolle handeln.

## Anforderungssyntax

```
{
   "Arn": "string",
   "MaxResults": number,
   "NextToken": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

Arn

Fordert die Tags an, die einem bestimmten Amazon-Ressourcennamen (ARN) zugeordnet sind. Ein ARN ist eine Kennung für eine bestimmte AWS Ressource, z. B. einen Server, einen Benutzer oder eine Rolle.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 1600 Zeichen.

Pattern: `arn:\S+`

Erforderlich: Ja

MaxResults

Gibt die Anzahl der Tags an, die als Antwort auf die `ListTagsForResource` Anfrage zurückgegeben werden sollen.

Typ: Ganzzahl

Gültiger Bereich: Mindestwert 1. Maximaler Wert von 1 000.

Erforderlich: Nein

NextToken

Wenn Sie zusätzliche Ergebnisse der `ListTagsForResource` Operation anfordern, wird in der Eingabe ein `NextToken` Parameter zurückgegeben. Sie können dann einen nachfolgenden Befehl an den `NextToken` Parameter übergeben, um weitere Tags aufzulisten.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge von 6144.

Erforderlich: Nein

## Antwortsyntax

```
{
   "Arn": "string",
   "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.

Arn

Der ARN, den Sie angegeben haben, um die Tags aufzulisten.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 1600 Zeichen.

Pattern: `arn:\S+`

NextToken

Wenn Sie mit dem `ListTagsForResource` Aufruf zusätzliche Ergebnisse erzielen können, wird in der Ausgabe ein `NextToken` Parameter zurückgegeben. Sie können dann einen nachfolgenden Befehl an den `NextToken` Parameter übergeben, um weitere Tags aufzulisten.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge von 6144.

Tags

Schlüssel-Wert-Paare, die einer Ressource zugewiesen werden, in der Regel zum Gruppieren und Suchen nach Elementen. Tags sind Metadaten, die Sie definieren.

Typ: Array von Tag-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Die maximale Anzahl beträgt 50 Elemente.

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidNextTokenException

Der übergebene `NextToken` Parameter ist ungültig.

HTTP Status Code: 400

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Beispiele

### Beispiel

Das folgende Beispiel listet die Tags für die Ressource mit dem von Ihnen angegebenen ARN auf.

Beispielanforderung

```
{
   "Arn": "arn:aws:transfer:us-east-1:176354371281:server/s-01234567890abcdef"
}
```

### Beispiel

Dieses Beispiel veranschaulicht eine Verwendung von ListTagsForResource.

Beispielantwort

```
{
   "Tags": [
      {
         "Key": "Name",
         "Value": "MyServer"
      }
   ]
}
```

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2

- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# ListUsers

Führt die Benutzer für einen Server mit aktiviertem Dateiübertragungsprotokoll auf, den Sie durch Übergabe des Parameters angeben. `ServerId`

## Anforderungssyntax

```
{
   "MaxResults": number,
   "NextToken": "string",
   "ServerId": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

MaxResults

Gibt die Anzahl der Benutzer an, die als Antwort auf die Anfrage zurückgegeben werden sollen. `ListUsers`

Typ: Ganzzahl

Gültiger Bereich: Mindestwert 1. Maximaler Wert von 1 000.

Erforderlich: Nein

NextToken

Wenn der `ListUsers` Aufruf zusätzliche Ergebnisse liefert, wird in der Ausgabe ein `NextToken` Parameter zurückgegeben. Sie können das dann `NextToken` an einen nachfolgenden `ListUsers` Befehl übergeben, um weitere Benutzer aufzulisten.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge von 6144.

Erforderlich: Nein

### ServerId

Ein vom System zugewiesener eindeutiger Bezeichner für einen Server, dem Benutzer zugewiesen sind.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Erforderlich: Ja

## Antwortsyntax

```
{
   "NextToken": "string",
   "ServerId": "string",
   "Users": [
      {
         "Arn": "string",
         "HomeDirectory": "string",
         "HomeDirectoryType": "string",
         "Role": "string",
         "SshPublicKeyCount": number,
         "UserName": "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

Wenn Sie mit dem `ListUsers` Aufruf zusätzliche Ergebnisse erzielen können, wird in der Ausgabe ein `NextToken` Parameter zurückgegeben. Sie können dann einen nachfolgenden Befehl an den `NextToken` Parameter übergeben, um weitere Benutzer aufzulisten.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge von 6144.

ServerId

Ein vom System zugewiesener eindeutiger Bezeichner für einen Server, dem die Benutzer zugewiesen sind.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Users

Gibt die Transfer Family Family-Benutzer und ihre Eigenschaften für den von Ihnen angegebenen `ServerId` Wert zurück.

Typ: Array von ListedUser-Objekten

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidNextTokenException

Der übergebene `NextToken` Parameter ist ungültig.

HTTP Status Code: 400

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Beispiele

### Beispiel

Der `ListUsers` API-Aufruf gibt eine Liste von Benutzern zurück, die einem von Ihnen angegebenen Server zugeordnet sind.

Beispielanforderung

```
        {
         "MaxResults": 100,
         "NextToken": "eyJNYXJrZXIiOiBudWxsLCAiYm90b1X0cnVuU2F0ZV9hbW91bnQiOiAyfQ==",
         "ServerId": "s-01234567890abcdef"
         }
```

### Beispiel

Dies ist eine Beispielantwort für diesen API-Aufruf.

Beispielantwort

```
{
   "NextToken": "eyJNYXJrZXIiOiBudWxsLCAiYm90b1X0cnVuU2F0ZV9hbW91bnQiOiAyfQ==",
   "ServerId": "s-01234567890abcdef",
   "Users": [
      {
         "Arn": "arn:aws:transfer:us-east-1:176354371281:user/s-01234567890abcdef/
charlie",
```

```
         "HomeDirectory": "/tests/home/charlie",
         "SshPublicKeyCount": 1,
         "Role": "arn:aws:iam::176354371281:role/transfer-role1",
         "Tags": [
            {
               "Key": "Name",
               "Value": "user1"
            }
         ],
         "UserName": "my_user"
      }
   ]
}
```

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# ListWorkflows

Listet alle Workflows auf, die mit Ihrem AWS-Konto für Ihre aktuelle Region verknüpft sind.

## Anforderungssyntax

```
{
   "MaxResults": number,
   "NextToken": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

MaxResults

Gibt die maximale Anzahl von Workflows an, die zurückgegeben werden sollen.

Typ: Ganzzahl

Gültiger Bereich: Mindestwert 1. Maximaler Wert von 1 000.

Erforderlich: Nein

NextToken

`ListWorkflows`gibt den `NextToken` Parameter in der Ausgabe zurück. Sie können den `NextToken` Parameter dann in einem nachfolgenden Befehl übergeben, um weitere Workflows aufzulisten.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge von 6144.

Erforderlich: Nein

## Antwortsyntax

```
{
```

```
   "NextToken": "string",
   "Workflows": [
      {
         "Arn": "string",
         "Description": "string",
         "WorkflowId": "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

`ListWorkflows`gibt den `NextToken` Parameter in der Ausgabe zurück. Sie können den `NextToken` Parameter dann in einem nachfolgenden Befehl übergeben, um weitere Workflows aufzulisten.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge von 6144.

Workflows

Gibt `Arn``WorkflowId`, und `Description` für jeden Workflow zurück.

Typ: Array von ListedWorkflow-Objekten

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidNextTokenException

Der übergebene `NextToken` Parameter ist ungültig.

HTTP Status Code: 400

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# SendWorkflowStepState

Sendet einen Callback für asynchrone benutzerdefinierte Schritte.

Die `ExecutionIdWorkflowId`, und `Token` werden während der Ausführung eines benutzerdefinierten Schritts eines Workflows an die Zielressource übergeben. Sie müssen diese in ihren Rückruf aufnehmen und einen Status angeben.

## Anforderungssyntax

```
{
   "ExecutionId": "string",
   "Status": "string",
   "Token": "string",
   "WorkflowId": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

### ExecutionId

Eine eindeutige Kennung für die Ausführung eines Workflows.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 36.

Pattern: `[0-9a-fA-F]{8}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{12}`

Erforderlich: Ja

### Status

Gibt an, ob der angegebene Schritt erfolgreich war oder nicht.

Typ: Zeichenfolge

Zulässige Werte: `SUCCESS | FAILURE`

Erforderlich: Ja

Token

Wird verwendet, um zwischen mehreren Callbacks für mehrere Lambda-Schritte innerhalb derselben Ausführung zu unterscheiden.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 64 Zeichen.

Pattern: `\w+`

Erforderlich: Ja

WorkflowId

Eine eindeutige Kennung für den Workflow.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `w-([a-z0-9]{17})`

Erforderlich: Ja

## Antwortelemente

Wenn die Aktion erfolgreich ist, gibt der Dienst eine HTTP 200-Antwort mit leerem HTTP-Textinhalt zurück.

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

AccessDeniedException

Sie haben keinen ausreichenden Zugriff zum Durchführen dieser Aktion.

HTTP Status Code: 400

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

ThrottlingException

Die Anforderung wurde aufgrund der Drosselung von Anforderungen abgelehnt.

HTTP Status Code: 400

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3

- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# StartDirectoryListing

Ruft eine Liste des Inhalts eines Verzeichnisses von einem Remote-SFTP-Server ab. Sie geben die Connector-ID, den Ausgabepfad und den Remote-Verzeichnispfad an. Sie können auch den optionalen `MaxItems` Wert angeben, um die maximale Anzahl von Elementen zu steuern, die im Remote-Verzeichnis aufgelistet werden. Diese API gibt eine Liste aller Dateien und Verzeichnisse im Remote-Verzeichnis zurück (bis zum Maximalwert), gibt jedoch keine Dateien oder Ordner in Unterverzeichnissen zurück. Das heißt, sie gibt nur eine Liste von Dateien und Verzeichnissen zurück, die eine Ebene tief sind.

Nachdem Sie die Auflistungsdatei erhalten haben, können Sie die Dateien, die Sie übertragen möchten, an den `RetrieveFilePaths` Parameter des `StartFileTransfer` API-Aufrufs übergeben.

Die Benennungskonvention für die Ausgabedatei lautet *`connector-ID-listing-ID`*`.json`. Die Ausgabedatei enthält die folgenden Informationen:

- `filePath`: Der vollständige Pfad einer Remote-Datei, relativ zum Verzeichnis der Listing-Anfrage für Ihren SFTP-Connector auf dem Remoteserver.
- `modifiedTimestamp`: das letzte Mal, als die Datei geändert wurde, im UTC-Zeitformat. Dies ist ein optionales Feld. Wenn die Attribute der entfernten Datei keinen Zeitstempel enthalten, wird er in der Dateiliste weggelassen.
- `size`: Die Größe der Datei in Byte. Dies ist ein optionales Feld. Wenn die Remote-Dateiattribute keine Dateigröße enthalten, wird sie in der Dateiliste weggelassen.
- `path`: der vollständige Pfad eines Remote-Verzeichnisses, relativ zum Verzeichnis der Listing-Anfrage für Ihren SFTP-Connector auf dem Remoteserver.
- `truncated`: ein Flag, das angibt, ob die Listenausgabe alle im Remote-Verzeichnis enthaltenen Elemente enthält oder nicht. Wenn Ihr `Truncated` Ausgabewert wahr ist, können Sie den im optionalen `max-items` Eingabeattribut angegebenen Wert erhöhen, um mehr Elemente auflisten zu können (bis zur maximal zulässigen Listengröße von 10.000 Elementen).

## Anforderungssyntax

```
{
   "ConnectorId": "string",
   "MaxItems": number,
   "OutputDirectoryPath": "string",
```

```
    "RemoteDirectoryPath": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

ConnectorId

Die eindeutige Kennung für den Konnektor.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `c-([0-9a-f]{17})`

Erforderlich: Ja

MaxItems

Ein optionaler Parameter, mit dem Sie die maximale Anzahl der abzurufenden Datei-/Verzeichnisnamen angeben können. Der Standardwert lautet 1.000.

Typ: Ganzzahl

Gültiger Bereich: Mindestwert 1. Maximaler Wert von 10000.

Erforderlich: Nein

OutputDirectoryPath

Gibt den Pfad (Bucket und Präfix) im Amazon S3 S3-Speicher an, um die Ergebnisse der Verzeichnisliste zu speichern.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 1. Maximale Länge beträgt 1024 Zeichen.

Pattern: `(.)+`

Erforderlich: Ja

RemoteDirectoryPath

Gibt das Verzeichnis auf dem Remote-SFTP-Server an, dessen Inhalt Sie auflisten möchten.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 1. Maximale Länge beträgt 1024 Zeichen.

Pattern: `(.)+`

Erforderlich: Ja

## Antwortsyntax

```
{
   "ListingId": "string",
   "OutputFileName": "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.

ListingId

Gibt einen eindeutigen Bezeichner für den Verzeichnisauflistungsaufruf zurück.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge von 512.

Pattern: `[0-9a-zA-Z./-]+`

OutputFileName

Gibt den Namen der Datei zurück, in der die Ergebnisse gespeichert sind. Dies ist eine Kombination aus der Connector-ID und der Listing-ID:`<connector-id>-<listing-id>.json`.

Typ: Zeichenfolge

Längenbeschränkungen: Mindestlänge von 26. Die maximale Länge beträgt 537.

Pattern: `c-([0-9a-f]{17})-[0-9a-zA-Z./-]+.json`

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

ThrottlingException

Die Anforderung wurde aufgrund der Drosselung von Anforderungen abgelehnt.

HTTP Status Code: 400

## Beispiele

Beispiel

Im folgenden Beispiel wird der Inhalt des `home` Ordners auf dem Remote-SFTP-Server aufgeführt, der durch den angegebenen Connector identifiziert wird. Die Ergebnisse werden am Amazon S3

S3-Speicherort `/DOC-EXAMPLE-BUCKET/connector-files` und in einer Datei mit dem Namen abgelegt`c-AAAA1111BBBB2222C-6666abcd-11aa-22bb-cc33-0000aaaa3333.json`.

### Beispielanforderung

```
{
    "ConnectorId": "c-AAAA1111BBBB2222C",
    "MaxItems": "10",
    "OutputDirectoryPath": "/DOC-EXAMPLE-BUCKET/connector-files",
    "RemoteDirectoryPath": "/home"
}
```

### Beispielantwort

```
{
    "ListingId": "6666abcd-11aa-22bb-cc33-0000aaaa3333",
    "OutputFileName": "c-AAAA1111BBBB2222C-6666abcd-11aa-22bb-cc33-0000aaaa3333.json"
}
```

```
// under bucket "DOC-EXAMPLE-BUCKET"
connector-files/c-AAAA1111BBBB2222C-6666abcd-11aa-22bb-cc33-0000aaaa3333.json
{
    "files": [
        {
            "filePath": "/home/what.txt",
            "modifiedTimestamp": "2024-01-30T20:34:54Z",
            "size" : 2323
        },
        {
            "filePath": "/home/how.pgp",
            "modifiedTimestamp": "2024-01-30T20:34:54Z",
            "size" : 51238
        }
    ],
    "paths": [
        {
            "path": "/home/magic"
        },
        {
            "path": "/home/aws"
        },
    ],
```

```
    "truncated": false
}
```

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# StartFileTransfer

Beginnt eine Dateiübertragung zwischen dem lokalen AWS Speicher und einem Remote-AS2- oder SFTP-Server.

- Bei einem AS2-Connector geben Sie den `ConnectorId` und einen oder mehrere an, um die Dateien `SendFilePaths` zu identifizieren, die Sie übertragen möchten.
- Bei einem SFTP-Connector kann die Dateiübertragung entweder ausgehend oder eingehend erfolgen. In beiden Fällen geben Sie die an. `ConnectorId` Abhängig von der Richtung der Übertragung geben Sie auch die folgenden Elemente an:
  - Wenn Sie eine Datei vom SFTP-Server eines Partners in den Amazon Web Services Services-Speicher übertragen, geben Sie eine oder mehrere an, `RetrieveFilePaths` um die Dateien zu identifizieren, die Sie übertragen möchten, und a, `LocalDirectoryPath` um den Zielordner anzugeben.
  - Wenn Sie eine Datei vom AWS Speicher auf den SFTP-Server eines Partners übertragen, geben Sie eine oder mehrere an, um die Dateien `SendFilePaths` zu identifizieren, die Sie übertragen möchten, und a, `RemoteDirectoryPath` um den Zielordner anzugeben.

## Anforderungssyntax

```
{
   "ConnectorId": "string",
   "LocalDirectoryPath": "string",
   "RemoteDirectoryPath": "string",
   "RetrieveFilePaths": [ "string" ],
   "SendFilePaths": [ "string" ]
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

### ConnectorId

Die eindeutige Kennung für den Connector.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `c-([0-9a-f]{17})`

Erforderlich: Ja

### LocalDirectoryPath

Bei einer eingehenden Übertragung `LocaDirectoryPath` gibt der das Ziel für eine oder mehrere Dateien an, die vom SFTP-Server des Partners übertragen werden.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 1. Maximale Länge beträgt 1024 Zeichen.

Pattern: `(.)+`

Erforderlich: Nein

### RemoteDirectoryPath

Bei einer ausgehenden Übertragung `RemoteDirectoryPath` gibt der das Ziel für eine oder mehrere Dateien an, die auf den SFTP-Server des Partners übertragen werden. Wenn Sie kein angeben`RemoteDirectoryPath`, ist das Ziel für die übertragenen Dateien das Home-Verzeichnis des SFTP-Benutzers.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 1. Maximale Länge beträgt 1024 Zeichen.

Pattern: `(.)+`

Erforderlich: Nein

### RetrieveFilePaths

Ein oder mehrere Quellpfade für den SFTP-Server des Partners. Jede Zeichenfolge steht für einen Quelldateipfad für eine eingehende Dateiübertragung.

Typ: Zeichenfolgen-Array

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Die maximale Anzahl beträgt 10 Elemente.

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 1024 Zeichen.

Pattern: `(.)+`

Erforderlich: Nein

SendFilePaths

Ein oder mehrere Quellpfade für den Amazon S3 S3-Speicher. Jede Zeichenfolge steht für einen Quelldateipfad für eine ausgehende Dateiübertragung. z. B. *DOC-EXAMPLE-BUCKET/myfile.txt* .

> **Note**
>
> *DOC-EXAMPLE-BUCKET* Ersetzen Sie ihn durch einen Ihrer aktuellen Buckets.

Typ: Zeichenfolgen-Array

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Die maximale Anzahl beträgt 10 Elemente.

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 1024 Zeichen.

Pattern: `(.)+`

Erforderlich: Nein

## Antwortsyntax

```
{
   "TransferId": "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.

TransferId

Gibt den eindeutigen Bezeichner für die Dateiübertragung zurück.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge von 512.

Pattern: `[0-9a-zA-Z./-]+`

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

ThrottlingException

Die Anforderung wurde aufgrund der Drosselung von Anforderungen abgelehnt.

HTTP Status Code: 400

## Beispiele

### Beispiel

Im folgenden Beispiel wird eine AS2-Dateiübertragung von einem Transfer Family Family-Server zum Endpunkt eines Remote-Handelspartners gestartet. *DOC-EXAMPLE-BUCKET* Ersetzen Sie es durch einen Ihrer aktuellen Buckets.

### Beispielanforderung

```
{
    "ConnectorId": "c-AAAA1111BBBB2222C",
    "SendFilePaths": [
        "/DOC-EXAMPLE-BUCKET/myfile-1.txt",
        "/DOC-EXAMPLE-BUCKET/myfile-2.txt",
        "/DOC-EXAMPLE-BUCKET/myfile-3.txt"
    ]
}
```

### Beispielantwort

```
{
   "TransferId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
}
```

### Beispiel

Im folgenden Beispiel wird eine Dateiübertragung vom lokalen AWS Speicher auf einen Remote-SFTP-Server gestartet.

### Beispielanforderung

```
{
    "ConnectorId": "c-01234567890abcdef",
    "SendFilePaths": [
        "/DOC-EXAMPLE-BUCKET/myfile-1.txt",
        "/DOC-EXAMPLE-BUCKET/myfile-2.txt",
        "/DOC-EXAMPLE-BUCKET/myfile-3.txt"
    ],
    "RemoteDirectoryPath": "/MySFTPRootFolder/fromTranferFamilyServer"
}
```

Beispielantwort

```
{
   "TransferId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222"
}
```

Beispiel

Im folgenden Beispiel wird eine Dateiübertragung von einem Remote-SFTP-Server zum lokalen AWS Speicher gestartet.

Beispielanforderung

```
{
    "ConnectorId": "c-111122223333AAAAA",
    "RetrieveFilePaths": [
        "/MySFTPFolder/toTranferFamily/myfile-1.txt",
        "/MySFTPFolder/toTranferFamily/myfile-2.txt",
        "/MySFTPFolder/toTranferFamily/myfile-3.txt"
    ],
    "LocalDirectoryPath": "/DOC-EXAMPLE-BUCKET/mySourceFiles"
}
```

Beispielantwort

```
{
   "TransferId": "a1b2c3d4-5678-90ab-cdef-EXAMPLEaaaaa"
}
```

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2

- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# StartServer

Ändert den Status eines Servers, für den das File Transfer Protocol aktiviert ist, von zu. `OFFLINE` `ONLINE` Dies hat keine Auswirkungen auf einen Server, der dies bereits getan hat. `ONLINE` Ein `ONLINE` Server kann Dateiübertragungsaufträge annehmen und verarbeiten.

Der Status von `STARTING` gibt an, dass sich der Server in einem Zwischenzustand befindet, entweder nicht vollständig antworten kann oder nicht vollständig online ist. Die Werte von `START_FAILED` können auf einen Fehler hinweisen.

Auf diesen Anruf wurde keine Antwort zurückgegeben.

## Anforderungssyntax

```
{
   "ServerId": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

### ServerId

Ein vom System zugewiesener eindeutiger Bezeichner für einen Server, den Sie starten.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Erforderlich: Ja

## Antwortelemente

Wenn die Aktion erfolgreich ist, gibt der Dienst eine HTTP 200-Antwort mit leerem HTTP-Textinhalt zurück.

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

ThrottlingException

Die Anforderung wurde aufgrund der Drosselung von Anforderungen abgelehnt.

HTTP Status Code: 400

## Beispiele

Beispiel

Im folgenden Beispiel wird ein Server gestartet.

Beispielanforderung

```
{
```

```
   "ServerId": "s-01234567890abcdef"
}
```

Beispiel

Dies ist eine Beispielantwort für diesen API-Aufruf.

Beispielantwort

```
{
   "ServerId": "s-01234567890abcdef"
}
```

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# StopServer

Ändert den Status eines Servers, für den das File Transfer Protocol aktiviert ist, von zu. `ONLINE` `OFFLINE` Ein `OFFLINE` Server kann keine Dateiübertragungsaufträge annehmen und verarbeiten. Informationen, die an Ihren Server gebunden sind, wie Server- und Benutzereigenschaften, werden durch das Stoppen Ihres Servers nicht beeinträchtigt.

> **Note**
>
> Das Stoppen des Servers hat keine Auswirkungen auf die Abrechnung Ihrer Dateiübertragungsprotokoll-Endpunkte. Sie müssen den Server löschen, damit Ihnen keine Rechnungen mehr berechnet werden.

Der Status von `STOPPING` gibt an, dass sich der Server in einem Zwischenzustand befindet und entweder nicht vollständig antworten kann oder nicht vollständig offline ist. Die Werte von `STOP_FAILED` können auf einen Fehler hinweisen.

Auf diesen Anruf wurde keine Antwort zurückgegeben.

## Anforderungssyntax

```
{
   "ServerId": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

### ServerId

Ein vom System zugewiesener eindeutiger Bezeichner für einen Server, den Sie gestoppt haben.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Erforderlich: Ja

## Antwortelemente

Wenn die Aktion erfolgreich ist, gibt der Dienst eine HTTP 200-Antwort mit leerem HTTP-Textinhalt zurück.

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

ThrottlingException

Die Anforderung wurde aufgrund der Drosselung von Anforderungen abgelehnt.

HTTP Status Code: 400

## Beispiele

### Beispiel

Im folgenden Beispiel wird ein Server gestoppt.

Beispielanforderung

```
{
   "ServerId": "s-01234567890abcdef"
}
```

### Beispiel

Dies ist eine Beispielantwort für diesen API-Aufruf.

Beispielantwort

```
{
   "ServerId": "s-01234567890abcdef"
}
```

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# TagResource

Hängt ein Schlüssel-Wert-Paar an eine Ressource an, die durch ihren Amazon-Ressourcennamen (ARN) identifiziert wird. Ressourcen sind Benutzer, Server, Rollen und andere Entitäten.

Auf diesen Anruf wurde keine Antwort zurückgegeben.

## Anforderungssyntax

```
{
   "Arn": "string",
   "Tags": [
      {
         "Key": "string",
         "Value": "string"
      }
   ]
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

### Arn

Ein Amazon-Ressourcenname (ARN) für eine bestimmte AWS Ressource, z. B. einen Server, einen Benutzer oder eine Rolle.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 1600 Zeichen.

Pattern: `arn:\S+`

Erforderlich: Ja

### Tags

ARNs zugewiesene Schlüssel-Wert-Paare, die Sie verwenden können, um Ressourcen nach Typ zu gruppieren und nach ihnen zu suchen. Sie können diese Metadaten für jeden Zweck an Ressourcen (Server, Benutzer, Workflows usw.) anhängen.

Typ: Array von Tag-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Die maximale Anzahl beträgt 50 Elemente.

Erforderlich: Ja

## Antwortelemente

Wenn die Aktion erfolgreich ist, gibt der Dienst eine HTTP 200-Antwort mit leerem HTTP-Textinhalt zurück.

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Beispiele

### Beispiel

Im folgenden Beispiel wird einem Server, für den das Dateiübertragungsprotokoll aktiviert ist, ein Tag hinzugefügt.

Beispielanforderung

```
{
   "Arn": "arn:aws:transfer:us-east-1:176354371281:server/s-01234567890abcdef",
   "Tags": [
      {
         "Key": "Group",
         "Value": "Europe"
      }
   ]
}
```

### Beispiel

Dieses Beispiel veranschaulicht eine Verwendung von. TagResource

Beispielantwort

```
         HTTP 200 response with an empty HTTP body.
```

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2

- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# TestConnection

Testet, ob Ihr SFTP-Connector erfolgreich eingerichtet wurde. Es wird dringend empfohlen, diesen Vorgang aufzurufen, um zu testen, ob Sie Dateien zwischen dem lokalen AWS Speicher und dem SFTP-Server eines Handelspartners übertragen können.

## Anforderungssyntax

```
{
   "ConnectorId": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

ConnectorId

Die eindeutige Kennung für den Konnektor.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `c-([0-9a-f]{17})`

Erforderlich: Ja

## Antwortsyntax

```
{
   "ConnectorId": "string",
   "Status": "string",
   "StatusMessage": "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.

ConnectorId

Gibt den Bezeichner des Connector-Objekts zurück, das Sie testen.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `c-([0-9a-f]{17})`

Status

Wird zurückgegeben`OK`, wenn der Test erfolgreich war oder `ERROR` wenn der Test fehlschlägt.

Typ: Zeichenfolge

StatusMessage

Gibt zurück`Connection succeeded`, ob der Test erfolgreich war. Oder gibt eine beschreibende Fehlermeldung zurück, wenn der Test fehlschlägt. Die folgende Liste enthält je nach der Fehlermeldung, die Sie erhalten, Einzelheiten zur Problembehandlung.

- Vergewissern Sie sich, dass Ihr geheimer Name mit dem Namen in den Berechtigungen für die Übertragung von Rollen übereinstimmt.
- Überprüfen Sie die Server-URL in der Connector-Konfiguration und stellen Sie sicher, dass die Anmeldeinformationen außerhalb des Connectors erfolgreich funktionieren.
- Stellen Sie sicher, dass das Geheimnis existiert und richtig formatiert ist.
- Stellen Sie sicher, dass der vertrauenswürdige Hostschlüssel in der Connectorkonfiguration mit der `ssh-keyscan` Ausgabe übereinstimmt.

Typ: Zeichenfolge

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Beispiele

### Beispiel

Im folgenden Beispiel wird die Verbindung zu einem Remoteserver getestet.

```
aws transfer test-connection --connector-id c-abcd1234567890fff
```

### Beispielantwort

Bei Erfolg gibt der API-Aufruf die folgenden Details zurück.

```
{

    "Status": "OK",

    "StatusMessage": "Connection succeeded"

}
```

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# TestIdentityProvider

Wenn der `IdentityProviderType` eines Servers, für den das File Transfer Protocol aktiviert ist`API_Gateway`, `AWS_DIRECTORY_SERVICE` oder ist, wird getestet, ob Ihr Identitätsanbieter erfolgreich eingerichtet wurde. Es wird dringend empfohlen, dass Sie diesen Vorgang aufrufen, um Ihre Authentifizierungsmethode zu testen, sobald Sie Ihren Server erstellt haben. Auf diese Weise können Sie Probleme mit der Integration des Identitätsanbieters beheben, um sicherzustellen, dass Ihre Benutzer den Dienst erfolgreich nutzen können.

Die Parameter `ServerId` und `UserName` müssen angegeben werden. Die `ServerProtocolSourceIp`, und `UserPassword` sind alle optional.

Beachten Sie Folgendes:

- Sie können nicht verwenden`TestIdentityProvider`, wenn `IdentityProviderType` der Ihres Servers ist`SERVICE_MANAGED`.
- `TestIdentityProvider`funktioniert nicht mit Schlüsseln: Es akzeptiert nur Passwörter.
- `TestIdentityProvider`kann den Passwortvorgang für einen benutzerdefinierten Identity Provider testen, der Schlüssel und Passwörter verarbeitet.
- Wenn Sie falsche Werte für Parameter angeben, ist das `Response` Feld leer.
- Wenn Sie eine Server-ID für einen Server angeben, der vom Dienst verwaltete Benutzer verwendet, wird eine Fehlermeldung angezeigt:

  ```
  An error occurred (InvalidRequestException) when calling the
  TestIdentityProvider operation: s-server-ID not configured for external
  auth
  ```

- Wenn Sie eine Server-ID für den `--server-id` Parameter eingeben, die keinen tatsächlichen Transferserver identifiziert, wird die folgende Fehlermeldung angezeigt:

  ```
  An error occurred (ResourceNotFoundException) when calling the
  TestIdentityProvider operation: Unknown server.
  ```

  Es ist möglich, dass sich Ihr Server in einer anderen Region befindet. Sie können eine Region angeben, indem Sie Folgendes hinzufügen:`--region region-code`, `--region us-east-2` um beispielsweise einen Server in USA Ost (Ohio) anzugeben.

## Anforderungssyntax

```
{
   "ServerId": "string",
   "ServerProtocol": "string",
   "SourceIp": "string",
   "UserName": "string",
   "UserPassword": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

ServerId

Ein vom System zugewiesener Bezeichner für einen bestimmten Server. Die Benutzerauthentifizierungsmethode dieses Servers wird mit einem Benutzernamen und einem Passwort getestet.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Erforderlich: Ja

ServerProtocol

Der Typ des zu testenden Dateiübertragungsprotokolls.

Die verfügbaren Protokolle sind:

- Secure Shell (SSH) File Transfer Protocol (SFTP)
- Sicheres Dateiübertragungsprotokoll (FTPS)
- Dateiübertragungsprotokoll (FTP)
- Erklärung zur Anwendbarkeit 2 (AS2)

Typ: Zeichenfolge

Zulässige Werte: `SFTP | FTP | FTPS | AS2`

Erforderlich: Nein

SourceIp

Die Quell-IP-Adresse des zu testenden Kontos.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 32 Zeichen.

Pattern: `\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3}`

Erforderlich: Nein

UserName

Der Name des Kontos, das getestet werden soll.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 3. Maximale Länge beträgt 100 Zeichen.

Pattern: `[\w][\w@.-]{2,99}`

Erforderlich: Ja

UserPassword

Das Passwort des Kontos, das getestet werden soll.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 1024 Zeichen.

Erforderlich: Nein

## Antwortsyntax

```
{
   "Message": "string",
```

```
   "Response": "string",
   "StatusCode": number,
   "Url": "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.

### Message

Eine Meldung, die angibt, ob der Test erfolgreich war oder nicht.

> **Note**
>
> Wenn eine leere Zeichenfolge zurückgegeben wird, ist die wahrscheinlichste Ursache, dass die Authentifizierung aufgrund eines falschen Benutzernamens oder Passworts fehlgeschlagen ist.

Typ: Zeichenfolge

### Response

Die Antwort, die von Ihrem API Gateway oder Ihrer Lambda-Funktion zurückgegeben wird.

Typ: Zeichenfolge

### StatusCode

Der HTTP-Statuscode, der die Antwort von Ihrem API Gateway oder Ihrer Lambda-Funktion ist.

Typ: Ganzzahl

### Url

Der Endpunkt des Dienstes, der zur Authentifizierung eines Benutzers verwendet wird.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 255 Zeichen.

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Beispiele

Beispiel

Die folgende Anforderung gibt eine Meldung von einem Identitätsanbieter zurück, dass eine Kombination aus Benutzername und Kennwort eine gültige Identität darstellt, mit der Sie verwendet werden können AWS Transfer Family.

Beispielanforderung

```
{
   "ServerID": "s-01234567890abcdef",
   "UserName": "my_user",
   "UserPassword": "MyPassword-1"
```

```
}
```

Beispiel

Die folgende Antwort zeigt eine Beispielantwort für einen erfolgreichen Test.

Beispielantwort

```
      "Response":"
      {\"homeDirectory\":\"/mybucket001\",\"homeDirectoryDetails\":null,
\"homeDirectoryType\":\"PATH\",\"posixProfile\":null,
      \"publicKeys\":\"[ssh-rsa-key]\",\"role\":\"arn:aws:iam::123456789012:role/
my_role\",\"policy\":null,\"username\":\"transferuser002\",
      \"identityProviderType\":null,\"userConfigMessage\":null)"}
      "StatusCode": "200",
      "Message": ""
```

Beispiel

Die folgende Antwort gibt an, dass der angegebene Benutzer zu mehr als einer Gruppe gehört, die Zugriff hat.

```
         "Response":"",
         "StatusCode":200,
         "Message":"More than one associated access found for user's groups."
```

Beispiel

Wenn Sie mithilfe eines API Gateway einen benutzerdefinierten Identitätsanbieter erstellt und konfiguriert haben, können Sie den folgenden Befehl eingeben, um Ihren Benutzer zu testen:

`aws transfer test-identity-provider --server-id` *`s-0123456789abcdefg`* `--user-name` *`myuser`*

wobei s-0123456789abcdefg Ihr Transferserver und myuser der Benutzername für Ihren benutzerdefinierten Benutzer ist.

Wenn der Befehl erfolgreich ist, ähnelt Ihre Antwort der folgenden, wobei:

- AWS-Konto Die ID ist 012345678901
- Die Benutzerrolle ist user-role-api-gateway
- Das Home-Verzeichnis ist myuser-bucket
- Der öffentliche Schlüssel ist der öffentliche Schlüssel
- Aufruf-URL ist Aufruf-URL

```
{
    "Response": "{\"Role\": \"arn:aws:iam::012345678901:role/user-role-api-gateway\",
\"HomeDirectory\": \"/myuser-bucket\",\"PublicKeys\": \"[public-key]\"}",
    "StatusCode": 200,
    "Message": "",
    "Url": "https://invocation-URL/servers/s-0123456789abcdefg/users/myuser/config"
}
```

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# UntagResource

Trennt ein Schlüssel-Wert-Paar von einer Ressource, die durch ihren Amazon-Ressourcennamen (ARN) identifiziert wird. Ressourcen sind Benutzer, Server, Rollen und andere Entitäten.

Auf diesen Anruf wurde keine Antwort zurückgegeben.

## Anforderungssyntax

```
{
   "Arn": "string",
   "TagKeys": [ "string" ]
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

Arn

Der Wert der Ressource, für die das Tag entfernt werden soll. Ein Amazon-Ressourcenname (ARN) ist eine Kennung für eine bestimmte AWS Ressource, z. B. einen Server, einen Benutzer oder eine Rolle.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 1600 Zeichen.

Pattern: `arn:\S+`

Erforderlich: Ja

TagKeys

TagKeys sind ARNs zugewiesene Schlüssel-Wert-Paare, die verwendet werden können, um Ressourcen nach Typ zu gruppieren und nach ihnen zu suchen. Diese Metadaten können für jeden Zweck an Ressourcen angehängt werden.

Typ: Zeichenfolgen-Array

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Die maximale Anzahl beträgt 50 Elemente.

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 128 Zeichen.

Erforderlich: Ja

## Antwortelemente

Wenn die Aktion erfolgreich ist, gibt der Dienst eine HTTP 200-Antwort mit leerem HTTP-Textinhalt zurück.

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

## Beispiele

### Beispiel

Im folgenden Beispiel wird ein Tag eines Servers entfernt, für den das Dateiübertragungsprotokoll aktiviert ist.

Beispielanforderung

```
{
   "Arn": "arn:aws:transfer:us-east-1:176354371281:server/s-01234567890abcdef",
   "TagKeys": "Europe" ]
}
```

### Beispiel

Dieses Beispiel veranschaulicht eine Verwendung von. UntagResource

Beispielantwort

```
        HTTP 200 response with an empty HTTP body.
```

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python

- AWS SDK for Ruby V3

# UpdateAccess

Ermöglicht das Aktualisieren von Parametern für den in den `ExternalID` Parametern `ServerID` und angegebenen Zugriff.

## Anforderungssyntax

```
{
   "ExternalId": "string",
   "HomeDirectory": "string",
   "HomeDirectoryMappings": [
      {
         "Entry": "string",
         "Target": "string",
         "Type": "string"
      }
   ],
   "HomeDirectoryType": "string",
   "Policy": "string",
   "PosixProfile": {
      "Gid": number,
      "SecondaryGids": [ number ],
      "Uid": number
   },
   "Role": "string",
   "ServerId": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

### ExternalId

Eine eindeutige Kennung, die zur Identifizierung bestimmter Gruppen in Ihrem Verzeichnis erforderlich ist. Die Benutzer der Gruppe, die Sie zuordnen, haben über die aktivierten Protokolle Zugriff auf Ihre Amazon S3- oder Amazon EFS-Ressourcen AWS Transfer Family. Wenn Sie den Gruppennamen kennen, können Sie die SID-Werte anzeigen, indem Sie den folgenden Befehl unter Windows ausführen PowerShell.

```
Get-ADGroup -Filter {samAccountName -like "YourGroupName*"} -Properties
* | Select SamAccountName,ObjectSid
```

Ersetzen Sie diesen Befehl YourGroupNamedurch den Namen Ihrer Active Directory-Gruppe.

Der reguläre Ausdruck, der zur Überprüfung dieses Parameters verwendet wird, ist eine Zeichenfolge, die aus alphanumerischen Groß- und Kleinbuchstaben ohne Leerzeichen besteht. Sie können auch Unterstriche oder eines der folgenden Zeichen verwenden: =, . @: /-

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 256 Zeichen.

Pattern: `S-1-[\d-]+`

Erforderlich: Ja

HomeDirectory

Das Zielverzeichnis (Ordner) für einen Benutzer bei der Serveranmeldung mithilfe des Client.

Ein Beispiel für `HomeDirectory` ist `/bucket_name/home/mydirectory`.

> **Note**
>
> Der `HomeDirectory`-Parameter wird nur verwendet, wenn `HomeDirectoryType` auf `PATH` gesetzt ist.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 1024 Zeichen.

Pattern: `(|/.*)`

Erforderlich: Nein

HomeDirectoryMappings

Logische Verzeichniszuordnungen, die angeben, welche Amazon S3- oder Amazon EFS-Pfade und -Schlüssel für Ihren Benutzer sichtbar sein sollen und wie Sie sie sichtbar machen möchten. Sie müssen das `Entry` `Target` Und-Paar angeben, das `Entry` zeigt, wie der Pfad sichtbar

gemacht `Target` wird und der tatsächliche Amazon S3- oder Amazon EFS-Pfad ist. Wenn Sie nur ein Ziel angeben, wird es unverändert angezeigt. Sie müssen außerdem sicherstellen, dass Ihre AWS Identity and Access Management (IAM-) Rolle Zugriff auf Pfade in `Target` bietet. Dieser Wert kann nur gesetzt werden, wenn er auf LOGICAL gesetzt `HomeDirectoryType` ist.

Das Folgende ist ein `Entry` Beispiel für ein `Target` Und-Paar.

```
[ { "Entry": "/directory1", "Target": "/bucket_name/home/
mydirectory" } ]
```

In den meisten Fällen können Sie diesen Wert anstelle der Sitzungsrichtlinie verwenden, um Ihren Benutzer auf das angegebene Home-Verzeichnis (“ `chroot` „) zu sperren. Dazu können Sie `Entry` auf `/` und auf den `HomeDirectory` Parameterwert setzen`Target`.

Im Folgenden finden Sie ein `Entry` Beispiel für ein `Target` Und-Paar`chroot`.

```
[ { "Entry": "/", "Target": "/bucket_name/home/mydirectory" } ]
```

Typ: Array von HomeDirectoryMapEntry-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Maximale Anzahl von 50000 Artikeln.

Erforderlich: Nein

HomeDirectoryType

Die Art des Zielverzeichnisses (Ordners), das das Home-Verzeichnis Ihrer Benutzer sein soll, wenn sie sich beim Server anmelden. Wenn Sie es auf einstellen`PATH`, sieht der Benutzer den absoluten Amazon S3-Bucket- oder Amazon EFS-Pfad so, wie er in seinen File Transfer Protocol-Clients ist. Wenn Sie es auf einstellen`LOGICAL`, müssen Sie Zuordnungen dafür angeben, wie Sie Amazon S3- oder Amazon EFS-Pfade `HomeDirectoryMappings` für Ihre Benutzer sichtbar machen möchten.

> **Note**
>
> Wenn `HomeDirectoryType` ja`LOGICAL`, müssen Sie mithilfe des Parameters Zuordnungen angeben. `HomeDirectoryMappings` Ist dies hingegen der Fall, `HomeDirectoryType` geben Sie mithilfe des Parameters einen absoluten Pfad an`HomeDirectory`. `PATH` Sie können nicht beides `HomeDirectory` und `HomeDirectoryMappings` in Ihrer Vorlage haben.

Typ: Zeichenfolge

Zulässige Werte: `PATH | LOGICAL`

Erforderlich: Nein

Policy

Eine Sitzungsrichtlinie für Ihren Benutzer, sodass Sie dieselbe AWS Identity and Access Management (IAM-) Rolle für mehrere Benutzer verwenden können. Diese Richtlinie beschränkt den Zugriff eines Benutzers auf Teile seines Amazon S3 S3-Buckets. Variablen, die Sie in dieser Richtlinie verwenden können: `${Transfer:UserName}`, `${Transfer:HomeDirectory}` und `${Transfer:HomeBucket}`.

> **Note**
>
> Diese Richtlinie gilt nur, wenn die Domain von Amazon S3 `ServerId` ist. Amazon EFS verwendet keine Sitzungsrichtlinien.
> AWS Transfer Family Speichert für Sitzungsrichtlinien die Richtlinie als JSON-Blob und nicht als Amazon-Ressourcennamen (ARN) der Richtlinie. Speichern Sie die Richtlinie als ein JSON-Blob und geben Sie sie in das `Policy`-Argument ein.
> Ein Beispiel für eine Sitzungsrichtlinien finden Sie unter Sitzungsrichtlinien.
> Weitere Informationen finden Sie AssumeRolein der AWS Security Token Service API-Referenz.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 2048 Zeichen.

Erforderlich: Nein

PosixProfile

Die vollständige POSIX-Identität, einschließlich Benutzer-ID (`Uid`), Gruppen-ID (`Gid`) und sekundärer Gruppen-IDs (`SecondaryGids`), die den Zugriff Ihrer Benutzer auf Ihre Amazon EFS-Dateisysteme (Elastic File System) steuert. Die POSIX-Berechtigungen, die für Dateien und Verzeichnisse in Ihrem Dateisystem festgelegt sind, bestimmen die Zugriffsebene, die Ihre Benutzer beim Übertragen von Dateien in und aus Ihren Amazon EFS-Dateisystemen erhalten.

Typ: PosixProfile Objekt

Erforderlich: Nein

Role

Der Amazon-Ressourcenname (ARN) der Rolle AWS Identity and Access Management (IAM), die den Zugriff Ihrer Benutzer auf Ihren Amazon S3-Bucket oder Ihr Amazon EFS-Dateisystem steuert. Die mit dieser Rolle verbundenen Richtlinien bestimmen die Zugriffsebene, die Sie Ihren Benutzern beim Übertragen von Dateien in und aus Ihrem Amazon-S3-Bucket oder Amazon-EFS-Dateisystem bereitstellen möchten. Die IAM-Rolle sollte außerdem eine Vertrauensstellung enthalten, mit der der Server Zugriff auf Ihre Ressourcen erhält, wenn er die Übertragungsanfragen Ihres Benutzers bearbeitet.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 2048 Zeichen.

Pattern: `arn:.*role/\S+`

Erforderlich: Nein

ServerId

Eine vom System zugewiesene eindeutige ID für eine Server-Instance. Dies ist der spezifische Server, dem Sie Ihren Benutzer hinzugefügt haben.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Erforderlich: Ja

## Antwortsyntax

```
{
   "ExternalId": "string",
   "ServerId": "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.

ExternalId

Die externe Kennung der Gruppe, deren Benutzer über die aktivierten Protokolle mithilfe von AWS Transfer Family Zugriff auf Ihre Amazon S3- oder Amazon EFS-Ressourcen haben.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 256 Zeichen.

Pattern: `S-1-[\d-]+`

ServerId

Die Kennung des Servers, mit dem der Benutzer verbunden ist.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceExistsException

Die angeforderte Ressource ist nicht vorhanden oder befindet sich in einer anderen Region als der für den Befehl angegebenen.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

ThrottlingException

Die Anforderung wurde aufgrund der Drosselung von Anforderungen abgelehnt.

HTTP Status Code: 400

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# UpdateAgreement

Aktualisiert einige Parameter für eine bestehende Vereinbarung. Geben Sie die `AgreementId` und die `ServerId` für die Vereinbarung an, die Sie aktualisieren möchten, zusammen mit den neuen Werten für die zu aktualisierenden Parameter.

## Anforderungssyntax

```
{
   "AccessRole": "string",
   "AgreementId": "string",
   "BaseDirectory": "string",
   "Description": "string",
   "LocalProfileId": "string",
   "PartnerProfileId": "string",
   "ServerId": "string",
   "Status": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

### AccessRole

Konnektoren werden verwendet, um Dateien entweder über das AS2- oder das SFTP-Protokoll zu senden. Geben Sie für die Zugriffsrolle den Amazon-Ressourcennamen (ARN) der zu AWS Identity and Access Management verwendenden Rolle an.

Für AS2-Konnektoren

Mit AS2 senden Sie Dateien, indem Sie `StartFileTransfer` aufrufen und die Dateipfade im Anforderungsparameter `SendFilePaths` angeben. Mit dem übergeordneten Verzeichnis der Datei (Beispiel: das übergeordnetes Verzeichnis für `--send-file-paths /bucket/dir/file.txt` ist `/bucket/dir/`) speichern wir eine verarbeitete AS2-Nachrichtendatei vorübergehend, speichern die MDN, wenn wir sie vom Partner erhalten, und schreiben eine endgültige JSON-Datei, die relevante Metadaten der Übertragung enthält. Daher

muss `AccessRole` Lese- und Schreibzugriff auf das übergeordnete Verzeichnis des in der `StartFileTransfer`-Anforderung verwendeten Dateispeicherorts gewähren. Darüber hinaus müssen Sie Lese- und Schreibzugriff für das übergeordnete Verzeichnis der Dateien gewähren, die Sie mit `StartFileTransfer` senden möchten.

Wenn Sie die Standardauthentifizierung für Ihren AS2-Connector verwenden, erfordert die Zugriffsrolle die `secretsmanager:GetSecretValue` Erlaubnis für den geheimen Schlüssel. Wenn das Geheimnis mit einem vom Kunden verwalteten Schlüssel anstelle des AWS verwalteten Schlüssels in Secrets Manager verschlüsselt wird, benötigt die Rolle auch die `kms:Decrypt` Erlaubnis für diesen Schlüssel.

Für SFTP-Konnektoren

Stellen Sie sicher, dass die Zugriffsrolle Lese- und Schreibzugriff auf das übergeordnete Verzeichnis des Dateispeicherorts bietet, der in der `StartFileTransfer` Anfrage verwendet wird. Stellen Sie außerdem sicher, dass die Rolle die `secretsmanager:GetSecretValue` Berechtigung dazu bietet AWS Secrets Manager.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 2048 Zeichen.

Pattern: `arn:.*role/\S+`

Erforderlich: Nein

## AgreementId

Eine eindeutige Kennung für die Vereinbarung. Diese Kennung wird zurückgegeben, wenn Sie eine Vereinbarung erstellen.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `a-([0-9a-f]{17})`

Erforderlich: Ja

## BaseDirectory

Um das Zielverzeichnis (Ordner) für übertragene Dateien zu ändern, geben Sie den Bucket-Ordner an, den Sie verwenden möchten, z. B. */DOC-EXAMPLE-BUCKET/home/mydirectory* .

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 1024 Zeichen.

Pattern: `(|/.*)`

Erforderlich: Nein

### Description

Um die bestehende Beschreibung zu ersetzen, geben Sie eine kurze Beschreibung der Vereinbarung ein.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 1. Höchstlänge = 200 Zeichen.

Pattern: `[\p{Graph}]+`

Erforderlich: Nein

### LocalProfileId

Eine eindeutige Kennung für das lokale AS2-Profil.

Um die lokale Profil-ID zu ändern, geben Sie hier einen neuen Wert ein.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `p-([0-9a-f]{17})`

Erforderlich: Nein

### PartnerProfileId

Eine eindeutige Kennung für das Partnerprofil. Um die Partnerprofil-ID zu ändern, geben Sie hier einen neuen Wert ein.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `p-([0-9a-f]{17})`

Erforderlich: Nein

ServerId

Eine vom System zugewiesene eindeutige ID für eine Server-Instance. Dies ist der spezifische Server, den die Vereinbarung verwendet.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Erforderlich: Ja

Status

Sie können den Status der Vereinbarung aktualisieren, indem Sie entweder eine inaktive Vereinbarung aktivieren oder umgekehrt.

Typ: Zeichenfolge

Zulässige Werte: `ACTIVE | INACTIVE`

Erforderlich: Nein

## Antwortsyntax

```
{
   "AgreementId": "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.

AgreementId

Eine eindeutige Kennung für die Vereinbarung. Diese Kennung wird zurückgegeben, wenn Sie eine Vereinbarung erstellen.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `a-([0-9a-f]{17})`

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceExistsException

Die angeforderte Ressource ist nicht vorhanden oder befindet sich in einer anderen Region als der für den Befehl angegebenen.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

ThrottlingException

Die Anforderung wurde aufgrund der Drosselung von Anforderungen abgelehnt.

HTTP Status Code: 400

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# UpdateCertificate

Aktualisiert die aktiven und inaktiven Daten für ein Zertifikat.

## Anforderungssyntax

```
{
   "ActiveDate": number,
   "CertificateId": "string",
   "Description": "string",
   "InactiveDate": number
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

ActiveDate

Ein optionales Datum, das angibt, wann das Zertifikat aktiv wird.

Typ: Zeitstempel

Erforderlich: Nein

CertificateId

Die ID des Zertifikatsobjekts, das Sie aktualisieren.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 22.

Pattern: `cert-([0-9a-f]{17})`

Erforderlich: Ja

Description

Eine kurze Beschreibung zur leichteren Identifizierung des Zertifikats.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 1. Höchstlänge = 200 Zeichen.

Pattern: `[\p{Graph}]+`

Erforderlich: Nein

InactiveDate

Ein optionales Datum, das angibt, wann das Zertifikat inaktiv wird.

Typ: Zeitstempel

Erforderlich: Nein

## Antwortsyntax

```
{
   "CertificateId": "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.

CertificateId

Gibt den Bezeichner des Zertifikatsobjekts zurück, das Sie aktualisieren.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 22.

Pattern: `cert-([0-9a-f]{17})`

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

ThrottlingException

Die Anforderung wurde aufgrund der Drosselung von Anforderungen abgelehnt.

HTTP Status Code: 400

## Beispiele

Beispiel

Im folgenden Beispiel wird das aktive Datum für ein Zertifikat aktualisiert und das aktive Datum auf den 16. Januar 2022 um 16:12:07 UTC — 5 Stunden festgelegt.

Beispielanforderung

```
aws transfer update-certificate --certificate-id c-abcdefg123456hijk --active-date
 2022-01-16T16:12:07-05:00
```

Beispiel

Im Folgenden finden Sie eine Beispielantwort für diesen API-Aufruf.

Beispielantwort

```
"CertificateId": "c-abcdefg123456hijk"
```

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# UpdateConnector

Aktualisiert einige Parameter für einen vorhandenen Konnektor. Geben Sie die Werte `ConnectorId` für den Connector an, den Sie aktualisieren möchten, zusammen mit den neuen Werten für die zu aktualisierenden Parameter.

## Anforderungssyntax

```
{
   "AccessRole": "string",
   "As2Config": {
      "BasicAuthSecretId": "string",
      "Compression": "string",
      "EncryptionAlgorithm": "string",
      "LocalProfileId": "string",
      "MdnResponse": "string",
      "MdnSigningAlgorithm": "string",
      "MessageSubject": "string",
      "PartnerProfileId": "string",
      "SigningAlgorithm": "string"
   },
   "ConnectorId": "string",
   "LoggingRole": "string",
   "SecurityPolicyName": "string",
   "SftpConfig": {
      "TrustedHostKeys": [ "string" ],
      "UserSecretId": "string"
   },
   "Url": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

## AccessRole

Konnektoren werden verwendet, um Dateien entweder über das AS2- oder das SFTP-Protokoll zu senden. Geben Sie für die Zugriffsrolle den Amazon-Ressourcennamen (ARN) der zu AWS Identity and Access Management verwendenden Rolle an.

Für AS2-Konnektoren

Mit AS2 senden Sie Dateien, indem Sie `StartFileTransfer` aufrufen und die Dateipfade im Anforderungsparameter `SendFilePaths` angeben. Mit dem übergeordneten Verzeichnis der Datei (Beispiel: das übergeordnetes Verzeichnis für `--send-file-paths /bucket/dir/file.txt` ist `/bucket/dir/`) speichern wir eine verarbeitete AS2-Nachrichtendatei vorübergehend, speichern die MDN, wenn wir sie vom Partner erhalten, und schreiben eine endgültige JSON-Datei, die relevante Metadaten der Übertragung enthält. Daher muss `AccessRole` Lese- und Schreibzugriff auf das übergeordnete Verzeichnis des in der `StartFileTransfer`-Anforderung verwendeten Dateispeicherorts gewähren. Darüber hinaus müssen Sie Lese- und Schreibzugriff für das übergeordnete Verzeichnis der Dateien gewähren, die Sie mit `StartFileTransfer` senden möchten.

Wenn Sie die Standardauthentifizierung für Ihren AS2-Connector verwenden, erfordert die Zugriffsrolle die `secretsmanager:GetSecretValue` Erlaubnis für den geheimen Schlüssel. Wenn das Geheimnis mit einem vom Kunden verwalteten Schlüssel anstelle des AWS verwalteten Schlüssels in Secrets Manager verschlüsselt wird, benötigt die Rolle auch die `kms:Decrypt` Erlaubnis für diesen Schlüssel.

Für SFTP-Konnektoren

Stellen Sie sicher, dass die Zugriffsrolle Lese- und Schreibzugriff auf das übergeordnete Verzeichnis des Dateispeicherorts bietet, der in der `StartFileTransfer` Anfrage verwendet wird. Stellen Sie außerdem sicher, dass die Rolle die `secretsmanager:GetSecretValue` Berechtigung dazu bietet AWS Secrets Manager.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 2048 Zeichen.

Pattern: `arn:.*role/\S+`

Erforderlich: Nein

As2Config

Eine Struktur, die die Parameter für ein AS2-Connector-Objekt enthält.

Typ: As2ConnectorConfig Objekt

Erforderlich: Nein

ConnectorId

Der eindeutige Bezeichner für den Konnektor.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `c-([0-9a-f]{17})`

Erforderlich: Ja

LoggingRole

Der Amazon-Ressourcenname (ARN) der Rolle AWS Identity and Access Management (IAM), der es einem Connector ermöglicht, die CloudWatch Protokollierung für Amazon S3-Ereignisse zu aktivieren. Wenn diese Option aktiviert ist, können Sie die Connector-Aktivität in Ihren CloudWatch Protokollen einsehen.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 2048 Zeichen.

Pattern: `arn:.*role/\S+`

Erforderlich: Nein

SecurityPolicyName

Gibt den Namen der Sicherheitsrichtlinie für den Connector an.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 100 Zeichen.

Pattern: `TransferSFTPConnectorSecurityPolicy-[A-Za-z0-9-]+`

Erforderlich: Nein

SftpConfig

Eine Struktur, die die Parameter für ein SFTP-Connector-Objekt enthält.

Typ: SftpConnectorConfig Objekt

Erforderlich: Nein

Url

Die URL des AS2- oder SFTP-Endpunkts des Partners.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 255 Zeichen.

Erforderlich: Nein

## Antwortsyntax

```
{
   "ConnectorId": "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.

ConnectorId

Gibt den Bezeichner des Connector-Objekts zurück, das Sie aktualisieren.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `c-([0-9a-f]{17})`

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceExistsException

Die angeforderte Ressource ist nicht vorhanden oder befindet sich in einer anderen Region als der für den Befehl angegebenen.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

ThrottlingException

Die Anforderung wurde aufgrund der Drosselung von Anforderungen abgelehnt.

HTTP Status Code: 400

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET

- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# UpdateHostKey

Aktualisiert die Beschreibung für den Host-Schlüssel, der durch die `HostKeyId` Parameter `ServerId` und angegeben ist.

## Anforderungssyntax

```
{
   "Description": "string",
   "HostKeyId": "string",
   "ServerId": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

Description

Eine aktualisierte Beschreibung für den Hostschlüssel.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Höchstlänge = 200 Zeichen.

Pattern: `[\p{Print}]*`

Erforderlich: Ja

HostKeyId

Die ID des Hostschlüssels, den Sie aktualisieren.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 25.

Pattern: `hostkey-[0-9a-f]{17}`

Erforderlich: Ja

ServerId

Die ID des Servers, der den Hostschlüssel enthält, den Sie aktualisieren.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Erforderlich: Ja

## Antwortsyntax

```
{
   "HostKeyId": "string",
   "ServerId": "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.

HostKeyId

Gibt die Hostschlüssel-ID für den aktualisierten Hostschlüssel zurück.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 25.

Pattern: `hostkey-[0-9a-f]{17}`

ServerId

Gibt die Server-ID für den Server zurück, der den aktualisierten Hostschlüssel enthält.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

ThrottlingException

Die Anforderung wurde aufgrund der Drosselung von Anforderungen abgelehnt.

HTTP Status Code: 400

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle

- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# UpdateProfile

Aktualisiert einige Parameter für ein vorhandenes Profil. Geben Sie die `ProfileId` für das Profil, das Sie aktualisieren möchten, zusammen mit den neuen Werten für die zu aktualisierenden Parameter an.

## Anforderungssyntax

```
{
   "CertificateIds": [ "string" ],
   "ProfileId": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

### CertificateIds

Ein Array von Kennungen für die importierten Zertifikate. Sie verwenden diese Kennung für die Arbeit mit Profilen und Partnerprofilen.

Typ: Zeichenfolgen-Array

Längenbeschränkungen: Feste Länge von 22.

Pattern: `cert-([0-9a-f]{17})`

Erforderlich: Nein

### ProfileId

Die ID des Profilobjekts, das Sie aktualisieren.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `p-([0-9a-f]{17})`

Erforderlich: Ja

## Antwortsyntax

```
{
   "ProfileId": "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.

ProfileId

Gibt den Bezeichner für das Profil zurück, das aktualisiert wird.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `p-([0-9a-f]{17})`

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

ThrottlingException

Die Anforderung wurde aufgrund der Drosselung von Anforderungen abgelehnt.

HTTP Status Code: 400

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# UpdateServer

Aktualisiert die Eigenschaften des Servers, für den das Dateiübertragungsprotokoll aktiviert ist, nachdem dieser Server erstellt wurde.

Der `UpdateServer` Aufruf gibt den des Servers zurück, `ServerId` den Sie aktualisiert haben.

## Anforderungssyntax

```
{
   "Certificate": "string",
   "EndpointDetails": {
      "AddressAllocationIds": [ "string" ],
      "SecurityGroupIds": [ "string" ],
      "SubnetIds": [ "string" ],
      "VpcEndpointId": "string",
      "VpcId": "string"
   },
   "EndpointType": "string",
   "HostKey": "string",
   "IdentityProviderDetails": {
      "DirectoryId": "string",
      "Function": "string",
      "InvocationRole": "string",
      "SftpAuthenticationMethods": "string",
      "Url": "string"
   },
   "LoggingRole": "string",
   "PostAuthenticationLoginBanner": "string",
   "PreAuthenticationLoginBanner": "string",
   "ProtocolDetails": {
      "As2Transports": [ "string" ],
      "PassiveIp": "string",
      "SetStatOption": "string",
      "TlsSessionResumptionMode": "string"
   },
   "Protocols": [ "string" ],
   "S3StorageOptions": {
      "DirectoryListingOptimization": "string"
   },
   "SecurityPolicyName": "string",
   "ServerId": "string",
   "StructuredLogDestinations": [ "string" ],
```

```
   "WorkflowDetails": {
      "OnPartialUpload": [
         {
            "ExecutionRole": "string",
            "WorkflowId": "string"
         }
      ],
      "OnUpload": [
         {
            "ExecutionRole": "string",
            "WorkflowId": "string"
         }
      ]
   }
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

Certificate

Der Amazon-Ressourcenname (ARN) des AWS Certificate Manager (ACM) -Zertifikats. Erforderlich, wenn `Protocols` auf `FTPS` eingestellt ist.

Informationen zum Anfordern eines neuen öffentlichen Zertifikats finden Sie unter Anfordern eines öffentlichen Zertifikats im AWS Certificate Manager Manager-Benutzerhandbuch.

Informationen zum Importieren eines vorhandenen Zertifikats in ACM finden Sie unter Zertifikate in ACM importieren im AWS Certificate Manager Manager-Benutzerhandbuch.

Informationen zum Anfordern eines privaten Zertifikats für die Verwendung von FTPS über private IP-Adressen finden Sie unter Anfordern eines privaten Zertifikats im AWS Certificate Manager Manager-Benutzerhandbuch.

Zertifikate mit den folgenden kryptografischen Algorithmen und Schlüsselgrößen werden unterstützt:

- 2048-Bit-RSA (RSA_2048)

- 4096-Bit-RSA (RSA_4096)
- Elliptic Prime Curve 256-Bit (EC_prime256v1)
- Elliptic Prime Curve 384-Bit (EC_secp384r1)
- Elliptic Prime Curve 521-Bit (EC_secp521r1)

> **Note**
>
> Das Zertifikat muss ein gültiges SSL/TLS X.509 Version 3-Zertifikat mit FQDN oder IP-Adresse und Informationen über den Aussteller sein.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 1600 Zeichen.

Erforderlich: Nein

EndpointDetails

Die Virtual Private Cloud (VPC)-Endpunkt-Einstellungen, die für Ihren Server konfiguriert sind. Wenn Sie Ihren Endpunkt in Ihrer VPC hosten, können Sie ihn nur für Ressourcen innerhalb Ihrer VPC zugänglich machen, oder Sie können elastische IP-Adressen anfügen und ihn für Clients über das Internet zugänglich machen. Die Standard-Sicherheitsgruppen Ihrer VPC werden Ihrem Endpunkt automatisch zugewiesen.

Typ: EndpointDetails Objekt

Erforderlich: Nein

EndpointType

Der Typ des Endpunkts, den Ihr Server verwenden soll. Sie können den Endpunkt Ihres Servers öffentlich zugänglich machen (PUBLIC) oder ihn in Ihrer VPC hosten. Mit einem Endpunkt, der in einer VPC gehostet wird, können Sie den Zugriff auf Ihren Server und Ressourcen nur innerhalb Ihrer VPC einschränken oder sich für das Internet entscheiden, indem Sie Elastic-IP-Adressen direkt an ihn anfügen.

> **Note**
>
> Nach dem 19. Mai 2021 können Sie mit `EndpointType=VPC_ENDPOINT` Ihrem Konto keinen Server mehr erstellen, wenn Ihr AWS Konto dies nicht bereits vor dem

19. Mai 2021 getan hat. Wenn du am oder vor dem 19. Mai 2021 bereits Server mit `EndpointType=VPC_ENDPOINT` deinem AWS Konto erstellt hast, bist du davon nicht betroffen. Verwenden Sie nach diesem Datum `EndpointType =VPC`.
Weitere Informationen finden Sie unter Einstellung der Verwendung von VPC_ENDPOINT.
Es wird empfohlen, dass Sie `VPC` als `EndpointType` verwenden. Bei diesem Endpunkttyp haben Sie die Möglichkeit, bis zu drei Elastic IPv4-Adressen (inklusive BYO IP) direkt mit dem Endpunkt Ihres Servers zu verknüpfen und VPC-Sicherheitsgruppen zu verwenden, um den Datenverkehr durch die öffentliche IP-Adresse des Clients zu beschränken. Dies ist nicht möglich, wenn `EndpointType` auf `VPC_ENDPOINT` gesetzt ist.

Typ: Zeichenfolge

Zulässige Werte: `PUBLIC | VPC | VPC_ENDPOINT`

Erforderlich: Nein

HostKey

Der private RSA-, ECDSA- oder ED25519-Schlüssel, der für Ihren SFTP-fähigen Server verwendet werden soll. Sie können mehrere Hostschlüssel hinzufügen, falls Sie Schlüssel rotieren möchten oder über einen Satz aktiver Schlüssel verfügen, die unterschiedliche Algorithmen verwenden.

Verwenden Sie den folgenden Befehl, um einen RSA 2048-Bit-Schlüssel ohne Passphrase zu generieren:

```
ssh-keygen -t rsa -b 2048 -N "" -m PEM -f my-new-server-key.
```

Verwenden Sie einen Mindestwert von 2048 für die Option. `-b` Sie können einen stärkeren Schlüssel erstellen, indem Sie 3072 oder 4096 verwenden.

Verwenden Sie den folgenden Befehl, um einen 256-Bit-ECDSA-Schlüssel ohne Passphrase zu generieren:

```
ssh-keygen -t ecdsa -b 256 -N "" -m PEM -f my-new-server-key.
```

Gültige Werte für die `-b` Option für ECDSA sind 256, 384 und 521.

Verwenden Sie den folgenden Befehl, um einen ED25519-Schlüssel ohne Passphrase zu generieren:

```
ssh-keygen -t ed25519 -N "" -f my-new-server-key.
```

Alle diese Befehle können Sie durch eine Zeichenfolge Ihrer Wahl my-new-server-keyersetzen.

> **Important**
>
> Wenn Sie nicht vorhaben, bestehende Benutzer von einem vorhandenen SFTP-fähigen Server auf einen neuen Server zu migrieren, aktualisieren Sie den Hostschlüssel nicht. Das versehentliche Ändern des Host-Schlüssels eines Servers kann zu Unterbrechungen führen.

Weitere Informationen finden Sie im Benutzerhandbuch unter Hostschlüssel für Ihren SFTP-fähigen Server aktualisieren. AWS Transfer Family

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge von 4096.

Erforderlich: Nein

IdentityProviderDetails

Ein Array, das alle Informationen enthält, die zum Aufrufen der API-Authentifizierungsmethode eines Kunden erforderlich sind.

Typ: IdentityProviderDetails Objekt

Erforderlich: Nein

LoggingRole

Der Amazon-Ressourcenname (ARN) der AWS Identity and Access Management (IAM) -Rolle, der es einem Server ermöglicht, die CloudWatch Amazon-Protokollierung für Amazon S3 oder Amazon EFSEvents zu aktivieren. Wenn diese Option aktiviert ist, können Sie Benutzeraktivitäten in Ihren Protokollen einsehen. CloudWatch

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 2048 Zeichen.

Pattern: `(|arn:.*role/\S+)`

Erforderlich: Nein

PostAuthenticationLoginBanner

Gibt eine Zeichenfolge an, die angezeigt wird, wenn Benutzer sich mit einem Server verbinden. Diese Zeichenfolge wird nach Authentifizierung des Benutzers angezeigt.

> **Note**
>
> Das SFTP-Protokoll unterstützt keine Anzeige-Banner nach der Authentifizierung.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge von 4096.

Pattern: `[\x09-\x0D\x20-\x7E]*`

Erforderlich: Nein

PreAuthenticationLoginBanner

Gibt eine Zeichenfolge an, die angezeigt wird, wenn Benutzer sich mit einem Server verbinden. Diese Zeichenfolge wird angezeigt, bevor sich der Benutzer authentifiziert. Das folgende Banner zeigt beispielsweise Details zur Verwendung des Systems an:

```
This system is for the use of authorized users only. Individuals using
this computer system without authority, or in excess of their authority,
are subject to having all of their activities on this system monitored
and recorded by system personnel.
```

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge von 4096.

Pattern: `[\x09-\x0D\x20-\x7E]*`

Erforderlich: Nein

### ProtocolDetails

Protokolleinstellungen, die für Ihren Server konfiguriert sind.

- Verwenden Sie den Parameter `PassiveIp` zur Angabe des passiven Modus (für FTP- und FTPS-Protokolle). Geben Sie eine einzelne gepunktete IPv4-Adresse ein, z. B. die externe IP-Adresse einer Firewall, eines Routers oder eines Load Balancers.
- Verwenden Sie den Parameter `SetStatOption`, um den Fehler zu ignorieren, der generiert wird, wenn der Client versucht, den Befehl `SETSTAT` für eine Datei zu verwenden, die Sie in einen Amazon-S3-Bucket hochladen. Damit der AWS Transfer Family Server den `SETSTAT` Befehl ignoriert und Dateien hochlädt, ohne Änderungen an Ihrem SFTP-Client vornehmen zu müssen, setzen Sie den Wert auf. `ENABLE_NO_OP` Wenn Sie den `SetStatOption` Parameter auf setzen`ENABLE_NO_OP`, generiert Transfer Family einen Protokolleintrag in Amazon CloudWatch Logs, sodass Sie feststellen können, wann der Client einen `SETSTAT` Anruf tätigt.
- Verwenden Sie den `TlsSessionResumptionMode` Parameter, um festzustellen, ob Ihr AWS Transfer Family Server die letzten, ausgehandelten Sitzungen über eine eindeutige Sitzungs-ID wieder aufnimmt.
- `As2Transports` gibt an, wie AS2-Nachrichten transportiert werden sollen. Derzeit wird nur HTTP unterstützt.

Typ: ProtocolDetails Objekt

Erforderlich: Nein

### Protocols

Gibt das/die Dateiübertragungsprotokoll(e) an, über das/die der Dateiübertragungsprotokoll-Client eine Verbindung zum Endpunkt des Servers herstellen kann. Die verfügbaren Protokolle sind:

- `SFTP` (Secure Shell (SSH) File Transfer Protocol): Dateiübertragung über SSH
- `FTPS` (File Transfer Protocol Secure): Dateiübertragung mit TLS-Verschlüsselung
- `FTP` (File Transfer Protocol): Unverschlüsselte Dateiübertragung
- `AS2`(Anwendbarkeitserklärung 2): Wird für den Transport strukturierter Daten verwendet business-to-business

> **Note**
>
> - Wenn Sie wählen`FTPS`, müssen Sie ein in AWS Certificate Manager (ACM) gespeichertes Zertifikat wählen, das zur Identifizierung Ihres Servers verwendet wird, wenn Clients über FTPS eine Verbindung zu ihm herstellen.

- Wenn `Protocol` entweder `FTP` oder `FTPS` enthält, muss `EndpointType` `VPC` lauten und `IdentityProviderType` muss `AWS_DIRECTORY_SERVICE`, `AWS_LAMBDA` oder `API_GATEWAY` lauten.
- Wenn `Protocol` `FTP` enthält, können `AddressAllocationIds` nicht zugeordnet werden.
- Wenn `Protocol` nur auf `SFTP` gesetzt ist, kann `EndpointType` auf `PUBLIC` gesetzt werden und `IdentityProviderType` kann auf einen der unterstützten Identitätstypen gesetzt werden: `SERVICE_MANAGED`, `AWS_DIRECTORY_SERVICE`, `AWS_LAMBDA` oder `API_GATEWAY`.
- Wenn das `Protocol` `AS2` enthält, dann muss der `EndpointType` `VPC` lauten und die Domain muss Amazon S3 sein.

Typ: Zeichenfolgen-Array

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Maximale Anzahl von 4 Elementen.

Zulässige Werte: `SFTP | FTP | FTPS | AS2`

Erforderlich: Nein

S3StorageOptions

Gibt an, ob die Leistung für Ihre Amazon S3 S3-Verzeichnisse optimiert ist oder nicht. Diese ist standardmäßig deaktiviert.

Standardmäßig haben Zuordnungen von Home-Verzeichnissen einen `TYPE` Wert von. `DIRECTORY` Wenn Sie diese Option aktivieren, müssten Sie den Wert dann explizit auf setzen `HomeDirectoryMapEntryType`, `FILE` wenn eine Zuordnung ein Dateiziel haben soll.

Typ: S3StorageOptions Objekt

Erforderlich: Nein

SecurityPolicyName

Gibt den Namen der Sicherheitsrichtlinie für den Server an.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 100 Zeichen.

Pattern: `Transfer[A-Za-z0-9]*SecurityPolicy-[A-Za-z0-9-]+`

Erforderlich: Nein

ServerId

Eine vom System zugewiesene eindeutige Kennung für eine Serverinstanz, der der Transfer Family Family-Benutzer zugewiesen ist.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Erforderlich: Ja

StructuredLogDestinations

Gibt die Protokollgruppen an, an die Ihre Serverprotokolle gesendet werden.

Um eine Protokollgruppe anzugeben, müssen Sie den ARN für eine bestehende Protokollgruppe angeben. In diesem Fall lautet das Format der Protokollgruppe wie folgt:

`arn:aws:logs:region-name:amazon-account-id:log-group:log-group-name:*`

Beispiel: `arn:aws:logs:us-east-1:111122223333:log-group:mytestgroup:*`

Wenn Sie zuvor eine Protokollgruppe für einen Server angegeben haben, können Sie diese löschen und somit die strukturierte Protokollierung deaktivieren, indem Sie in einem `update-server` Aufruf einen leeren Wert für diesen Parameter angeben. Beispielsweise:

```
update-server --server-id s-1234567890abcdef0 --structured-log-destinations
```

Typ: Zeichenfolge-Array

Array-Mitglieder: Die Mindestanzahl beträgt 0 Elemente. Die maximale Anzahl beträgt 1 Element.

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 1600 Zeichen.

Pattern: `arn:\S+`

Erforderlich: Nein

WorkflowDetails

Gibt die Workflow-ID für den zuzuweisenden Workflow und die für die Ausführung des Workflows verwendete Ausführungsrolle an.

Zusätzlich zu einem Workflow, der ausgeführt wird, wenn eine Datei vollständig hochgeladen wurde, kann `WorkflowDetails` auch eine Workflow-ID (und Ausführungsrolle) für einen Workflow enthalten, der beim teilweisen Upload ausgeführt werden soll. Ein teilweiser Upload erfolgt, wenn die Serversitzung unterbrochen wird, während die Datei noch hochgeladen wird.

Um einen zugeordneten Workflow von einem Server zu entfernen, können Sie einen leeren `OnUpload`-Datentyp bereitstellen, wie im folgenden Beispiel.

```
aws transfer update-server --server-id s-01234567890abcdef --workflow-details '{"OnUpload":[]}'
```

Typ: WorkflowDetails Objekt

Erforderlich: Nein

## Antwortsyntax

```
{
   "ServerId": "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.

ServerId

Eine vom System zugewiesene eindeutige Kennung für einen Server, dem der Transfer Family Family-Benutzer zugewiesen ist.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

AccessDeniedException

Sie haben keinen ausreichenden Zugriff zum Durchführen dieser Aktion.

HTTP Status Code: 400

ConflictException

Diese Ausnahme wird ausgelöst, wenn der für einen Server mit aktiviertem Dateiübertragungsprotokoll aufgerufen `UpdateServer` wird, dessen Endpunkttyp VPC ist und sich der Server nicht im verfügbaren `VpcEndpointID` Status befindet.

HTTP Status Code: 400

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im Dienst ein Fehler auftritt. AWS Transfer Family

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceExistsException

Die angeforderte Ressource ist nicht vorhanden oder befindet sich in einer anderen Region als der für den Befehl angegebenen.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

ThrottlingException

Die Anforderung wurde aufgrund der Drosselung von Anforderungen abgelehnt.

HTTP Status Code: 400

## Beispiele

### Beispiel

Im folgenden Beispiel wird die Rolle eines Servers aktualisiert.

Beispielanforderung

```
{
   "EndpointDetails": {
   "VpcEndpointId": "vpce-01234f056f3g13",
   "LoggingRole": "CloudWatchS3Events",
   "ServerId": "s-01234567890abcdef"
   }
}
```

### Beispiel

Im folgenden Beispiel werden alle zugehörigen Workflows vom Server entfernt.

Beispielanforderung

```
aws transfer update-server --server-id s-01234567890abcdef --workflow-details
 '{"OnUpload":[]}'
```

### Beispiel

Dies ist eine Beispielantwort für diesen API-Aufruf.

Beispielantwort

```
{
   "ServerId": "s-01234567890abcdef"
}
```

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

# UpdateUser

Weist einem Benutzer neue Eigenschaften zu. Parameter, die Sie übergeben, ändern einige oder alle der folgenden Elemente: das Basisverzeichnis, die Rolle und die Richtlinie für die `UserName` und, die `ServerId` Sie angeben.

Die Antwort gibt das `ServerId` und das `UserName` für den aktualisierten Benutzer zurück.

In der Konsole können Sie Eingeschränkt auswählen, wenn Sie einen Benutzer erstellen oder aktualisieren. Dadurch wird sichergestellt, dass der Benutzer auf nichts außerhalb seines Home-Verzeichnisses zugreifen kann. Die programmatische Methode zur Konfiguration dieses Verhaltens besteht darin, den Benutzer zu aktualisieren. Setzen Sie sie `HomeDirectoryType` auf `LOGICAL` und geben Sie sie `HomeDirectoryMappings` mit `Entry` als root (`/`) und `Target` als ihr Heimatverzeichnis an.

Wenn das Home-Verzeichnis des Benutzers beispielsweise lautet`/test/admin-user`, aktualisiert der folgende Befehl den Benutzer so, dass in seiner Konfiguration in der Konsole das Kennzeichen Eingeschränkt als ausgewählt angezeigt wird.

```
aws transfer update-user --server-id <server-id> --user-name admin-user --
home-directory-type LOGICAL --home-directory-mappings "[{\"Entry\":\"/\",
\"Target\":\"/test/admin-user\"}]"
```

## Anforderungssyntax

```
{
   "HomeDirectory": "string",
   "HomeDirectoryMappings": [
      {
         "Entry": "string",
         "Target": "string",
         "Type": "string"
      }
   ],
   "HomeDirectoryType": "string",
   "Policy": "string",
   "PosixProfile": {
      "Gid": number,
      "SecondaryGids": [ number ],
      "Uid": number
   },
```

```
    "Role": "string",
    "ServerId": "string",
    "UserName": "string"
}
```

## Anforderungsparameter

Informationen zu den Parametern, die alle Aktionen gemeinsam haben, finden Sie unter Allgemeine Parameter.

Die Anforderung akzeptiert die folgenden Daten im JSON-Format.

HomeDirectory

Das Zielverzeichnis (Ordner) für einen Benutzer bei der Serveranmeldung mithilfe des Client.

Ein Beispiel für `HomeDirectory` ist `/bucket_name/home/mydirectory`.

> **Note**
>
> Der `HomeDirectory`-Parameter wird nur verwendet, wenn `HomeDirectoryType` auf `PATH` gesetzt ist.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 1024 Zeichen.

Pattern: `(|/.*)`

Erforderlich: Nein

HomeDirectoryMappings

Logische Verzeichniszuordnungen, die angeben, welche Amazon S3- oder Amazon EFS-Pfade und -Schlüssel für Ihren Benutzer sichtbar sein sollen und wie Sie sie sichtbar machen möchten. Sie müssen das `Entry` `Target` Und-Paar angeben, das `Entry` zeigt, wie der Pfad sichtbar gemacht `Target` wird und der tatsächliche Amazon S3- oder Amazon EFS-Pfad ist. Wenn Sie nur ein Ziel angeben, wird es unverändert angezeigt. Sie müssen außerdem sicherstellen, dass Ihre AWS Identity and Access Management (IAM-) Rolle Zugriff auf Pfade in `Target` bietet. Dieser Wert kann nur gesetzt werden, wenn er auf LOGICAL gesetzt `HomeDirectoryType` ist.

Das Folgende ist ein `Entry` Beispiel für ein `Target` Und-Paar.

```
[ { "Entry": "/directory1", "Target": "/bucket_name/home/
mydirectory" } ]
```

In den meisten Fällen können Sie diesen Wert anstelle der Sitzungsrichtlinie verwenden, um Ihren Benutzer auf das angegebene Home-Verzeichnis (“ `chroot` „) zu sperren. Dazu können Sie den Wert `Entry` auf '/' und dann `Target` auf den HomeDirectory Parameterwert setzen.

Im Folgenden finden Sie ein `Entry` `Target` Und-Pair-Beispiel für`chroot`.

```
[ { "Entry": "/", "Target": "/bucket_name/home/mydirectory" } ]
```

Typ: Array von HomeDirectoryMapEntry-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Maximale Anzahl von 50000 Artikeln.

Erforderlich: Nein

HomeDirectoryType

Die Art des Zielverzeichnisses (Ordners), das das Home-Verzeichnis Ihrer Benutzer sein soll, wenn sie sich beim Server anmelden. Wenn Sie es auf einstellen`PATH`, sieht der Benutzer den absoluten Amazon S3-Bucket- oder Amazon EFS-Pfad so, wie er in seinen File Transfer Protocol-Clients ist. Wenn Sie es auf einstellen`LOGICAL`, müssen Sie Zuordnungen dafür angeben, wie Sie Amazon S3- oder Amazon EFS-Pfade `HomeDirectoryMappings` für Ihre Benutzer sichtbar machen möchten.

> **Note**
>
> `HomeDirectoryType`Ist dies der `LOGICAL` Fall, müssen Sie mithilfe des Parameters Zuordnungen angeben. `HomeDirectoryMappings` Ist dies hingegen der Fall, `HomeDirectoryType` geben Sie mithilfe des Parameters einen absoluten Pfad an`HomeDirectory`. `PATH` Sie können nicht beides `HomeDirectory` und `HomeDirectoryMappings` in Ihrer Vorlage haben.

Typ: Zeichenfolge

Zulässige Werte: `PATH | LOGICAL`

Erforderlich: Nein

Policy

Eine Sitzungsrichtlinie für Ihren Benutzer, sodass Sie dieselbe AWS Identity and Access Management (IAM-) Rolle für mehrere Benutzer verwenden können. Diese Richtlinie beschränkt den Zugriff eines Benutzers auf Teile seines Amazon S3 S3-Buckets. Variablen, die Sie in dieser Richtlinie verwenden können: `${Transfer:UserName}`, `${Transfer:HomeDirectory}` und `${Transfer:HomeBucket}`.

> **Note**
>
> Diese Richtlinie gilt nur, wenn die Domain von Amazon S3 `ServerId` ist. Amazon EFS verwendet keine Sitzungsrichtlinien.
> AWS Transfer Family Speichert für Sitzungsrichtlinien die Richtlinie als JSON-Blob und nicht als Amazon-Ressourcennamen (ARN) der Richtlinie. Speichern Sie die Richtlinie als ein JSON-Blob und geben Sie sie in das `Policy`-Argument ein.
> Ein Beispiel für eine Sitzungsrichtlinien finden Sie unter Sitzungsrichtlinien.
> Weitere Informationen finden Sie AssumeRolein der AWS Security Token Service API-Referenz.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 2048 Zeichen.

Erforderlich: Nein

PosixProfile

Gibt die vollständige POSIX-Identität an, einschließlich Benutzer-ID (`Uid`), Gruppen-ID (`Gid`) und aller sekundären Gruppen-IDs (`SecondaryGids`), die den Zugriff Ihrer Benutzer auf Ihre Amazon Elastic File Systems (Amazon EFS) steuert. Die POSIX-Berechtigungen, die für Dateien und Verzeichnisse in Ihrem Dateisystem festgelegt sind, bestimmen die Zugriffsebene, die Ihre Benutzer beim Übertragen von Dateien in und aus Ihren Amazon EFS-Dateisystemen erhalten.

Typ: PosixProfile Objekt

Erforderlich: Nein

Role

Der Amazon-Ressourcenname (ARN) der Rolle AWS Identity and Access Management (IAM), die den Zugriff Ihrer Benutzer auf Ihren Amazon S3-Bucket oder Ihr Amazon EFS-

Dateisystem steuert. Die mit dieser Rolle verbundenen Richtlinien bestimmen die Zugriffsebene, die Sie Ihren Benutzern beim Übertragen von Dateien in und aus Ihrem Amazon-S3-Bucket oder Amazon-EFS-Dateisystem bereitstellen möchten. Die IAM-Rolle sollte außerdem eine Vertrauensstellung enthalten, mit der der Server Zugriff auf Ihre Ressourcen erhält, wenn er die Übertragungsanfragen Ihres Benutzers bearbeitet.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 2048 Zeichen.

Pattern: `arn:.*role/\S+`

Erforderlich: Nein

## ServerId

Eine vom System zugewiesene eindeutige Kennung für eine Transfer Family Family-Serverinstanz, der der Benutzer zugewiesen ist.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Erforderlich: Ja

## UserName

Eine eindeutige Zeichenfolge, die einen Benutzer identifiziert und einem Server zugeordnet ist, wie durch die `ServerId` festgelegt. Dieser Benutzername muss mindestens 3 und maximal 100 Zeichen enthalten. Folgende Zeichen sind gültig: a-z, A-Z, 0-9, Unterstrich '_', Bindestrich '-', Punkt '.'und beim Zeichen '@'. Der Benutzername kann nicht mit einem Bindestrich, einem Punkt oder einem At beginnen.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 3. Maximale Länge beträgt 100 Zeichen.

Pattern: `[\w][\w@.-]{2,99}`

Erforderlich: Ja

## Antwortsyntax

```
{
   "ServerId": "string",
   "UserName": "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.

ServerId

Eine vom System zugewiesene eindeutige Kennung für eine Transfer Family Family-Serverinstanz, der das Konto zugewiesen ist.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

UserName

Der eindeutige Bezeichner für einen Benutzer, der einer Serverinstanz zugewiesen ist, die in der Anfrage angegeben wurde.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 3. Maximale Länge beträgt 100 Zeichen.

Pattern: `[\w][\w@.-]{2,99}`

## Fehler

Weitere Informationen zu den allgemeinen Fehlern, die bei allen Aktionen zurückgegeben werden, finden Sie unter Häufige Fehler.

InternalServiceError

Diese Ausnahme wird ausgelöst, wenn im AWS Transfer Family Dienst ein Fehler auftritt.

HTTP Status Code: 500

InvalidRequestException

Diese Ausnahme wird ausgelöst, wenn der Client eine falsch formatierte Anfrage sendet.

HTTP Status Code: 400

ResourceNotFoundException

Diese Ausnahme wird ausgelöst, wenn eine Ressource vom AWS Transfer Family Family-Dienst nicht gefunden wird.

HTTP Status Code: 400

ServiceUnavailableException

Die Anfrage ist fehlgeschlagen, da der AWS Transfer Family Family-Dienst nicht verfügbar ist.

HTTP Status Code: 500

ThrottlingException

Die Anforderung wurde aufgrund der Drosselung von Anforderungen abgelehnt.

HTTP Status Code: 400

## Beispiele

### Beispiel

Im folgenden Beispiel wird ein Transfer Family Family-Benutzer aktualisiert.

Beispielanforderung

```
{
   "HomeDirectory": "/bucket2/documentation",
   "HomeDirectoryMappings": [
      {
         "Entry": "/directory1",
         "Target": "/bucket_name/home/mydirectory"
      }
   ],
   "HomeDirectoryType:" "PATH",
   "Role": "AssumeRole",
```

```
   "ServerId": "s-01234567890abcdef",
   "UserName": "my_user"
}
```

Beispiel

Dies ist eine Beispielantwort für diesen API-Aufruf.

Beispielantwort

```
{
   "ServerId": "s-01234567890abcdef",
   "UserName": "my_user"
}
```

### Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS -Befehlszeilenschnittstelle
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go v2
- AWS SDK for Java V2
- AWS SDK für JavaScript V3
- AWS SDK for PHP V3
- AWS SDK für Python
- AWS SDK for Ruby V3

## Datentypen

Die folgenden Datentypen werden unterstützt:

- As2ConnectorConfig
- CopyStepDetails
- CustomStepDetails

- DecryptStepDetails
- DeleteStepDetails
- DescribedAccess
- DescribedAgreement
- DescribedCertificate
- DescribedConnector
- DescribedExecution
- DescribedHostKey
- DescribedProfile
- DescribedSecurityPolicy
- DescribedServer
- DescribedUser
- DescribedWorkflow
- EfsFileLocation
- EndpointDetails
- ExecutionError
- ExecutionResults
- ExecutionStepResult
- FileLocation
- HomeDirectoryMapEntry
- IdentityProviderDetails
- InputFileLocation
- ListedAccess
- ListedAgreement
- ListedCertificate
- ListedConnector
- ListedExecution
- ListedHostKey
- ListedProfile
- ListedServer

- ListedUser
- ListedWorkflow
- LoggingConfiguration
- PosixProfile
- ProtocolDetails
- S3FileLocation
- S3InputFileLocation
- S3StorageOptions
- S3Tag
- ServiceMetadata
- SftpConnectorConfig
- SshPublicKey
- Tag
- TagStepDetails
- UserDetails
- WorkflowDetail
- WorkflowDetails
- WorkflowStep

# As2ConnectorConfig

Enthält die Details für ein AS2-Connector-Objekt. Das Connector-Objekt wird für ausgehende AS2-Prozesse verwendet, um den AWS Transfer Family Kunden mit dem Geschäftspartner zu verbinden.

## Inhalt

BasicAuthSecretId

Bietet Unterstützung für die Standardauthentifizierung für die AS2 Connectors-API. Um die Standardauthentifizierung zu verwenden, müssen Sie den Namen oder den Amazon-Ressourcennamen (ARN) eines Geheimnisses in angeben AWS Secrets Manager.

Der Standardwert für diesen Parameter ist`null`, was darauf hinweist, dass die Standardauthentifizierung für den Connector nicht aktiviert ist.

Wenn der Connector die Standardauthentifizierung verwenden soll, muss das Geheimnis das folgende Format haben:

```
{ "Username": "user-name", "Password": "user-password" }
```

Ersetzen Sie `user-name` und `user-password` durch die Anmeldeinformationen für den tatsächlichen Benutzer, der authentifiziert wird.

Beachten Sie Folgendes:

- Sie speichern diese Anmeldeinformationen in Secrets Manager und geben sie nicht direkt an diese API weiter.
- Wenn Sie die API, SDKs oder CloudFormation zur Konfiguration Ihres Connectors verwenden, müssen Sie das Geheimnis erstellen, bevor Sie die Standardauthentifizierung aktivieren können. Wenn Sie jedoch die AWS Managementkonsole verwenden, können Sie das System das Geheimnis für Sie erstellen lassen.

Wenn Sie zuvor die Standardauthentifizierung für einen Connector aktiviert haben, können Sie sie mithilfe des `UpdateConnector` API-Aufrufs deaktivieren. Wenn Sie beispielsweise die CLI verwenden, können Sie den folgenden Befehl ausführen, um die Standardauthentifizierung zu entfernen:

```
update-connector --connector-id my-connector-id --as2-config
'BasicAuthSecretId=""'
```

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 2048 Zeichen.

Erforderlich: Nein

Compression

Gibt an, ob die AS2-Datei komprimiert ist.

Typ: Zeichenfolge

Zulässige Werte: `ZLIB | DISABLED`

Erforderlich: Nein

EncryptionAlgorithm

Der Algorithmus, der zum Verschlüsseln der Datei verwendet wird.

Beachten Sie Folgendes:

- Verwenden Sie den `DES_EDE3_CBC` Algorithmus nur, wenn Sie einen Legacy-Client unterstützen müssen, der ihn benötigt, da es sich um einen schwachen Verschlüsselungsalgorithmus handelt.
- Sie können nur angeben`NONE`, ob die URL für Ihren Connector HTTPS verwendet. Durch die Verwendung von HTTPS wird sichergestellt, dass kein Datenverkehr im Klartext gesendet wird.

Typ: Zeichenfolge

Zulässige Werte: `AES128_CBC | AES192_CBC | AES256_CBC | DES_EDE3_CBC | NONE`

Erforderlich: Nein

LocalProfileId

Eine eindeutige Kennung für das lokale AS2-Profil.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `p-([0-9a-f]{17})`

Erforderlich: Nein

MdnResponse

Wird für ausgehende Anfragen (von einem AWS Transfer Family Server an einen AS2-Partnerserver) verwendet, um festzustellen, ob die Partnerantwort für Übertragungen synchron oder asynchron ist. Geben Sie einen der folgenden Werte an:

- `SYNC`: Das System erwartet eine synchrone MDN-Antwort, die bestätigt, dass die Datei erfolgreich übertragen wurde (oder nicht).
- `NONE`: Gibt an, dass keine MDN-Antwort erforderlich ist.

Typ: Zeichenfolge

Zulässige Werte: `SYNC | NONE`

Erforderlich: Nein

MdnSigningAlgorithm

Der Signaturalgorithmus für die MDN-Antwort.

> **Note**
>
> Wenn der Wert auf DEFAULT (oder überhaupt nicht gesetzt) gesetzt ist, `SigningAlgorithm` wird der Wert für verwendet.

Typ: Zeichenfolge

Zulässige Werte: `SHA256 | SHA384 | SHA512 | SHA1 | NONE | DEFAULT`

Erforderlich: Nein

MessageSubject

Wird als `Subject` HTTP-Header-Attribut in AS2-Nachrichten verwendet, die mit dem Connector gesendet werden.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 1. Maximale Länge beträgt 1024 Zeichen.

Pattern: `[\p{Print}\p{Blank}]+`

Erforderlich: Nein

PartnerProfileId

Eine eindeutige Kennung für das Partnerprofil für den Connector.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `p-([0-9a-f]{17})`

Erforderlich: Nein

SigningAlgorithm

Der Algorithmus, der verwendet wird, um die mit dem Connector gesendeten AS2-Nachrichten zu signieren.

Typ: Zeichenfolge

Zulässige Werte: `SHA256 | SHA384 | SHA512 | SHA1 | NONE`

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# CopyStepDetails

Jeder Schritttyp hat seine eigene `StepDetails` Struktur.

## Inhalt

DestinationFileLocation

Gibt den Speicherort für die kopierte Datei an. Verwenden Sie `${Transfer:UserName}` oder `${Transfer:UploadDate}` in diesem Feld, um das Zielpräfix nach Benutzername oder Upload-Datum zu parametrisieren.

- Legen Sie den Wert auf fest`DestinationFileLocation`, `${Transfer:UserName}` um hochgeladene Dateien in einen Amazon S3 S3-Bucket zu kopieren, dem der Name des Transfer Family Family-Benutzers vorangestellt wird, der die Datei hochgeladen hat.
- Legen Sie den Wert auf fest`DestinationFileLocation`, `${Transfer:UploadDate}` um hochgeladene Dateien in einen Amazon S3 S3-Bucket zu kopieren, dem das Datum des Uploads vorangestellt ist.

> **Note**
>
> Das System löst `UploadDate` das Datumsformat YYYY-MM-DD auf, das auf dem Datum basiert, an dem die Datei in UTC hochgeladen wird.

Typ: InputFileLocation Objekt

Erforderlich: Nein

Name

Der Name des Schritts, der als Kennung verwendet wird.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Die maximale Länge beträgt 30.

Pattern: `[\w-]*`

Erforderlich: Nein

OverwriteExisting

Ein Flag, das angibt, ob eine vorhandene Datei mit demselben Namen überschrieben werden. Der Standardwert ist `FALSE`.

Wenn der Workflow eine Datei verarbeitet, die denselben Namen wie eine vorhandene Datei hat, sieht das Verhalten wie folgt aus:

- `OverwriteExisting`Ist dies der `TRUE` Fall, wird die vorhandene Datei durch die Datei ersetzt, die gerade verarbeitet wird.
- Wenn `OverwriteExisting` ja`FALSE`, passiert nichts und die Workflow-Verarbeitung wird gestoppt.

Typ: Zeichenfolge

Zulässige Werte: `TRUE | FALSE`

Erforderlich: Nein

SourceFileLocation

Gibt an, welche Datei als Eingabe für den Workflow-Schritt verwendet werden soll: entweder die Ausgabe des vorherigen Schritts oder die ursprünglich hochgeladene Datei für den Workflow.

- Um die vorherige Datei als Eingabe zu verwenden, geben Sie ein`${previous.file}`. In diesem Fall verwendet dieser Workflow-Schritt die Ausgabedatei aus dem vorherigen Workflow-Schritt als Eingabe. Dies ist der Standardwert.
- Um den Speicherort der ursprünglich hochgeladenen Datei als Eingabe für diesen Schritt zu verwenden, geben Sie ein`${original.file}`.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 256 Zeichen.

Pattern: `\$\{(\w+.)+\w+\}`

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# CustomStepDetails

Jeder Schritttyp hat seine eigene `StepDetails` Struktur.

## Inhalt

Name

Der Name des Schritts, der als Kennung verwendet wird.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Die maximale Länge beträgt 30.

Pattern: `[\w-]*`

Erforderlich: Nein

SourceFileLocation

Gibt an, welche Datei als Eingabe für den Workflow-Schritt verwendet werden soll: entweder die Ausgabe des vorherigen Schritts oder die ursprünglich hochgeladene Datei für den Workflow.

- Um die vorherige Datei als Eingabe zu verwenden, geben Sie ein`${previous.file}`. In diesem Fall verwendet dieser Workflow-Schritt die Ausgabedatei aus dem vorherigen Workflow-Schritt als Eingabe. Dies ist der Standardwert.
- Um den Speicherort der ursprünglich hochgeladenen Datei als Eingabe für diesen Schritt zu verwenden, geben Sie ein`${original.file}`.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 256 Zeichen.

Pattern: `\$\{(\w+.)+\w+\}`

Erforderlich: Nein

Target

Der ARN für die Lambda-Funktion, die aufgerufen wird.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Höchstlänge = 170 Zeichen.

Pattern: `arn:[a-z-]+:lambda:.*`

Erforderlich: Nein

TimeoutSeconds

Timeout für den Schritt in Sekunden.

Typ: Ganzzahl

Gültiger Bereich: Mindestwert 1. Maximalwert von 1800.

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# DecryptStepDetails

Jeder Schritttyp hat seine eigene `StepDetails` Struktur.

## Inhalt

DestinationFileLocation

Gibt den Speicherort für die zu entschlüsselnde Datei an. Verwenden Sie `${Transfer:UserName}` oder `${Transfer:UploadDate}` in diesem Feld, um das Zielpräfix nach Benutzername oder Upload-Datum zu parametrisieren.

- Legen Sie den Wert von `DestinationFileLocation` to fest`${Transfer:UserName}`, um hochgeladene Dateien in einen Amazon S3 S3-Bucket zu entschlüsseln, dem der Name des Transfer Family Family-Benutzers vorangestellt ist, der die Datei hochgeladen hat.
- Legen Sie den Wert von `DestinationFileLocation` to fest`${Transfer:UploadDate}`, um hochgeladene Dateien in einen Amazon S3 S3-Bucket zu entschlüsseln, dem das Datum des Uploads vorangestellt ist.

> **Note**
>
> Das System löst `UploadDate` das Datumsformat YYYY-MM-DD auf, das auf dem Datum basiert, an dem die Datei in UTC hochgeladen wird.

Typ: InputFileLocation Objekt

Erforderlich: Ja

Type

Die Art der verwendeten Verschlüsselung. Derzeit muss dieser Wert sein`PGP`.

Typ: Zeichenfolge

Zulässige Werte: `PGP`

Erforderlich: Ja

Name

Der Name des Schritts, der als Bezeichner verwendet wird.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Die maximale Länge beträgt 30.

Pattern: `[\w-]*`

Erforderlich: Nein

OverwriteExisting

Ein Flag, das angibt, ob eine vorhandene Datei mit demselben Namen überschrieben werden. Der Standardwert ist `FALSE`.

Wenn der Workflow eine Datei verarbeitet, die denselben Namen wie eine vorhandene Datei hat, sieht das Verhalten wie folgt aus:

- `OverwriteExisting`Ist dies der `TRUE` Fall, wird die vorhandene Datei durch die Datei ersetzt, die gerade verarbeitet wird.
- `OverwriteExisting`Ist dies der Fall`FALSE`, passiert nichts und die Workflow-Verarbeitung wird gestoppt.

Typ: Zeichenfolge

Zulässige Werte: `TRUE | FALSE`

Erforderlich: Nein

SourceFileLocation

Gibt an, welche Datei als Eingabe für den Workflow-Schritt verwendet werden soll: entweder die Ausgabe des vorherigen Schritts oder die ursprünglich hochgeladene Datei für den Workflow.

- Um die vorherige Datei als Eingabe zu verwenden, geben Sie ein`${previous.file}`. In diesem Fall verwendet dieser Workflow-Schritt die Ausgabedatei aus dem vorherigen Workflow-Schritt als Eingabe. Dies ist der Standardwert.
- Um den Speicherort der ursprünglich hochgeladenen Datei als Eingabe für diesen Schritt zu verwenden, geben Sie ein`${original.file}`.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 256 Zeichen.

Pattern: `\$\{(\w+.)+\w+\}`

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# DeleteStepDetails

Der Name des Schritts, der zur Identifizierung des Löschschritts verwendet wird.

## Inhalt

Name

Der Name des Schritts, der als Kennung verwendet wird.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Die maximale Länge beträgt 30.

Pattern: `[\w-]*`

Erforderlich: Nein

SourceFileLocation

Gibt an, welche Datei als Eingabe für den Workflow-Schritt verwendet werden soll: entweder die Ausgabe des vorherigen Schritts oder die ursprünglich hochgeladene Datei für den Workflow.

- Um die vorherige Datei als Eingabe zu verwenden, geben Sie ein`${previous.file}`. In diesem Fall verwendet dieser Workflow-Schritt die Ausgabedatei aus dem vorherigen Workflow-Schritt als Eingabe. Dies ist der Standardwert.
- Um den Speicherort der ursprünglich hochgeladenen Datei als Eingabe für diesen Schritt zu verwenden, geben Sie ein`${original.file}`.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 256 Zeichen.

Pattern: `\$\{(\w+.)+\w+\}`

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++

- AWS SDK for Java V2
- AWS SDK for Ruby V3

# DescribedAccess

Beschreibt die Eigenschaften des angegebenen Zugriffs.

## Inhalt

ExternalId

Eine eindeutige Kennung, die zur Identifizierung bestimmter Gruppen in Ihrem Verzeichnis erforderlich ist. Die Benutzer der Gruppe, die Sie zuordnen, haben über die aktivierten Protokolle Zugriff auf Ihre Amazon S3- oder Amazon EFS-Ressourcen AWS Transfer Family. Wenn Sie den Gruppennamen kennen, können Sie die SID-Werte anzeigen, indem Sie den folgenden Befehl unter Windows ausführen PowerShell.

```
Get-ADGroup -Filter {samAccountName -like "YourGroupName*"} -Properties
* | Select SamAccountName,ObjectSid
```

Ersetzen Sie diesen Befehl YourGroupNamedurch den Namen Ihrer Active Directory-Gruppe.

Der reguläre Ausdruck, der zur Überprüfung dieses Parameters verwendet wird, ist eine Zeichenfolge, die aus alphanumerischen Groß- und Kleinbuchstaben ohne Leerzeichen besteht. Sie können auch Unterstriche oder eines der folgenden Zeichen verwenden: =, . @: /-

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 256 Zeichen.

Pattern: `S-1-[\d-]+`

Erforderlich: Nein

HomeDirectory

Das Zielverzeichnis (Ordner) für einen Benutzer bei der Serveranmeldung mithilfe des Client.

Ein Beispiel für `HomeDirectory` ist `/bucket_name/home/mydirectory`.

> **Note**
>
> Der `HomeDirectory`-Parameter wird nur verwendet, wenn `HomeDirectoryType` auf `PATH` gesetzt ist.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 1024 Zeichen.

Pattern: `(|/.*)`

Erforderlich: Nein

HomeDirectoryMappings

Logische Verzeichniszuordnungen, die angeben, welche Amazon S3- oder Amazon EFS-Pfade und -Schlüssel für Ihren Benutzer sichtbar sein sollen und wie Sie sie sichtbar machen möchten. Sie müssen das `Entry` `Target` Und-Paar angeben, das `Entry` zeigt, wie der Pfad sichtbar gemacht `Target` wird und der tatsächliche Amazon S3- oder Amazon EFS-Pfad ist. Wenn Sie nur ein Ziel angeben, wird es unverändert angezeigt. Sie müssen außerdem sicherstellen, dass Ihre AWS Identity and Access Management (IAM-) Rolle Zugriff auf Pfade in `Target` bietet. Dieser Wert kann nur gesetzt werden, wenn er auf LOGICAL gesetzt `HomeDirectoryType` ist.

In den meisten Fällen können Sie diesen Wert anstelle der Sitzungsrichtlinie verwenden, um den zugehörigen Zugriff auf das angegebene Home-Verzeichnis (“ `chroot` „) zu sperren. Dazu können Sie den Wert `Entry` auf '/' und dann `Target` auf den `HomeDirectory` Parameterwert setzen.

Typ: Array von HomeDirectoryMapEntry-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Maximale Anzahl von 50000 Elementen.

Erforderlich: Nein

HomeDirectoryType

Die Art des Zielverzeichnisses (Ordners), das das Home-Verzeichnis Ihrer Benutzer sein soll, wenn sie sich beim Server anmelden. Wenn Sie es auf einstellen`PATH`, sieht der Benutzer den absoluten Amazon S3-Bucket- oder Amazon EFS-Pfad so, wie er in seinen File Transfer Protocol-Clients ist. Wenn Sie es auf einstellen`LOGICAL`, müssen Sie Zuordnungen dafür angeben, wie Sie Amazon S3- oder Amazon EFS-Pfade `HomeDirectoryMappings` für Ihre Benutzer sichtbar machen möchten.

> **Note**
>
> `HomeDirectoryType`Ist dies der `LOGICAL` Fall, müssen Sie mithilfe des Parameters Zuordnungen angeben. `HomeDirectoryMappings` Ist dies hingegen der Fall,

HomeDirectoryType geben Sie mithilfe des Parameters einen absoluten Pfad anHomeDirectory. PATH Sie können nicht beides HomeDirectory und HomeDirectoryMappings in Ihrer Vorlage haben.

Typ: Zeichenfolge

Zulässige Werte: PATH | LOGICAL

Erforderlich: Nein

Policy

Eine Sitzungsrichtlinie für Ihren Benutzer, sodass Sie dieselbe AWS Identity and Access Management (IAM-) Rolle für mehrere Benutzer verwenden können. Diese Richtlinie beschränkt den Zugriff eines Benutzers auf Teile seines Amazon S3 S3-Buckets. Variablen, die Sie in dieser Richtlinie verwenden können: ${Transfer:UserName}, ${Transfer:HomeDirectory} und ${Transfer:HomeBucket}.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 2048 Zeichen.

Erforderlich: Nein

PosixProfile

Die vollständige POSIX-Identität, einschließlich Benutzer-ID (Uid), Gruppen-ID (Gid) und sekundärer Gruppen-IDs (SecondaryGids), die den Zugriff Ihrer Benutzer auf Ihre Amazon EFS-Dateisysteme (Elastic File System) steuert. Die POSIX-Berechtigungen, die für Dateien und Verzeichnisse in Ihrem Dateisystem festgelegt sind, bestimmen die Zugriffsebene, die Ihre Benutzer beim Übertragen von Dateien in und aus Ihren Amazon EFS-Dateisystemen erhalten.

Typ: PosixProfile Objekt

Erforderlich: Nein

Role

Der Amazon-Ressourcenname (ARN) der Rolle AWS Identity and Access Management (IAM), die den Zugriff Ihrer Benutzer auf Ihren Amazon S3-Bucket oder Ihr Amazon EFS-Dateisystem steuert. Die mit dieser Rolle verbundenen Richtlinien bestimmen die Zugriffsebene, die Sie Ihren Benutzern beim Übertragen von Dateien in und aus Ihrem Amazon-S3-Bucket

oder Amazon-EFS-Dateisystem bereitstellen möchten. Die IAM-Rolle sollte außerdem eine Vertrauensstellung enthalten, mit der der Server Zugriff auf Ihre Ressourcen erhält, wenn er die Übertragungsanfragen Ihres Benutzers bearbeitet.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 2048 Zeichen.

Pattern: `arn:.*role/\S+`

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# DescribedAgreement

Beschreibt die Eigenschaften einer Vereinbarung.

## Inhalt

Arn

Der eindeutige Amazon-Ressourcenname (ARN) für die Vereinbarung.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 1600 Zeichen.

Pattern: `arn:\S+`

Erforderlich: Ja

AccessRole

Konnektoren werden verwendet, um Dateien entweder über das AS2- oder das SFTP-Protokoll zu senden. Geben Sie für die Zugriffsrolle den Amazon-Ressourcennamen (ARN) der zu AWS Identity and Access Management verwendenden Rolle an.

Für AS2-Konnektoren

Mit AS2 senden Sie Dateien, indem Sie `StartFileTransfer` aufrufen und die Dateipfade im Anforderungsparameter `SendFilePaths` angeben. Mit dem übergeordneten Verzeichnis der Datei (Beispiel: das übergeordnetes Verzeichnis für `--send-file-paths /bucket/dir/file.txt` ist `/bucket/dir/`) speichern wir eine verarbeitete AS2-Nachrichtendatei vorübergehend, speichern die MDN, wenn wir sie vom Partner erhalten, und schreiben eine endgültige JSON-Datei, die relevante Metadaten der Übertragung enthält. Daher muss `AccessRole` Lese- und Schreibzugriff auf das übergeordnete Verzeichnis des in der `StartFileTransfer`-Anforderung verwendeten Dateispeicherorts gewähren. Darüber hinaus müssen Sie Lese- und Schreibzugriff für das übergeordnete Verzeichnis der Dateien gewähren, die Sie mit `StartFileTransfer` senden möchten.

Wenn Sie die Standardauthentifizierung für Ihren AS2-Connector verwenden, erfordert die Zugriffsrolle die `secretsmanager:GetSecretValue` Erlaubnis für den geheimen Schlüssel. Wenn das Geheimnis mit einem vom Kunden verwalteten Schlüssel anstelle des AWS verwalteten Schlüssels in Secrets Manager verschlüsselt wird, benötigt die Rolle auch die `kms:Decrypt` Erlaubnis für diesen Schlüssel.

Für SFTP-Konnektoren

Stellen Sie sicher, dass die Zugriffsrolle Lese- und Schreibzugriff auf das übergeordnete Verzeichnis des Dateispeicherorts bietet, der in der `StartFileTransfer` Anfrage verwendet wird. Stellen Sie außerdem sicher, dass die Rolle die `secretsmanager:GetSecretValue` Berechtigung dazu bietet AWS Secrets Manager.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 2048 Zeichen.

Pattern: `arn:.*role/\S+`

Erforderlich: Nein

AgreementId

Eine eindeutige Kennung für die Vereinbarung. Diese Kennung wird zurückgegeben, wenn Sie eine Vereinbarung erstellen.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `a-([0-9a-f]{17})`

Erforderlich: Nein

BaseDirectory

Das Zielverzeichnis (der Ordner) für Dateien, die mithilfe des AS2-Protokolls übertragen werden.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 1024 Zeichen.

Pattern: `(|/.*)`

Erforderlich: Nein

Description

Der Name oder die kurze Beschreibung, der/die zur Identifizierung der Vereinbarung verwendet wird.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 1. Höchstlänge = 200 Zeichen.

Pattern: `[\p{Graph}]+`

Erforderlich: Nein

LocalProfileId

Eine eindeutige Kennung für das lokale AS2-Profil.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `p-([0-9a-f]{17})`

Erforderlich: Nein

PartnerProfileId

Eine eindeutige Kennung für das in der Vereinbarung verwendete Partnerprofil.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `p-([0-9a-f]{17})`

Erforderlich: Nein

ServerId

Eine vom System zugewiesene eindeutige ID für eine Server-Instance. Diese Kennung gibt den spezifischen Server an, den die Vereinbarung verwendet.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Erforderlich: Nein

Status

Der aktuelle Status der Vereinbarung, entweder `ACTIVE` oder `INACTIVE`.

Typ: Zeichenfolge

Zulässige Werte: `ACTIVE | INACTIVE`

Erforderlich: Nein

Tags

Schlüssel-Wert-Paare, die zur Gruppierung und Suche von Vereinbarungen verwendet werden können.

Typ: Array von Tag-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Die maximale Anzahl beträgt 50 Elemente.

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# DescribedCertificate

Beschreibt die Eigenschaften eines Zertifikats.

## Inhalt

Arn

Der eindeutige Amazon-Ressourcenname (ARN) für das Zertifikat.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 1600 Zeichen.

Pattern: `arn:\S+`

Erforderlich: Ja

ActiveDate

Ein optionales Datum, das angibt, wann das Zertifikat aktiv wird.

Typ: Zeitstempel

Erforderlich: Nein

Certificate

Der Dateiname des Zertifikats.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 16384 Zeichen.

Pattern: `[\u0009\u000A\u000D\u0020-\u00FF]*`

Erforderlich: Nein

CertificateChain

Die Liste der Zertifikate, aus denen sich die Zertifikatskette zusammensetzt.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Die maximale Länge beträgt 2097152.

Pattern: `[\u0009\u000A\u000D\u0020-\u00FF]*`

Erforderlich: Nein

CertificateId

Ein Array von Kennungen für die importierten Zertifikate. Sie verwenden diese Kennung für die Arbeit mit Profilen und Partnerprofilen.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 22.

Pattern: `cert-([0-9a-f]{17})`

Erforderlich: Nein

Description

Der Name oder die Beschreibung zur Identifizierung des Zertifikats.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 1. Höchstlänge = 200 Zeichen.

Pattern: `[\p{Graph}]+`

Erforderlich: Nein

InactiveDate

Ein optionales Datum, das angibt, wann das Zertifikat inaktiv wird.

Typ: Zeitstempel

Erforderlich: Nein

NotAfterDate

Das letzte Datum, an dem das Zertifikat gültig ist.

Typ: Zeitstempel

Erforderlich: Nein

NotBeforeDate

Das früheste Datum, an dem das Zertifikat gültig ist.

Typ: Zeitstempel

Erforderlich: Nein

Serial

Die Seriennummer des Zertifikats.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge von 48.

Pattern: `[\p{XDigit}{2}:?]*`

Erforderlich: Nein

Status

Das Zertifikat kann entweder `ACTIVE`, `PENDING_ROTATION` oder `INACTIVE` sein. `PENDING_ROTATION`bedeutet, dass dieses Zertifikat das aktuelle Zertifikat ersetzt, wenn letzteres abläuft.

Typ: Zeichenfolge

Zulässige Werte: `ACTIVE | PENDING_ROTATION | INACTIVE`

Erforderlich: Nein

Tags

Schlüssel-Wert-Paare, die zur Gruppierung und Suche von Zertifikaten verwendet werden.

Typ: Array von Tag-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Die maximale Anzahl beträgt 50 Elemente.

Erforderlich: Nein

Type

Wurde ein privater Schlüssel für das Zertifikat angegeben, ist sein Typ `CERTIFICATE_WITH_PRIVATE_KEY`. Ist kein privater Schlüssel vorhanden, ist der Typ `CERTIFICATE`.

Typ: Zeichenfolge

Zulässige Werte: `CERTIFICATE | CERTIFICATE_WITH_PRIVATE_KEY`

Erforderlich: Nein

Usage

Gibt an, wie dieses Zertifikat verwendet wird. Es kann auf folgende Weise verwendet werden:

- `SIGNING`: Zum Signieren von AS2-Nachrichten
- `ENCRYPTION`: Zum Verschlüsseln von AS2-Nachrichten
- `TLS`: Zur Sicherung von AS2-Kommunikation, die über HTTPS gesendet wird

Typ: Zeichenfolge

Zulässige Werte: `SIGNING | ENCRYPTION`

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# DescribedConnector

Beschreibt die Parameter für den Konnektor, wie durch den identifiziertConnectorId.

## Inhalt

Arn

Der eindeutige Amazon-Ressourcenname (ARN) für den Connector.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 1600 Zeichen.

Pattern: `arn:\S+`

Erforderlich: Ja

AccessRole

Konnektoren werden verwendet, um Dateien entweder über das AS2- oder das SFTP-Protokoll zu senden. Geben Sie für die Zugriffsrolle den Amazon-Ressourcennamen (ARN) der zu AWS Identity and Access Management verwendenden Rolle an.

Für AS2-Konnektoren

Mit AS2 senden Sie Dateien, indem Sie `StartFileTransfer` aufrufen und die Dateipfade im Anforderungsparameter `SendFilePaths` angeben. Mit dem übergeordneten Verzeichnis der Datei (Beispiel: das übergeordnetes Verzeichnis für `--send-file-paths /bucket/dir/file.txt` ist `/bucket/dir/`) speichern wir eine verarbeitete AS2-Nachrichtendatei vorübergehend, speichern die MDN, wenn wir sie vom Partner erhalten, und schreiben eine endgültige JSON-Datei, die relevante Metadaten der Übertragung enthält. Daher muss `AccessRole` Lese- und Schreibzugriff auf das übergeordnete Verzeichnis des in der `StartFileTransfer`-Anforderung verwendeten Dateispeicherorts gewähren. Darüber hinaus müssen Sie Lese- und Schreibzugriff für das übergeordnete Verzeichnis der Dateien gewähren, die Sie mit `StartFileTransfer` senden möchten.

Wenn Sie die Standardauthentifizierung für Ihren AS2-Connector verwenden, erfordert die Zugriffsrolle die `secretsmanager:GetSecretValue` Erlaubnis für den geheimen Schlüssel. Wenn das Geheimnis mit einem vom Kunden verwalteten Schlüssel anstelle des AWS verwalteten Schlüssels in Secrets Manager verschlüsselt wird, benötigt die Rolle auch die `kms:Decrypt` Erlaubnis für diesen Schlüssel.

Für SFTP-Konnektoren

Stellen Sie sicher, dass die Zugriffsrolle Lese- und Schreibzugriff auf das übergeordnete Verzeichnis des Dateispeicherorts bietet, der in der `StartFileTransfer` Anfrage verwendet wird. Stellen Sie außerdem sicher, dass die Rolle die `secretsmanager:GetSecretValue` Berechtigung dazu bietet AWS Secrets Manager.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 2048 Zeichen.

Pattern: `arn:.*role/\S+`

Erforderlich: Nein

As2Config

Eine Struktur, die die Parameter für ein AS2-Connector-Objekt enthält.

Typ: As2ConnectorConfig Objekt

Erforderlich: Nein

ConnectorId

Der eindeutige Bezeichner für den Konnektor.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `c-([0-9a-f]{17})`

Erforderlich: Nein

LoggingRole

Der Amazon-Ressourcenname (ARN) der Rolle AWS Identity and Access Management (IAM), der es einem Connector ermöglicht, die CloudWatch Protokollierung für Amazon S3-Ereignisse zu aktivieren. Wenn diese Option aktiviert ist, können Sie die Connector-Aktivität in Ihren CloudWatch Protokollen einsehen.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 2048 Zeichen.

Pattern: `arn:.*role/\S+`

Erforderlich: Nein

SecurityPolicyName

Der Textname der Sicherheitsrichtlinie für den angegebenen Connector.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 100 Zeichen.

Pattern: `TransferSFTPConnectorSecurityPolicy-[A-Za-z0-9-]+`

Erforderlich: Nein

ServiceManagedEgressIpAddresses

Die Liste der ausgehenden IP-Adressen dieses Connectors. Diese IP-Adressen werden automatisch zugewiesen, wenn Sie den Connector erstellen.

Typ: Zeichenfolgen-Array

Pattern: `\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3}`

Erforderlich: Nein

SftpConfig

Eine Struktur, die die Parameter für ein SFTP-Connector-Objekt enthält.

Typ: SftpConnectorConfig Objekt

Erforderlich: Nein

Tags

Schlüssel-Wert-Paare, die zur Gruppierung und Suche von Connectors verwendet werden.

Typ: Array von Tag-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Die maximale Anzahl beträgt 50 Elemente.

Erforderlich: Nein

Url

Die URL des AS2- oder SFTP-Endpunkts des Partners.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 255 Zeichen.

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# DescribedExecution

Die Details für ein Ausführungsobjekt.

## Inhalt

ExecutionId

Eine eindeutige Kennung für die Ausführung eines Workflows.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 36.

Pattern: `[0-9a-fA-F]{8}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{12}`

Erforderlich: Nein

ExecutionRole

Die mit der Ausführung verknüpfte IAM-Rolle.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 2048 Zeichen.

Pattern: `arn:.*role/\S+`

Erforderlich: Nein

InitialFileLocation

Eine Struktur, die den Speicherort der Amazon S3- oder EFS-Datei beschreibt. Dies ist der Dateispeicherort, an dem die Ausführung beginnt: Wenn die Datei kopiert wird, ist dies der ursprüngliche Dateispeicherort (und nicht der Zielspeicherort).

Typ: FileLocation Objekt

Erforderlich: Nein

LoggingConfiguration

Die der Ausführung zugeordnete IAM-Protokollierungsrolle.

Typ: LoggingConfiguration Objekt

Erforderlich: Nein

PosixProfile

Die vollständige POSIX-Identität, einschließlich Benutzer-ID (`Uid`), Gruppen-ID (`Gid`) und sekundärer Gruppen-IDs (`SecondaryGids`), die den Zugriff Ihrer Benutzer auf Ihre Amazon EFS-Dateisysteme (Elastic File System) steuert. Die POSIX-Berechtigungen, die für Dateien und Verzeichnisse in Ihrem Dateisystem festgelegt sind, bestimmen die Zugriffsebene, die Ihre Benutzer beim Übertragen von Dateien in und aus Ihren Amazon EFS-Dateisystemen erhalten.

Typ: PosixProfile Objekt

Erforderlich: Nein

Results

Eine Struktur, die die Ausführungsergebnisse beschreibt. Dies beinhaltet eine Liste der Schritte zusammen mit den Details zu jedem Schritt, dem Fehlertyp und der Fehlermeldung (falls vorhanden) sowie der `OnExceptionSteps` Struktur.

Typ: ExecutionResults Objekt

Erforderlich: Nein

ServiceMetadata

Ein Container-Objekt für die Sitzungsdetails, die einem Workflow zugeordnet sind.

Typ: ServiceMetadata Objekt

Erforderlich: Nein

Status

Der Status ist „Ausführung“. Kann in Bearbeitung sein, abgeschlossen sein, eine Ausnahme aufgetreten sein oder die Ausnahme wird behandelt.

Typ: Zeichenfolge

Zulässige Werte: `IN_PROGRESS | COMPLETED | EXCEPTION | HANDLING_EXCEPTION`

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# DescribedHostKey

Die Details für einen Server-Host-Schlüssel.

## Inhalt

Arn

Der eindeutige Amazon-Ressourcenname (ARN) für den Hostschlüssel.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 1600 Zeichen.

Pattern: `arn:\S+`

Erforderlich: Ja

DateImported

Das Datum, an dem der Hostschlüssel zum Server hinzugefügt wurde.

Typ: Zeitstempel

Erforderlich: Nein

Description

Die Textbeschreibung für diesen Hostschlüssel.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Höchstlänge = 200 Zeichen.

Pattern: `[\p{Print}]*`

Erforderlich: Nein

HostKeyFingerprint

Der Fingerabdruck des öffentlichen Schlüssels, bei dem es sich um eine kurze Bytefolge handelt, die zur Identifizierung des längeren öffentlichen Schlüssels verwendet wird.

Typ: Zeichenfolge

Erforderlich: Nein

HostKeyId

Eine eindeutige Kennung für den Host-Schlüssel.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 25.

Pattern: `hostkey-[0-9a-f]{17}`

Erforderlich: Nein

Tags

Schlüssel-Wert-Paare, die verwendet werden können, um Hostschlüssel zu gruppieren und nach ihnen zu suchen.

Typ: Array von Tag-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Die maximale Anzahl beträgt 50 Elemente.

Erforderlich: Nein

Type

Der Verschlüsselungsalgorithmus, der für den Hostschlüssel verwendet wird. Der `Type` Parameter wird mit einem der folgenden Werte angegeben:

- `ssh-rsa`
- `ssh-ed25519`
- `ecdsa-sha2-nistp256`
- `ecdsa-sha2-nistp384`
- `ecdsa-sha2-nistp521`

Typ: Zeichenfolge

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# DescribedProfile

Die Details für ein lokales AS2-Profil oder ein Partner-AS2-Profil.

## Inhalt

Arn

Der eindeutige Amazon-Ressourcenname (ARN) für das Profil.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 1600 Zeichen.

Pattern: `arn:\S+`

Erforderlich: Ja

As2Id

Die `As2Id` ist der AS2-Name, wie in RFC 4130 definiert. Bei eingehenden Übertragungen ist dies der `AS2-From`-Header für die vom Partner gesendeten AS2-Nachrichten. Bei ausgehenden Connectors ist dies der `AS2-To`-Header für die AS2-Nachrichten, die mithilfe der `StartFileTransfer`-API-Operation an den Partner gesendet werden. Diese ID darf keine Leerzeichen enthalten.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 128 Zeichen.

Pattern: `[\p{Print}\s]*`

Erforderlich: Nein

CertificateIds

Ein Array von Kennungen für die importierten Zertifikate. Sie verwenden diese Kennung für die Arbeit mit Profilen und Partnerprofilen.

Typ: Zeichenfolgen-Array

Längenbeschränkungen: Feste Länge von 22.

Pattern: `cert-([0-9a-f]{17})`

Erforderlich: Nein

ProfileId

Eine eindeutige Kennung für das lokale AS2-Profil oder das AS2-Profil eines Partners.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `p-([0-9a-f]{17})`

Erforderlich: Nein

ProfileType

Gibt an, ob nur `LOCAL`- oder nur `PARTNER`-Typprofile aufgelistet werden sollen. Sind diese nicht in der Anforderung nicht, listet der Befehl alle Profilarten auf.

Typ: Zeichenfolge

Zulässige Werte: `LOCAL | PARTNER`

Erforderlich: Nein

Tags

Schlüssel-Wert-Paare, die zur Gruppierung und Suche von Profilen verwendet werden.

Typ: Array von Tag-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Die maximale Anzahl beträgt 50 Elemente.

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2

- AWS SDK for Ruby V3

# DescribedSecurityPolicy

Beschreibt die Eigenschaften einer Sicherheitsrichtlinie, die Sie angeben. Weitere Informationen zu Sicherheitsrichtlinien finden Sie unter Arbeiten mit Sicherheitsrichtlinien für Server oder Arbeiten mit Sicherheitsrichtlinien für SFTP-Connectors.

## Inhalt

SecurityPolicyName

Der Textname der angegebenen Sicherheitsrichtlinie.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 100 Zeichen.

Pattern: `Transfer[A-Za-z0-9]*SecurityPolicy-[A-Za-z0-9-]+`

Erforderlich: Ja

Fips

Gibt an, ob diese Richtlinie die Federal Information Processing Standards (FIPS) aktiviert. Dieser Parameter gilt sowohl für Server- als auch für Connector-Sicherheitsrichtlinien.

Typ: Boolesch

Erforderlich: Nein

Protocols

Listet die Dateiübertragungsprotokolle auf, für die die Sicherheitsrichtlinie gilt.

Typ: Zeichenfolgen-Array

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Die maximale Anzahl beträgt 5 Elemente.

Zulässige Werte: `SFTP | FTPS`

Erforderlich: Nein

SshCiphers

Führt die aktivierten Verschlüsselungsalgorithmen für Secure Shell (SSH) in der Sicherheitsrichtlinie auf, die an den Server oder Connector angehängt ist. Dieser Parameter gilt sowohl für Server- als auch für Connector-Sicherheitsrichtlinien.

Typ: Zeichenfolgen-Array

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge = 50 Zeichen.

Erforderlich: Nein

SshHostKeyAlgorithms

Listet die Host-Schlüsselalgorithmen für die Sicherheitsrichtlinie auf.

> **Note**
>
> Dieser Parameter gilt nur für Sicherheitsrichtlinien für Connectors.

Typ: Zeichenfolgen-Array

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge = 50 Zeichen.

Erforderlich: Nein

SshKexs

Listet die aktivierten Verschlüsselungsalgorithmen für den SSH-Schlüsselaustausch (KEX) in der Sicherheitsrichtlinie auf, die an den Server oder Connector angehängt ist. Dieser Parameter gilt sowohl für Server- als auch für Connector-Sicherheitsrichtlinien.

Typ: Zeichenfolgen-Array

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge = 50 Zeichen.

Erforderlich: Nein

SshMacs

Führt die aktivierten SSH-MAC-Verschlüsselungsalgorithmen (Message Authentication Code) in der Sicherheitsrichtlinie auf, die an den Server oder Connector angehängt ist. Dieser Parameter gilt sowohl für Server- als auch für Connector-Sicherheitsrichtlinien.

Typ: Zeichenfolgen-Array

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge = 50 Zeichen.

Erforderlich: Nein

TlsCiphers

Führt die aktivierten TLS-Verschlüsselungsalgorithmen (Transport Layer Security) in der Sicherheitsrichtlinie auf, die an den Server angehängt ist.

> **Note**
>
> Dieser Parameter gilt nur für Sicherheitsrichtlinien für Server.

Typ: Zeichenfolgen-Array

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge = 50 Zeichen.

Erforderlich: Nein

Type

Der Ressourcentyp, für den die Sicherheitsrichtlinie gilt, entweder Server oder Connector.

Typ: Zeichenfolge

Zulässige Werte: `SERVER | CONNECTOR`

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# DescribedServer

Beschreibt die Eigenschaften eines Servers mit aktiviertem Dateiübertragungsprotokoll, der angegeben wurde.

## Inhalt

Arn

Gibt den eindeutigen Amazon-Ressourcennamen (ARN) des Servers an.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 1600 Zeichen.

Pattern: `arn:\S+`

Erforderlich: Ja

As2ServiceManagedEgressIpAddresses

Die Liste der Ausgangs-IP-Adressen dieses Servers. Diese IP-Adressen sind nur für Server relevant, die das AS2-Protokoll verwenden. Sie werden zum Senden von asynchronen mDNS verwendet.

Diese IP-Adressen werden automatisch zugewiesen, wenn Sie einen AS2-Server erstellen. Wenn Sie einen vorhandenen Server aktualisieren und das AS2-Protokoll hinzufügen, werden außerdem statische IP-Adressen zugewiesen.

Typ: Zeichenfolgen-Array

Pattern: `\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3}`

Erforderlich: Nein

Certificate

Gibt den ARN des AWS Certificate Manager (ACM) -Zertifikats an. Erforderlich, wenn `Protocols` auf `FTPS` eingestellt ist.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 1600 Zeichen.

Erforderlich: Nein

Domain

Gibt die Domain des Speichersystems an, das für Dateiübertragungen verwendet wird. Es sind zwei Domänen verfügbar: Amazon Simple Storage Service (Amazon S3) und Amazon Elastic File System (Amazon EFS). Der Standardwert ist S3.

Typ: Zeichenfolge

Zulässige Werte: `S3 | EFS`

Erforderlich: Nein

EndpointDetails

Die Virtual Private Cloud (VPC)-Endpunkt-Einstellungen, die für Ihren Server konfiguriert sind. Wenn Sie Ihren Endpunkt in Ihrer VPC hosten, können Sie ihn nur für Ressourcen innerhalb Ihrer VPC zugänglich machen, oder Sie können elastische IP-Adressen anfügen und ihn für Clients über das Internet zugänglich machen. Die Standard-Sicherheitsgruppen Ihrer VPC werden Ihrem Endpunkt automatisch zugewiesen.

Typ: EndpointDetails Objekt

Erforderlich: Nein

EndpointType

Definiert den Endpunkttyp, mit dem Ihr Server verbunden ist. Wenn Ihr Server mit einem VPC-Endpunkt verbunden ist, ist Ihr Server nicht über das öffentliche Internet zugänglich.

Typ: Zeichenfolge

Zulässige Werte: `PUBLIC | VPC | VPC_ENDPOINT`

Erforderlich: Nein

HostKeyFingerprint

Gibt den Base64-codierten SHA256-Fingerabdruck des Hostschlüssels des Servers an. Dieser Wert entspricht der Ausgabe des Befehls. `ssh-keygen -l -f my-new-server-key`

Typ: Zeichenfolge

Erforderlich: Nein

IdentityProviderDetails

Gibt Informationen zum Aufrufen einer vom Kunden bereitgestellten Authentifizierungs-API an. Dieses Feld wird nicht gefüllt, wenn `IdentityProviderType` der eines Servers `AWS_DIRECTORY_SERVICE` oder `SERVICE_MANAGED` ist.

Typ: IdentityProviderDetails Objekt

Erforderlich: Nein

IdentityProviderType

Das Authentifizierungsverfahren für einen Server. Der Standardwert ist`SERVICE_MANAGED`, der es Ihnen ermöglicht, Benutzeranmeldeinformationen innerhalb des AWS Transfer Family Dienstes zu speichern und darauf zuzugreifen.

Wird verwendet`AWS_DIRECTORY_SERVICE`, um Zugriff auf Active Directory-Gruppen in AWS Directory Service for Microsoft Active Directory oder Microsoft Active Directory in Ihrer lokalen Umgebung oder bei der AWS Verwendung von AD Connector bereitzustellen. Für diese Option ist es auch erforderlich, dass Sie mithilfe des Parameters `IdentityProviderDetails` eine Directory-ID angeben.

Verwenden Sie den `API_GATEWAY`-Wert für die Integration eines Identitätsanbieters Ihrer Wahl. Die `API_GATEWAY`-Einstellung erfordert, dass Sie mithilfe des Parameters `IdentityProviderDetails` die URL eines Amazon-API-Gateway-Endpunkts angeben, der zur Authentifizierung aufgerufen wird.

Verwenden Sie den `AWS_LAMBDA` Wert, um eine AWS Lambda Funktion direkt als Identitätsanbieter zu verwenden. Wenn Sie diesen Wert wählen, müssen Sie den ARN für die Lambda-Funktion im `Function` Parameter für den `IdentityProviderDetails` Datentyp angeben.

Typ: Zeichenfolge

Zulässige Werte: `SERVICE_MANAGED | API_GATEWAY | AWS_DIRECTORY_SERVICE | AWS_LAMBDA`

Erforderlich: Nein

LoggingRole

Der Amazon-Ressourcenname (ARN) der AWS Identity and Access Management (IAM) -Rolle, der es einem Server ermöglicht, die CloudWatch Amazon-Protokollierung für Amazon S3 oder

Amazon EFSEvents zu aktivieren. Wenn diese Option aktiviert ist, können Sie Benutzeraktivitäten in Ihren Protokollen einsehen. CloudWatch

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 2048 Zeichen.

Pattern: `(|arn:.*role/\S+)`

Erforderlich: Nein

PostAuthenticationLoginBanner

Gibt eine Zeichenfolge an, die angezeigt wird, wenn Benutzer sich mit einem Server verbinden. Diese Zeichenfolge wird nach Authentifizierung des Benutzers angezeigt.

> **Note**
>
> Das SFTP-Protokoll unterstützt keine Anzeige-Banner nach der Authentifizierung.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge von 4096.

Pattern: `[\x09-\x0D\x20-\x7E]*`

Erforderlich: Nein

PreAuthenticationLoginBanner

Gibt eine Zeichenfolge an, die angezeigt wird, wenn Benutzer sich mit einem Server verbinden. Diese Zeichenfolge wird angezeigt, bevor sich der Benutzer authentifiziert. Das folgende Banner zeigt beispielsweise Details zur Verwendung des Systems an:

```
This system is for the use of authorized users only. Individuals using
this computer system without authority, or in excess of their authority,
are subject to having all of their activities on this system monitored
and recorded by system personnel.
```

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge von 4096.

Pattern: `[\x09-\x0D\x20-\x7E]*`

Erforderlich: Nein

ProtocolDetails

Protokolleinstellungen, die für Ihren Server konfiguriert sind.

- Verwenden Sie den Parameter `PassiveIp` zur Angabe des passiven Modus (für FTP- und FTPS-Protokolle). Geben Sie eine einzelne gepunktete IPv4-Adresse ein, z. B. die externe IP-Adresse einer Firewall, eines Routers oder eines Load Balancers.
- Verwenden Sie den Parameter `SetStatOption`, um den Fehler zu ignorieren, der generiert wird, wenn der Client versucht, den Befehl `SETSTAT` für eine Datei zu verwenden, die Sie in einen Amazon-S3-Bucket hochladen. Damit der AWS Transfer Family Server den `SETSTAT` Befehl ignoriert und Dateien hochlädt, ohne dass Sie Änderungen an Ihrem SFTP-Client vornehmen müssen, setzen Sie den Wert auf. `ENABLE_NO_OP` Wenn Sie den `SetStatOption` Parameter auf setzen`ENABLE_NO_OP`, generiert Transfer Family einen Protokolleintrag in Amazon CloudWatch Logs, sodass Sie feststellen können, wann der Client einen `SETSTAT` Anruf tätigt.
- Verwenden Sie den `TlsSessionResumptionMode` Parameter, um festzustellen, ob Ihr AWS Transfer Family Server die letzten, ausgehandelten Sitzungen über eine eindeutige Sitzungs-ID wieder aufnimmt.
- `As2Transports` gibt an, wie AS2-Nachrichten transportiert werden sollen. Derzeit wird nur HTTP unterstützt.

Typ: ProtocolDetails Objekt

Erforderlich: Nein

Protocols

Gibt das/die Dateiübertragungsprotokoll(e) an, über das/die der Dateiübertragungsprotokoll-Client eine Verbindung zum Endpunkt des Servers herstellen kann. Die verfügbaren Protokolle sind:

- `SFTP` (Secure Shell (SSH) File Transfer Protocol): Dateiübertragung über SSH
- `FTPS` (File Transfer Protocol Secure): Dateiübertragung mit TLS-Verschlüsselung
- `FTP` (File Transfer Protocol): Unverschlüsselte Dateiübertragung
- `AS2`(Anwendbarkeitserklärung 2): Wird für den Transport strukturierter Daten verwendet business-to-business

> **Note**
>
> - Wenn Sie wählen`FTPS`, müssen Sie ein in AWS Certificate Manager (ACM) gespeichertes Zertifikat wählen, das zur Identifizierung Ihres Servers verwendet wird, wenn Clients über FTPS eine Verbindung zu ihm herstellen.
> - Wenn `Protocol` entweder `FTP` oder `FTPS` enthält, muss `EndpointType` `VPC` lauten und `IdentityProviderType` muss `AWS_DIRECTORY_SERVICE`, `AWS_LAMBDA` oder `API_GATEWAY` lauten.
> - Wenn `Protocol` `FTP` enthält, können `AddressAllocationIds` nicht zugeordnet werden.
> - Wenn `Protocol` nur auf `SFTP` gesetzt ist, kann `EndpointType` auf `PUBLIC` gesetzt werden und `IdentityProviderType` kann auf einen der unterstützten Identitätstypen gesetzt werden: `SERVICE_MANAGED`, `AWS_DIRECTORY_SERVICE`, `AWS_LAMBDA` oder `API_GATEWAY`.
> - Wenn das `Protocol` `AS2` enthält, dann muss der `EndpointType` `VPC` lauten und die Domain muss Amazon S3 sein.

Typ: Zeichenfolgen-Array

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Maximale Anzahl von 4 Elementen.

Zulässige Werte: `SFTP | FTP | FTPS | AS2`

Erforderlich: Nein

S3StorageOptions

Gibt an, ob die Leistung für Ihre Amazon S3 S3-Verzeichnisse optimiert ist oder nicht. Diese ist standardmäßig deaktiviert.

Standardmäßig haben Zuordnungen von Home-Verzeichnissen einen `TYPE` Wert von. `DIRECTORY` Wenn Sie diese Option aktivieren, müssten Sie den Wert dann explizit auf setzen, `FILE` wenn Sie möchten `HomeDirectoryMapEntryType`, dass eine Zuordnung ein Dateiziel hat.

Typ: S3StorageOptions Objekt

Erforderlich: Nein

SecurityPolicyName

Gibt den Namen der Sicherheitsrichtlinie für den Server an.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 100 Zeichen.

Pattern: `Transfer[A-Za-z0-9]*SecurityPolicy-[A-Za-z0-9-]+`

Erforderlich: Nein

ServerId

Gibt den eindeutigen, vom System zugewiesenen Bezeichner für einen Server an, den Sie instanziieren.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Erforderlich: Nein

State

Der Zustand des Servers, der beschrieben wurde. Der Wert von `ONLINE` gibt an, dass der Server Aufträge annehmen und Dateien übertragen kann. `State`Der Wert von `OFFLINE` bedeutet, dass der Server keine Dateiübertragungsvorgänge ausführen kann.

Der Status von `STARTING` und `STOPPING` gibt an, dass sich der Server in einem Zwischenzustand befindet, entweder nicht vollständig antworten kann oder nicht vollständig offline ist. Die Werte von `START_FAILED` oder `STOP_FAILED` können auf einen Fehler hinweisen.

Typ: Zeichenfolge

Zulässige Werte: `OFFLINE | ONLINE | STARTING | STOPPING | START_FAILED | STOP_FAILED`

Erforderlich: Nein

StructuredLogDestinations

Gibt die Protokollgruppen an, an die Ihre Serverprotokolle gesendet werden.

Um eine Protokollgruppe anzugeben, müssen Sie den ARN für eine bestehende Protokollgruppe angeben. In diesem Fall lautet das Format der Protokollgruppe wie folgt:

`arn:aws:logs:region-name:amazon-account-id:log-group:log-group-name:*`

Beispiel: `arn:aws:logs:us-east-1:111122223333:log-group:mytestgroup:*`

Wenn Sie zuvor eine Protokollgruppe für einen Server angegeben haben, können Sie diese löschen und somit die strukturierte Protokollierung deaktivieren, indem Sie in einem `update-server` Aufruf einen leeren Wert für diesen Parameter angeben. Beispielsweise:

```
update-server --server-id s-1234567890abcdef0 --structured-log-destinations
```

Typ: Zeichenfolge-Array

Array-Mitglieder: Die Mindestanzahl beträgt 0 Elemente. Die maximale Anzahl beträgt 1 Element.

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 1600 Zeichen.

Pattern: `arn:\S+`

Erforderlich: Nein

Tags

Gibt die Schlüssel-Wert-Paare an, mit denen Sie nach Servern suchen und diese gruppieren können, die dem beschriebenen Server zugewiesen wurden.

Typ: Array von Tag-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Die maximale Anzahl beträgt 50 Elemente.

Erforderlich: Nein

UserCount

Gibt die Anzahl der Benutzer an, die einem Server zugewiesen sind, den Sie mit dem angegeben haben. `ServerId`

Typ: Ganzzahl

Erforderlich: Nein

WorkflowDetails

Gibt die Workflow-ID für den zuzuweisenden Workflow und die für die Ausführung des Workflows verwendete Ausführungsrolle an.

Zusätzlich zu einem Workflow, der ausgeführt wird, wenn eine Datei vollständig hochgeladen wurde, kann `WorkflowDetails` auch eine Workflow-ID (und Ausführungsrolle) für einen Workflow enthalten, der beim teilweisen Upload ausgeführt werden soll. Ein teilweiser Upload erfolgt, wenn die Serversitzung unterbrochen wird, während die Datei noch hochgeladen wird.

Typ: WorkflowDetails Objekt

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# DescribedUser

Beschreibt die Eigenschaften eines angegebenen Benutzers.

## Inhalt

Arn

Gibt den eindeutigen Amazon-Ressourcennamen (ARN) für den Benutzer an, dessen Beschreibung angefordert wurde.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 1600 Zeichen.

Pattern: `arn:\S+`

Erforderlich: Ja

HomeDirectory

Das Zielverzeichnis (Ordner) für einen Benutzer bei der Serveranmeldung mithilfe des Client.

Ein Beispiel für `HomeDirectory` ist `/bucket_name/home/mydirectory`.

> **Note**
>
> Der `HomeDirectory`-Parameter wird nur verwendet, wenn `HomeDirectoryType` auf `PATH` gesetzt ist.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 1024 Zeichen.

Pattern: `(|/.*)`

Erforderlich: Nein

HomeDirectoryMappings

Logische Verzeichniszuordnungen, die angeben, welche Amazon S3- oder Amazon EFS-Pfade und -Schlüssel für Ihren Benutzer sichtbar sein sollen und wie Sie sie sichtbar machen möchten.

Sie müssen das `Entry` `Target` Und-Paar angeben, das `Entry` zeigt, wie der Pfad sichtbar gemacht `Target` wird und der tatsächliche Amazon S3- oder Amazon EFS-Pfad ist. Wenn Sie nur ein Ziel angeben, wird es unverändert angezeigt. Sie müssen außerdem sicherstellen, dass Ihre AWS Identity and Access Management (IAM-) Rolle Zugriff auf Pfade in `Target` bietet. Dieser Wert kann nur gesetzt werden, wenn er auf LOGICAL gesetzt `HomeDirectoryType` ist.

In den meisten Fällen können Sie diesen Wert anstelle der Sitzungsrichtlinie verwenden, um Ihren Benutzer auf das angegebene Home-Verzeichnis (“ `chroot` „) zu beschränken. Dazu können Sie den Wert `Entry` auf '/' und dann `Target` auf den HomeDirectory Parameterwert setzen.

Typ: Array von HomeDirectoryMapEntry-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Maximale Anzahl von 50000 Elementen.

Erforderlich: Nein

HomeDirectoryType

Die Art des Zielverzeichnisses (Ordners), das das Home-Verzeichnis Ihrer Benutzer sein soll, wenn sie sich beim Server anmelden. Wenn Sie es auf einstellen`PATH`, sieht der Benutzer den absoluten Amazon S3-Bucket- oder Amazon EFS-Pfad so, wie er in seinen File Transfer Protocol-Clients ist. Wenn Sie es auf einstellen`LOGICAL`, müssen Sie Zuordnungen dafür angeben, wie Sie Amazon S3- oder Amazon EFS-Pfade `HomeDirectoryMappings` für Ihre Benutzer sichtbar machen möchten.

> **Note**
>
> `HomeDirectoryType`Ist dies der `LOGICAL` Fall, müssen Sie mithilfe des Parameters Zuordnungen angeben. `HomeDirectoryMappings` Ist dies hingegen der Fall, `HomeDirectoryType` geben Sie mithilfe des Parameters einen absoluten Pfad an`HomeDirectory`. `PATH` Sie können nicht beides `HomeDirectory` und `HomeDirectoryMappings` in Ihrer Vorlage haben.

Typ: Zeichenfolge

Zulässige Werte: `PATH | LOGICAL`

Erforderlich: Nein

Policy

Eine Sitzungsrichtlinie für Ihren Benutzer, sodass Sie dieselbe AWS Identity and Access Management (IAM-) Rolle für mehrere Benutzer verwenden können. Diese Richtlinie beschränkt den Zugriff eines Benutzers auf Teile seines Amazon S3 S3-Buckets. Variablen, die Sie in dieser Richtlinie verwenden können: `${Transfer:UserName}`, `${Transfer:HomeDirectory}` und `${Transfer:HomeBucket}`.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 2048 Zeichen.

Erforderlich: Nein

PosixProfile

Gibt die vollständige POSIX-Identität an, einschließlich Benutzer-ID (`Uid`), Gruppen-ID (`Gid`) und sekundärer Gruppen-IDs (`SecondaryGids`), die den Zugriff Ihrer Benutzer auf Ihre Amazon Elastic File System (Amazon EFS)-Dateisysteme steuert. Die POSIX-Berechtigungen, die für Dateien und Verzeichnisse in Ihrem Dateisystem festgelegt sind, bestimmen die Zugriffsebene, die Ihre Benutzer beim Übertragen von Dateien in und aus Ihren Amazon EFS-Dateisystemen erhalten.

Typ: PosixProfile Objekt

Erforderlich: Nein

Role

Der Amazon-Ressourcenname (ARN) der Rolle AWS Identity and Access Management (IAM), die den Zugriff Ihrer Benutzer auf Ihren Amazon S3-Bucket oder Ihr Amazon EFS-Dateisystem steuert. Die mit dieser Rolle verbundenen Richtlinien bestimmen die Zugriffsebene, die Sie Ihren Benutzern beim Übertragen von Dateien in und aus Ihrem Amazon-S3-Bucket oder Amazon-EFS-Dateisystem bereitstellen möchten. Die IAM-Rolle sollte außerdem eine Vertrauensstellung enthalten, mit der der Server Zugriff auf Ihre Ressourcen erhält, wenn er die Übertragungsanfragen Ihres Benutzers bearbeitet.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 2048 Zeichen.

Pattern: `arn:.*role/\S+`

Erforderlich: Nein

SshPublicKeys

Gibt den öffentlichen Schlüsselteil der für den beschriebenen Benutzer gespeicherten Secure Shell (SSH)-Schlüssel an.

Typ: Array von SshPublicKey-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 0 Elemente. Die maximale Anzahl beträgt 5 Elemente.

Erforderlich: Nein

Tags

Gibt die Schlüssel-Wert-Paare für den angeforderten Benutzer an. Tag kann für eine Vielzahl von Zwecken verwendet werden, um nach Benutzern zu suchen und diese zu gruppieren.

Typ: Array von Tag-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Die maximale Anzahl beträgt 50 Elemente.

Erforderlich: Nein

UserName

Gibt den Namen des Benutzers an, dessen Beschreibung angefordert wurde. Benutzernamen werden zu Authentifizierungszwecken verwendet. Dies ist die Zeichenfolge, die von Ihrem Benutzer verwendet wird, wenn er sich bei Ihrem Server anmeldet.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 3. Maximale Länge beträgt 100 Zeichen.

Pattern: `[\w][\w@.-]{2,99}`

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++

- AWS SDK for Java V2
- AWS SDK for Ruby V3

# DescribedWorkflow

Beschreibt die Eigenschaften des angegebenen Workflows

## Inhalt

Arn

Gibt den eindeutigen Amazon-Ressourcennamen (ARN) für den Workflow an.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 1600 Zeichen.

Pattern: `arn:\S+`

Erforderlich: Ja

Description

Gibt die Textbeschreibung für den Workflow an.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 256 Zeichen.

Pattern: `[\w- ]*`

Erforderlich: Nein

OnExceptionSteps

Gibt die Schritte (Aktionen) an, die ausgeführt werden sollen, wenn bei der Ausführung des Workflows Fehler auftreten.

Typ: Array von WorkflowStep-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 0 Elemente. Maximale Anzahl von 8 Artikeln.

Erforderlich: Nein

Steps

Gibt die Details für die Schritte an, die im angegebenen Workflow enthalten sind.

Typ: Array von WorkflowStep-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 0 Elemente. Maximale Anzahl von 8 Artikeln.

Erforderlich: Nein

Tags

Schlüssel-Wert-Paare, die zum Gruppieren und Suchen nach Workflows verwendet werden können. Tags sind Metadaten, die zu beliebigen Zwecken an Workflows angefügt werden.

Typ: Array von Tag-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Die maximale Anzahl beträgt 50 Elemente.

Erforderlich: Nein

WorkflowId

Eine eindeutige Kennung für den Workflow.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `w-([a-z0-9]{17})`

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# EfsFileLocation

Gibt die Details für den Dateispeicherort der Datei an, die im Workflow verwendet wird. Gilt nur, wenn Sie Amazon Elastic File Systems (Amazon EFS) als Speicher verwenden.

## Inhalt

FileSystemId

Die ID des Dateisystems, die von Amazon EFS zugewiesen wurde.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. 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: Nein

Path

Der Pfadname für den Ordner, der von einem Workflow verwendet wird.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Die maximale Länge beträgt 65536.

Pattern: `[^\x00]+`

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# EndpointDetails

Die VPC-Endpunkteinstellungen (Virtual Private Cloud), die für Ihren Server mit aktiviertem Dateiübertragungsprotokoll konfiguriert sind. Mit einem VPC-Endpunkt können Sie den Zugriff auf Ihren Server und Ressourcen nur innerhalb Ihrer VPC einschränken. Um den eingehenden Internetverkehr zu kontrollieren, rufen Sie die `UpdateServer` API auf und hängen Sie eine Elastic IP-Adresse an den Endpunkt Ihres Servers an.

> **Note**
>
> Nach dem 19. Mai 2021 können Sie `EndpointType=VPC_ENDPOINT` in Ihrem AWS Konto keinen Server mehr erstellen, wenn Ihr Konto dies nicht bereits vor dem 19. Mai 2021 getan hat. Wenn du am oder vor dem 19. Mai 2021 bereits Server mit `EndpointType=VPC_ENDPOINT` deinem AWS Konto erstellt hast, bist du davon nicht betroffen. Verwenden Sie nach diesem Datum `EndpointType =VPC`.
> Weitere Informationen finden Sie unter Einstellung der Verwendung von VPC_ENDPOINT.

## Inhalt

AddressAllocationIds

Eine Liste der Adressenzuweisungs-IDs, die erforderlich sind, um eine Elastic IP-Adresse an den Endpunkt Ihres Servers anzuhängen.

Eine Adresszuweisungs-ID entspricht der Zuweisungs-ID einer Elastic IP-Adresse. Dieser Wert kann aus dem `allocationId` Feld des Amazon EC2 EC2-Adressdatentyps abgerufen werden. Eine Möglichkeit, diesen Wert abzurufen, besteht darin, die DescribeAddressesEC2-API aufzurufen.

Dieser Parameter ist optional. Legen Sie diesen Parameter fest, wenn Sie Ihren VPC-Endpunkt öffentlich zugänglich machen möchten. Einzelheiten finden Sie unter Erstellen eines mit dem Internet verbundenen Endpunkts für Ihren Server.

> **Note**
>
> Diese Eigenschaft kann nur wie folgt festgelegt werden:
>
> - `EndpointType`muss auf gesetzt sein `VPC`
> - Der Transfer Family Family-Server muss offline sein.

- Sie können diesen Parameter nicht für Transfer Family Family-Server festlegen, die das FTP-Protokoll verwenden.
- Der Server muss bereits `SubnetIds` gefüllt sein (`SubnetIds`und `AddressAllocationIds` kann nicht gleichzeitig aktualisiert werden).
- `AddressAllocationIds`darf keine Duplikate enthalten und muss die gleiche Länge wie haben. `SubnetIds` Wenn Sie beispielsweise drei Subnetz-IDs haben, müssen Sie auch drei Adresszuweisungs-IDs angeben.
- Rufen Sie die `UpdateServer` API auf, um diesen Parameter festzulegen oder zu ändern.

Typ: Zeichenfolgen-Array

Erforderlich: Nein

SecurityGroupIds

Eine Liste der Sicherheitsgruppen-IDs, die zum Anhängen an den Endpunkt Ihres Servers verfügbar sind.

> **Note**
>
> Diese Eigenschaft kann nur festgelegt werden, wenn `EndpointType` auf `VPC` gesetzt ist. Sie können die `SecurityGroupIds` Eigenschaft in der UpdateServerAPI nur bearbeiten, wenn Sie die Option `EndpointType` von `PUBLIC` oder `VPC_ENDPOINT` nach ändern`VPC`. Verwenden Sie die Amazon EC2 ModifyVpcEndpointEC2-API, um Sicherheitsgruppen zu ändern, die dem VPC-Endpunkt Ihres Servers nach der Erstellung zugeordnet sind.

Typ: Zeichenfolgen-Array

Längenbeschränkungen: Mindestlänge von 11. Maximale Länge von 20.

Pattern: `sg-[0-9a-f]{8,17}`

Erforderlich: Nein

SubnetIds

Eine Liste der Subnetz-IDs, die zum Hosten des Server-Endpunkts in Ihrer VPC erforderlich sind.

> **Note**
>
> Diese Eigenschaft kann nur festgelegt werden, wenn `EndpointType` auf `VPC` gesetzt ist.

Typ: Zeichenfolgen-Array

Erforderlich: Nein

VpcEndpointId

Die ID des VPC-Endpunkts.

> **Note**
>
> Diese Eigenschaft kann nur festgelegt werden, wenn `EndpointType` auf `VPC_ENDPOINT` gesetzt ist.
> Weitere Informationen finden Sie unter Einstellung der Verwendung von VPC_ENDPOINT.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 22.

Pattern: `vpce-[0-9a-f]{17}`

Erforderlich: Nein

VpcId

Die VPC-ID der VPC, in der der Endpunkt eines Servers gehostet wird.

> **Note**
>
> Diese Eigenschaft kann nur festgelegt werden, wenn `EndpointType` auf `VPC` gesetzt ist.

Typ: Zeichenfolge

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# ExecutionError

Gibt die Fehlermeldung und den Typ für einen Fehler an, der während der Ausführung des Workflows auftritt.

## Inhalt

Message

Gibt die beschreibende Meldung an, die dem `ErrorType` entspricht.

Typ: Zeichenfolge

Erforderlich: Ja

Type

Gibt den Fehlertyp an.

- `ALREADY_EXISTS`: tritt bei einem Kopierschritt auf, wenn die Option Überschreiben nicht ausgewählt ist und am Zielort bereits eine Datei mit demselben Namen existiert.
- `BAD_REQUEST`: eine allgemeine schlechte Anfrage: Ein Schritt, der versucht, eine EFS-Datei zu taggen`BAD_REQUEST`, kehrt zurück, da nur S3-Dateien markiert werden können.
- `CUSTOM_STEP_FAILED`: tritt auf, wenn der benutzerdefinierte Schritt einen Rückruf bereitgestellt hat, der auf einen Fehler hinweist.
- `INTERNAL_SERVER_ERROR`: ein Sammelfehler, der aus verschiedenen Gründen auftreten kann.
- `NOT_FOUND`: tritt auf, wenn eine angeforderte Entität, z. B. eine Quelldatei für einen Kopierschritt, nicht existiert.
- `PERMISSION_DENIED`: tritt auf, wenn Ihre Richtlinie nicht die richtigen Berechtigungen für die Ausführung eines oder mehrerer Schritte im Workflow enthält.
- `TIMEOUT`: tritt auf, wenn bei der Ausführung das Timeout überschritten wird.

  > **Note**
  >
  > Sie können den Wert `TimeoutSeconds` für einen benutzerdefinierten Schritt zwischen 1 Sekunde und 1800 Sekunden (30 Minuten) festlegen.

- `THROTTLED`: tritt auf, wenn Sie die neue Ausführungsrate von einem Workflow pro Sekunde überschreiten.

Typ: Zeichenfolge

Zulässige Werte: `PERMISSION_DENIED | CUSTOM_STEP_FAILED | THROTTLED | ALREADY_EXISTS | NOT_FOUND | BAD_REQUEST | TIMEOUT | INTERNAL_SERVER_ERROR`

Erforderlich: Ja

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# ExecutionResults

Gibt die Schritte im Workflow sowie die Schritte an, die ausgeführt werden müssen, falls während der Workflow-Ausführung Fehler auftreten.

## Inhalt

OnExceptionSteps

Gibt die Schritte (Aktionen) an, die ausgeführt werden sollen, wenn bei der Ausführung des Workflows Fehler auftreten.

Typ: Array von ExecutionStepResult-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Die maximale Anzahl beträgt 50 Elemente.

Erforderlich: Nein

Steps

Gibt die Details für die Schritte an, die im angegebenen Workflow enthalten sind.

Typ: Array von ExecutionStepResult-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Die maximale Anzahl beträgt 50 Elemente.

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# ExecutionStepResult

Gibt die folgenden Details für den Schritt an: Fehler (falls vorhanden), Ausgaben (falls vorhanden) und Schritttyp.

## Inhalt

Error

Gibt die Details für einen Fehler an, falls er während der Ausführung des angegebenen Workflow-Schritts aufgetreten ist.

Typ: ExecutionError Objekt

Erforderlich: Nein

Outputs

Die Werte für das Schlüssel/Wert-Paar, das als Tag auf die Datei angewendet wurde. Gilt nur, wenn der Schritttyp ist. `TAG`

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Die maximale Länge beträgt 65536.

Erforderlich: Nein

StepType

Einer der verfügbaren Schritttypen.

- **`COPY`** – Die Datei an einen anderen Ort kopieren.
- **`CUSTOM`**- Führen Sie einen benutzerdefinierten Schritt mit einem AWS Lambda Funktionsziel aus.
- **`DECRYPT`** – Eine Datei entschlüsseln, die vor dem Hochladen verschlüsselt wurde.
- **`DELETE`** – Die Datei löschen.
- **`TAG`** – Der Datei ein Tag hinzufügen.

Typ: Zeichenfolge

Zulässige Werte: `COPY | CUSTOM | TAG | DELETE | DECRYPT`

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# FileLocation

Gibt die Amazon S3- oder EFS-Dateidetails an, die in dem Schritt verwendet werden sollen.

## Inhalt

EfsFileLocation

Gibt den Amazon EFS-Bezeichner und den Pfad für die verwendete Datei an.

Typ: EfsFileLocation Objekt

Erforderlich: Nein

S3FileLocation

Gibt die S3-Details für die verwendete Datei an, z. B. Bucket, ETag usw.

Typ: S3FileLocation Objekt

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# HomeDirectoryMapEntry

Stellt ein Objekt dar, das Einträge und Ziele für `HomeDirectoryMappings` enthält.

Das Folgende ist ein `Entry` `Target` Und-Pair-Beispiel für`chroot`.

```
[ { "Entry": "/", "Target": "/bucket_name/home/mydirectory" } ]
```

## Inhalt

Entry

Stellt einen Eintrag für `HomeDirectoryMappings` dar.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 1024 Zeichen.

Pattern: `/.*`

Erforderlich: Ja

Target

Stellt das Zuweisungsziel dar, das in einem `HomeDirectoryMapEntry` verwendet wird.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 1024 Zeichen.

Pattern: `/.*`

Erforderlich: Ja

Type

Gibt den Zuordnungstyp an. Stellen Sie den Typ auf ein, `FILE` wenn die Zuordnung auf eine Datei verweisen soll oder `DIRECTORY` wenn das Verzeichnis auf ein Verzeichnis verweisen soll.

> **Note**
>
> Standardmäßig haben Basisverzeichniszuordnungen ein „`Type`von“, `DIRECTORY` wenn Sie einen Transfer Family Family-Server erstellen. Sie müssten den Wert explizit `Type` auf festlegen, `FILE` wenn eine Zuordnung ein Dateiziel haben soll.

Typ: Zeichenfolge

Zulässige Werte: `FILE | DIRECTORY`

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# IdentityProviderDetails

Gibt Informationen zur Art der Benutzerauthentifizierung zurück, die für die Benutzer eines Servers mit aktiviertem Dateiübertragungsprotokoll verwendet wird. Ein Server kann nur über eine Authentifizierungsmethode verfügen.

## Inhalt

DirectoryId

Die ID des AWS Directory Service Verzeichnisses, das Sie als Identitätsanbieter verwenden möchten.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 12.

Pattern: `d-[0-9a-f]{10}`

Erforderlich: Nein

Function

Der ARN für eine Lambda-Funktion, die für den Identity Provider verwendet werden soll.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 1. Höchstlänge = 170 Zeichen.

Pattern: `arn:[a-z-]+:lambda:.*`

Erforderlich: Nein

InvocationRole

Dieser Parameter ist nur anwendbar, wenn Sie `IdentityProviderType` es sind`API_GATEWAY`. Gibt den Typ von `InvocationRole` an, der zur Authentifizierung des Benutzerkontos verwendet wird.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 2048 Zeichen.

Pattern: `arn:.*role/\S+`

Erforderlich: Nein

SftpAuthenticationMethods

Für SFTP-fähige Server und nur für benutzerdefinierte Identitätsanbieter können Sie angeben, ob Sie sich mit einem Passwort, einem SSH-Schlüsselpaar oder beidem authentifizieren möchten.

- `PASSWORD`- Benutzer müssen ihr Passwort angeben, um eine Verbindung herzustellen.
- `PUBLIC_KEY`- Benutzer müssen ihren privaten Schlüssel angeben, um eine Verbindung herzustellen.
- `PUBLIC_KEY_OR_PASSWORD`- Benutzer können sich entweder mit ihrem Passwort oder ihrem Schlüssel authentifizieren. Dies ist der Standardwert.
- `PUBLIC_KEY_AND_PASSWORD`- Benutzer müssen sowohl ihren privaten Schlüssel als auch ihr Passwort angeben, um eine Verbindung herzustellen. Der Server überprüft zuerst den Schlüssel. Wenn der Schlüssel gültig ist, fordert das System dann zur Eingabe eines Kennworts auf. Wenn der angegebene private Schlüssel nicht mit dem gespeicherten öffentlichen Schlüssel übereinstimmt, schlägt die Authentifizierung fehl.

Typ: Zeichenfolge

Zulässige Werte: `PASSWORD | PUBLIC_KEY | PUBLIC_KEY_OR_PASSWORD | PUBLIC_KEY_AND_PASSWORD`

Erforderlich: Nein

Url

Stellt den Speicherort des Service-Endpunkts bereit, der zur Authentifizierung von Benutzern verwendet wird.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 255 Zeichen.

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++

- AWS SDK for Java V2
- AWS SDK for Ruby V3

# InputFileLocation

Gibt den Speicherort für die Datei an, die verarbeitet wird.

## Inhalt

EfsFileLocation

Gibt die Details für die Amazon Elastic File System (Amazon EFS) -Datei an, die entschlüsselt wird.

Typ: EfsFileLocation Objekt

Erforderlich: Nein

S3FileLocation

Gibt die Details für die Amazon S3 S3-Datei an, die kopiert oder entschlüsselt wird.

Typ: S3InputFileLocation Objekt

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# ListedAccess

Listet die Eigenschaften für einen oder mehrere angegebene zugehörige Zugriffe auf.

## Inhalt

ExternalId

Eine eindeutige Kennung, die zur Identifizierung bestimmter Gruppen in Ihrem Verzeichnis erforderlich ist. Die Benutzer der Gruppe, die Sie zuordnen, haben über die aktivierten Protokolle Zugriff auf Ihre Amazon S3- oder Amazon EFS-Ressourcen AWS Transfer Family. Wenn Sie den Gruppennamen kennen, können Sie die SID-Werte anzeigen, indem Sie den folgenden Befehl unter Windows ausführen PowerShell.

```
Get-ADGroup -Filter {samAccountName -like "YourGroupName*"} -Properties
* | Select SamAccountName,ObjectSid
```

Ersetzen Sie diesen Befehl YourGroupNamedurch den Namen Ihrer Active Directory-Gruppe.

Der reguläre Ausdruck, der zur Überprüfung dieses Parameters verwendet wird, ist eine Zeichenfolge, die aus alphanumerischen Groß- und Kleinbuchstaben ohne Leerzeichen besteht. Sie können auch Unterstriche oder eines der folgenden Zeichen verwenden: =, . @: /-

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 256 Zeichen.

Pattern: `S-1-[\d-]+`

Erforderlich: Nein

HomeDirectory

Das Zielverzeichnis (Ordner) für einen Benutzer bei der Serveranmeldung mithilfe des Client.

Ein Beispiel für `HomeDirectory` ist `/bucket_name/home/mydirectory`.

> **Note**
>
> Der `HomeDirectory`-Parameter wird nur verwendet, wenn `HomeDirectoryType` auf `PATH` gesetzt ist.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 1024 Zeichen.

Pattern: `(|/.*)`

Erforderlich: Nein

HomeDirectoryType

Die Art des Zielverzeichnisses (Ordners), das das Home-Verzeichnis Ihrer Benutzer sein soll, wenn sie sich beim Server anmelden. Wenn Sie es auf einstellen`PATH`, sieht der Benutzer den absoluten Amazon S3-Bucket- oder Amazon EFS-Pfad so, wie er in seinen File Transfer Protocol-Clients ist. Wenn Sie es auf einstellen`LOGICAL`, müssen Sie Zuordnungen dafür angeben, wie Sie Amazon S3- oder Amazon EFS-Pfade `HomeDirectoryMappings` für Ihre Benutzer sichtbar machen möchten.

> **Note**
>
> `HomeDirectoryType`Ist dies der `LOGICAL` Fall, müssen Sie mithilfe des Parameters Zuordnungen angeben. `HomeDirectoryMappings` Ist dies hingegen der Fall, `HomeDirectoryType` geben Sie mithilfe des Parameters einen absoluten Pfad an`HomeDirectory`. `PATH` Sie können nicht beides `HomeDirectory` und `HomeDirectoryMappings` in Ihrer Vorlage haben.

Typ: Zeichenfolge

Zulässige Werte: `PATH | LOGICAL`

Erforderlich: Nein

Role

Der Amazon-Ressourcenname (ARN) der Rolle AWS Identity and Access Management (IAM), die den Zugriff Ihrer Benutzer auf Ihren Amazon S3-Bucket oder Ihr Amazon EFS-Dateisystem steuert. Die mit dieser Rolle verbundenen Richtlinien bestimmen die Zugriffsebene, die Sie Ihren Benutzern beim Übertragen von Dateien in und aus Ihrem Amazon-S3-Bucket oder Amazon-EFS-Dateisystem bereitstellen möchten. Die IAM-Rolle sollte außerdem eine Vertrauensstellung enthalten, mit der der Server Zugriff auf Ihre Ressourcen erhält, wenn er die Übertragungsanfragen Ihres Benutzers bearbeitet.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 2048 Zeichen.

Pattern: `arn:.*role/\S+`

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# ListedAgreement

Beschreibt die Eigenschaften einer Vereinbarung.

## Inhalt

AgreementId

Ein eindeutiger Bezeichner für die Vereinbarung. Diese Kennung wird zurückgegeben, wenn Sie eine Vereinbarung erstellen.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `a-([0-9a-f]{17})`

Erforderlich: Nein

Arn

Der Amazon-Ressourcenname (ARN) der angegebenen Vereinbarung.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 1600 Zeichen.

Pattern: `arn:\S+`

Erforderlich: Nein

Description

Die aktuelle Beschreibung der Vereinbarung. Sie können sie ändern, indem Sie den `UpdateAgreement` Vorgang aufrufen und eine neue Beschreibung angeben.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 1. Höchstlänge = 200 Zeichen.

Pattern: `[\p{Graph}]+`

Erforderlich: Nein

LocalProfileId

Eine eindeutige Kennung für das lokale AS2-Profil.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `p-([0-9a-f]{17})`

Erforderlich: Nein

PartnerProfileId

Eine eindeutige Kennung für das Partnerprofil.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `p-([0-9a-f]{17})`

Erforderlich: Nein

ServerId

Die eindeutige Kennung für die Vereinbarung.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Erforderlich: Nein

Status

Die Vereinbarung kann entweder `ACTIVE` oder sein`INACTIVE`.

Typ: Zeichenfolge

Zulässige Werte: `ACTIVE | INACTIVE`

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# ListedCertificate

Beschreibt die Eigenschaften eines Zertifikats.

## Inhalt

ActiveDate

Ein optionales Datum, das angibt, wann das Zertifikat aktiv wird.

Typ: Zeitstempel

Erforderlich: Nein

Arn

Der Amazon-Ressourcenname (ARN) des angegebenen Zertifikats.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 1600 Zeichen.

Pattern: `arn:\S+`

Erforderlich: Nein

CertificateId

Ein Array von Kennungen für die importierten Zertifikate. Sie verwenden diese Kennung für die Arbeit mit Profilen und Partnerprofilen.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 22.

Pattern: `cert-([0-9a-f]{17})`

Erforderlich: Nein

Description

Der Name oder die Kurzbeschreibung, die zur Identifizierung des Zertifikats verwendet wird.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 1. Höchstlänge = 200 Zeichen.

Pattern: `[\p{Graph}]+`

Erforderlich: Nein

InactiveDate

Ein optionales Datum, das angibt, wann das Zertifikat inaktiv wird.

Typ: Zeitstempel

Erforderlich: Nein

Status

Das Zertifikat kann entweder `ACTIVE`, `PENDING_ROTATION` oder `INACTIVE` sein. `PENDING_ROTATION`bedeutet, dass dieses Zertifikat das aktuelle Zertifikat ersetzt, wenn letzteres abläuft.

Typ: Zeichenfolge

Zulässige Werte: `ACTIVE | PENDING_ROTATION | INACTIVE`

Erforderlich: Nein

Type

Der Typ für das Zertifikat. Wurde ein privater Schlüssel für das Zertifikat angegeben, ist sein Typ `CERTIFICATE_WITH_PRIVATE_KEY`. Ist kein privater Schlüssel vorhanden, ist der Typ `CERTIFICATE`.

Typ: Zeichenfolge

Zulässige Werte: `CERTIFICATE | CERTIFICATE_WITH_PRIVATE_KEY`

Erforderlich: Nein

Usage

Gibt an, wie dieses Zertifikat verwendet wird. Es kann auf folgende Weise verwendet werden:

- `SIGNING`: Zum Signieren von AS2-Nachrichten
- `ENCRYPTION`: Zum Verschlüsseln von AS2-Nachrichten
- `TLS`: Zur Sicherung von AS2-Kommunikation, die über HTTPS gesendet wird

Typ: Zeichenfolge

Zulässige Werte: `SIGNING | ENCRYPTION`

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# ListedConnector

Gibt Details des angegebenen Connectors zurück.

## Inhalt

Arn

Der Amazon-Ressourcenname (ARN) des angegebenen Connectors.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 1600 Zeichen.

Pattern: `arn:\S+`

Erforderlich: Nein

ConnectorId

Die eindeutige Kennung für den Konnektor.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `c-([0-9a-f]{17})`

Erforderlich: Nein

Url

Die URL des AS2- oder SFTP-Endpunkts des Partners.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 255 Zeichen.

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# ListedExecution

Gibt Eigenschaften der angegebenen Ausführung zurück.

## Inhalt

ExecutionId

Ein eindeutiger Bezeichner für die Ausführung eines Workflows.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 36.

Pattern: `[0-9a-fA-F]{8}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{4}\-[0-9a-fA-F]{12}`

Erforderlich: Nein

InitialFileLocation

Eine Struktur, die den Speicherort der Amazon S3- oder EFS-Datei beschreibt. Dies ist der Dateispeicherort, an dem die Ausführung beginnt: Wenn die Datei kopiert wird, ist dies der ursprüngliche Dateispeicherort (und nicht der Zielspeicherort).

Typ: FileLocation Objekt

Erforderlich: Nein

ServiceMetadata

Ein Container-Objekt für die Sitzungsdetails, die einem Workflow zugeordnet sind.

Typ: ServiceMetadata Objekt

Erforderlich: Nein

Status

Der Status ist „Ausführung“. Kann in Bearbeitung sein, abgeschlossen sein, eine Ausnahme aufgetreten sein oder die Ausnahme wird behandelt.

Typ: Zeichenfolge

Zulässige Werte: `IN_PROGRESS | COMPLETED | EXCEPTION | HANDLING_EXCEPTION`

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# ListedHostKey

Gibt Eigenschaften des angegebenen Host-Schlüssels zurück.

## Inhalt

Arn

Der eindeutige Amazon-Ressourcenname (ARN) des Hostschlüssels.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 1600 Zeichen.

Pattern: `arn:\S+`

Erforderlich: Ja

DateImported

Das Datum, an dem der Hostschlüssel zum Server hinzugefügt wurde.

Typ: Zeitstempel

Erforderlich: Nein

Description

Die aktuelle Beschreibung für den Hostschlüssel. Sie können sie ändern, indem Sie den `UpdateHostKey` Vorgang aufrufen und eine neue Beschreibung angeben.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Höchstlänge = 200 Zeichen.

Pattern: `[\p{Print}]*`

Erforderlich: Nein

Fingerprint

Der Fingerabdruck des öffentlichen Schlüssels, bei dem es sich um eine kurze Bytefolge handelt, die zur Identifizierung des längeren öffentlichen Schlüssels verwendet wird.

Typ: Zeichenfolge

Erforderlich: Nein

HostKeyId

Eine eindeutige Kennung für den Host-Schlüssel.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 25.

Pattern: `hostkey-[0-9a-f]{17}`

Erforderlich: Nein

Type

Der Verschlüsselungsalgorithmus, der für den Hostschlüssel verwendet wird. Der `Type` Parameter wird mit einem der folgenden Werte angegeben:

- `ssh-rsa`
- `ssh-ed25519`
- `ecdsa-sha2-nistp256`
- `ecdsa-sha2-nistp384`
- `ecdsa-sha2-nistp521`

Typ: Zeichenfolge

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# ListedProfile

Gibt die Eigenschaften des angegebenen Profils zurück.

## Inhalt

Arn

Der Amazon-Ressourcenname (ARN) des angegebenen Profils.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 1600 Zeichen.

Pattern: `arn:\S+`

Erforderlich: Nein

As2Id

Die `As2Id` ist der AS2-Name, wie in RFC 4130 definiert. Bei eingehenden Übertragungen ist dies der `AS2-From`-Header für die vom Partner gesendeten AS2-Nachrichten. Bei ausgehenden Connectors ist dies der `AS2-To`-Header für die AS2-Nachrichten, die mithilfe der `StartFileTransfer`-API-Operation an den Partner gesendet werden. Diese ID darf keine Leerzeichen enthalten.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 128 Zeichen.

Pattern: `[\p{Print}\s]*`

Erforderlich: Nein

ProfileId

Eine eindeutige Kennung für das lokale AS2-Profil oder das AS2-Profil eines Partners.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `p-([0-9a-f]{17})`

Erforderlich: Nein

ProfileType

Gibt an, ob nur `LOCAL`- oder nur `PARTNER`-Typprofile aufgelistet werden sollen. Sind diese nicht in der Anforderung nicht, listet der Befehl alle Profilarten auf.

Typ: Zeichenfolge

Zulässige Werte: `LOCAL | PARTNER`

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# ListedServer

Gibt Eigenschaften eines angegebenen Servers mit aktiviertem Dateiübertragungsprotokoll zurück.

## Inhalt

Arn

Gibt den eindeutigen Amazon-Ressourcennamen (ARN) für einen Server an, der aufgelistet werden soll.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 1600 Zeichen.

Pattern: `arn:\S+`

Erforderlich: Ja

Domain

Gibt die Domain des Speichersystems an, das für Dateiübertragungen verwendet wird. Es sind zwei Domänen verfügbar: Amazon Simple Storage Service (Amazon S3) und Amazon Elastic File System (Amazon EFS). Der Standardwert ist S3.

Typ: Zeichenfolge

Zulässige Werte: `S3 | EFS`

Erforderlich: Nein

EndpointType

Gibt den Typ des VPC-Endpunkts an, mit dem Ihr Server verbunden ist. Wenn Ihr Server mit einem VPC-Endpunkt verbunden ist, ist Ihr Server nicht über das öffentliche Internet zugänglich.

Typ: Zeichenfolge

Zulässige Werte: `PUBLIC | VPC | VPC_ENDPOINT`

Erforderlich: Nein

IdentityProviderType

Das Authentifizierungsverfahren für einen Server. Der Standardwert ist`SERVICE_MANAGED`, der es Ihnen ermöglicht, Benutzeranmeldeinformationen innerhalb des AWS Transfer Family Dienstes zu speichern und darauf zuzugreifen.

Wird verwendet`AWS_DIRECTORY_SERVICE`, um Zugriff auf Active Directory-Gruppen in AWS Directory Service for Microsoft Active Directory oder Microsoft Active Directory in Ihrer lokalen Umgebung oder bei der AWS Verwendung von AD Connector bereitzustellen. Für diese Option ist es auch erforderlich, dass Sie mithilfe des Parameters `IdentityProviderDetails` eine Directory-ID angeben.

Verwenden Sie den `API_GATEWAY`-Wert für die Integration eines Identitätsanbieters Ihrer Wahl. Die `API_GATEWAY`-Einstellung erfordert, dass Sie mithilfe des Parameters `IdentityProviderDetails` die URL eines Amazon-API-Gateway-Endpunkts angeben, der zur Authentifizierung aufgerufen wird.

Verwenden Sie den `AWS_LAMBDA` Wert, um eine AWS Lambda Funktion direkt als Identitätsanbieter zu verwenden. Wenn Sie diesen Wert wählen, müssen Sie den ARN für die Lambda-Funktion im `Function` Parameter für den `IdentityProviderDetails` Datentyp angeben.

Typ: Zeichenfolge

Zulässige Werte: `SERVICE_MANAGED | API_GATEWAY | AWS_DIRECTORY_SERVICE | AWS_LAMBDA`

Erforderlich: Nein

LoggingRole

Der Amazon-Ressourcenname (ARN) der AWS Identity and Access Management (IAM) -Rolle, der es einem Server ermöglicht, die CloudWatch Amazon-Protokollierung für Amazon S3 oder Amazon EFSEvents zu aktivieren. Wenn diese Option aktiviert ist, können Sie Benutzeraktivitäten in Ihren Protokollen einsehen. CloudWatch

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 2048 Zeichen.

Pattern: `arn:.*role/\S+`

Erforderlich: Nein

ServerId

Gibt die eindeutige vom System zugewiesene Kennung für die aufgelisteten Server an.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Erforderlich: Nein

State

Der Zustand des Servers, der beschrieben wurde. Der Wert von `ONLINE` gibt an, dass der Server Aufträge annehmen und Dateien übertragen kann. `State`Der Wert von `OFFLINE` bedeutet, dass der Server keine Dateiübertragungsvorgänge ausführen kann.

Der Status von `STARTING` und `STOPPING` gibt an, dass sich der Server in einem Zwischenzustand befindet, entweder nicht vollständig antworten kann oder nicht vollständig offline ist. Die Werte von `START_FAILED` oder `STOP_FAILED` können auf einen Fehler hinweisen.

Typ: Zeichenfolge

Zulässige Werte: `OFFLINE | ONLINE | STARTING | STOPPING | START_FAILED | STOP_FAILED`

Erforderlich: Nein

UserCount

Gibt die Anzahl der Benutzer an, die einem Server zugewiesen sind, den Sie mit dem angegeben haben`ServerId`.

Typ: Ganzzahl

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# ListedUser

Gibt Eigenschaften des von Ihnen angegebenen Benutzers zurück.

## Inhalt

Arn

Stellt den eindeutigen Amazon-Ressourcennamen (ARN) für den Benutzer bereit, über den Sie mehr erfahren möchten.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 1600 Zeichen.

Pattern: `arn:\S+`

Erforderlich: Ja

HomeDirectory

Das Zielverzeichnis (Ordner) für einen Benutzer bei der Serveranmeldung mithilfe des Client.

Ein Beispiel für `HomeDirectory` ist `/bucket_name/home/mydirectory`.

> **Note**
>
> Der `HomeDirectory`-Parameter wird nur verwendet, wenn `HomeDirectoryType` auf `PATH` gesetzt ist.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 1024 Zeichen.

Pattern: `(|/.*)`

Erforderlich: Nein

HomeDirectoryType

Die Art des Zielverzeichnisses (Ordners), das das Home-Verzeichnis Ihrer Benutzer sein soll, wenn sie sich beim Server anmelden. Wenn Sie es auf einstellen`PATH`, sieht der Benutzer den absoluten Amazon S3-Bucket- oder Amazon EFS-Pfad so, wie er in seinen File Transfer Protocol-

Clients ist. Wenn Sie es auf einstellen`LOGICAL`, müssen Sie Zuordnungen dafür angeben, wie Sie Amazon S3- oder Amazon EFS-Pfade `HomeDirectoryMappings` für Ihre Benutzer sichtbar machen möchten.

> **Note**
>
> `HomeDirectoryType`Ist dies der `LOGICAL` Fall, müssen Sie mithilfe des Parameters Zuordnungen angeben. `HomeDirectoryMappings` Ist dies hingegen der Fall, `HomeDirectoryType` geben Sie mithilfe des Parameters einen absoluten Pfad an`HomeDirectory`. `PATH` Sie können nicht beides `HomeDirectory` und `HomeDirectoryMappings` in Ihrer Vorlage haben.

Typ: Zeichenfolge

Zulässige Werte: `PATH | LOGICAL`

Erforderlich: Nein

Role

Der Amazon-Ressourcenname (ARN) der Rolle AWS Identity and Access Management (IAM), die den Zugriff Ihrer Benutzer auf Ihren Amazon S3-Bucket oder Ihr Amazon EFS-Dateisystem steuert. Die mit dieser Rolle verbundenen Richtlinien bestimmen die Zugriffsebene, die Sie Ihren Benutzern beim Übertragen von Dateien in und aus Ihrem Amazon-S3-Bucket oder Amazon-EFS-Dateisystem bereitstellen möchten. Die IAM-Rolle sollte außerdem eine Vertrauensstellung enthalten, mit der der Server Zugriff auf Ihre Ressourcen erhält, wenn er die Übertragungsanfragen Ihres Benutzers bearbeitet.

> **Note**
>
> Die IAM-Rolle, die den Zugriff Ihrer Benutzer auf Ihren Amazon S3 S3-Bucket für Server mit `Domain=S3` oder Ihr EFS-Dateisystem für Server mit `Domain=EFS` steuert.
> Die mit dieser Rolle verknüpften Richtlinien bestimmen die Zugriffsebene, die Sie Ihren Benutzern beim Übertragen von Dateien in und aus Ihren S3-Buckets oder EFS-Dateisystemen gewähren möchten.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 2048 Zeichen.

Pattern: `arn:.*role/\S+`

Erforderlich: Nein

SshPublicKeyCount

Gibt die Anzahl der öffentlichen SSH-Schlüssel an, die für den von Ihnen angegebenen Benutzer gespeichert sind.

Typ: Ganzzahl

Erforderlich: Nein

UserName

Gibt den Namen des Benutzers an, dessen ARN angegeben wurde. Benutzernamen werden für Authentifizierungszwecke verwendet.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 3. Maximale Länge beträgt 100 Zeichen.

Pattern: `[\w][\w@.-]{2,99}`

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# ListedWorkflow

Enthält den Bezeichner, die Textbeschreibung und den Amazon-Ressourcennamen (ARN) für den Workflow.

## Inhalt

Arn

Gibt den eindeutigen Amazon-Ressourcennamen (ARN) für den Workflow an.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 1600 Zeichen.

Pattern: `arn:\S+`

Erforderlich: Nein

Description

Gibt die Textbeschreibung für den Workflow an.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 256 Zeichen.

Pattern: `[\w- ]*`

Erforderlich: Nein

WorkflowId

Eine eindeutige Kennung für den Workflow.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `w-([a-z0-9]{17})`

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# LoggingConfiguration

Besteht aus der Protokollierungsrolle und dem Namen der Protokollgruppe.

## Inhalt

LoggingRole

Der Amazon-Ressourcenname (ARN) der AWS Identity and Access Management (IAM) -Rolle, der es einem Server ermöglicht, die CloudWatch Amazon-Protokollierung für Amazon S3 oder Amazon EFSEvents zu aktivieren. Wenn diese Option aktiviert ist, können Sie Benutzeraktivitäten in Ihren Protokollen einsehen. CloudWatch

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 2048 Zeichen.

Pattern: `arn:.*role/\S+`

Erforderlich: Nein

LogGroupName

Der Name der CloudWatch Protokollierungsgruppe für den AWS Transfer Family Server, zu dem dieser Workflow gehört.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge von 512.

Pattern: `[\.\-_/#A-Za-z0-9]*`

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2

- AWS SDK for Ruby V3

# PosixProfile

Die vollständige POSIX-Identität, einschließlich Benutzer-ID (`Uid`), Gruppen-ID (`Gid`) und sekundärer Gruppen-IDs (`SecondaryGids`), die den Zugriff Ihrer Benutzer auf Ihre Amazon EFS-Dateisysteme (Elastic File System) steuert. Die POSIX-Berechtigungen, die für Dateien und Verzeichnisse in Ihrem Dateisystem festgelegt sind, bestimmen die Zugriffsebene, die Ihre Benutzer beim Übertragen von Dateien in und aus Ihren Amazon EFS-Dateisystemen erhalten.

## Inhalt

Gid

Die POSIX-Gruppen-ID, die von diesem Benutzer für alle EFS-Vorgänge verwendet wird.

Type: Long

Gültiger Bereich: Mindestwert 0. Maximaler Wert von 4294967295.

Erforderlich: Ja

Uid

Die POSIX-Benutzer-ID, die von diesem Benutzer für alle EFS-Vorgänge verwendet wird.

Type: Long

Gültiger Bereich: Mindestwert 0. Maximaler Wert von 4294967295.

Erforderlich: Ja

SecondaryGids

Die sekundären POSIX-Gruppen-IDs, die von diesem Benutzer für alle EFS-Vorgänge verwendet werden.

Typ: Array von Longs

Array-Mitglieder: Die Mindestanzahl beträgt 0 Elemente. Die maximale Anzahl beträgt 16 Elemente.

Gültiger Bereich: Mindestwert 0. Maximaler Wert von 4294967295.

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# ProtocolDetails

Protokolleinstellungen, die für Ihren Server konfiguriert sind.

## Inhalt

As2Transports

Gibt an, wie AS2-Nachrichten transportiert werden sollen. Derzeit wird nur HTTP unterstützt.

Typ: Zeichenfolgen-Array

Array-Mitglieder: Feste Anzahl von 1 Element.

Zulässige Werte: `HTTP`

Erforderlich: Nein

PassiveIp

Zeigt den passiven Modus für FTP- und FTPS-Protokolle an. Geben Sie eine einzelne IPv4-Adresse ein, z. B. die öffentliche IP-Adresse einer Firewall, eines Routers oder eines Load Balancers. Beispiel:

```
aws transfer update-server --protocol-details PassiveIp=0.0.0.0
```

Ersetzen Sie `0.0.0.0` im obigen Beispiel durch die tatsächliche IP-Adresse, die Sie verwenden möchten.

> **Note**
>
> Wenn Sie den `PassiveIp`-Wert ändern, müssen Sie Ihren Transfer-Family-Server stoppen und dann neu starten, damit die Änderung wirksam wird. Einzelheiten zur Verwendung des passiven Modus (PASV) in einer NAT-Umgebung finden Sie unter Konfiguration Ihres FTPS-Servers hinter einer Firewall oder NAT mit. AWS Transfer Family

Spezielle Werte

`AUTO` und `0.0.0.0` sind spezielle Werte für den `PassiveIp`-Parameter. Der Wert `PassiveIp=AUTO` ist standardmäßig FTP- und FTPS-Servern zugewiesen. In diesem Fall

antwortet der Server automatisch mit einer der Endpunkt-IPs innerhalb der PASV-Antwort. `PassiveIp=0.0.0.0` wird in spezielleren Fällen verwendet. Wenn Sie beispielsweise eine High Availability (HA) Network Load Balancer (NLB)-Umgebung haben, in der Sie über 3 Subnetze verfügen, können Sie mit dem `PassiveIp`-Parameter eine einzige IP-Adresse angeben. Dies verringert die Effektivität der Hochverfügbarkeit. In diesem Fall können Sie `PassiveIp=0.0.0.0` angeben. Dadurch wird der Client angewiesen, dieselbe IP-Adresse wie die Kontrollverbindung zu verwenden und alle AZs für seine Verbindungen zu verwenden. Beachten Sie jedoch, dass nicht alle FTP-Clients die `PassiveIp=0.0.0.0` Antwort unterstützen. FileZilla und WinSCP unterstützt es. Wenn Sie andere Clients verwenden, überprüfen Sie, ob Ihr Client die `PassiveIp=0.0.0.0`-Antwort unterstützt.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 15 Zeichen.

Erforderlich: Nein

SetStatOption

Verwenden Sie die `SetStatOption`, um den Fehler zu ignorieren, der generiert wird, wenn der Client versucht, `SETSTAT` für eine Datei zu verwenden, die Sie in einen S3-Bucket hochladen.

Einige SFTP-Dateiübertragungs-Clients können versuchen, die Attribute von Remote-Dateien, einschließlich Zeitstempel und Berechtigungen, mithilfe von Befehlen wie `SETSTAT` beim Hochladen der Datei zu ändern. Diese Befehle sind jedoch nicht mit Objekt-Speichersystemen wie Amazon S3 kompatibel. Aufgrund dieser Inkompatibilität können Datei-Uploads von diesen Clients zu Fehlern führen, selbst wenn die Datei anderweitig erfolgreich hochgeladen wurde.

Setzen Sie den Wert auf `ENABLE_NO_OP`, damit der Transfer Family-Server den `SETSTAT`-Befehl ignoriert und Dateien hochlädt, ohne Änderungen an Ihrem SFTP-Client vornehmen zu müssen. Die `SetStatOption` `ENABLE_NO_OP` Einstellung ignoriert zwar den Fehler, generiert aber einen Protokolleintrag in Amazon CloudWatch Logs, sodass Sie feststellen können, wann der Client einen `SETSTAT` Anruf tätigt.

> **Note**
>
> Wenn Sie den ursprünglichen Zeitstempel für Ihre Datei beibehalten und andere Dateiattribute mit `SETSTAT` ändern möchten, können Sie Amazon EFS als Backend-Speicher mit Transfer Family verwenden.

Typ: Zeichenfolge

Zulässige Werte: `DEFAULT | ENABLE_NO_OP`

Erforderlich: Nein

TlsSessionResumptionMode

Eine Eigenschaft, die mit Servern der Transfer Family verwendet wird, die das FTPS-Protokoll verwenden. Die TLS-Sitzungs-Wiederaufnahme bietet einen Mechanismus zum Fortsetzen oder Freigeben eines ausgehandelten geheimen Schlüssels zwischen der Steuerungs- und der Datenverbindung für eine FTPS-Sitzung. `TlsSessionResumptionMode` bestimmt über eine eindeutige Sitzungs-ID, ob der Server kürzlich ausgehandelte Sitzungen wieder aufnimmt oder nicht. Diese Eigenschaft ist während `CreateServer`- und `UpdateServer`-Aufrufen verfügbar. Wenn während `CreateServer` kein `TlsSessionResumptionMode`-Wert angegeben wird, wird er standardmäßig auf `ENFORCED` gesetzt.

- `DISABLED`: Der Server verarbeitet keine Client-Anforderungen zur Wiederaufnahme der TLS-Sitzung und erstellt für jede Anforderung eine neue TLS-Sitzung.
- `ENABLED`: Der Server verarbeitet und akzeptiert Clients, die die Wiederaufnahme der TLS-Sitzung durchführen. Der Server lehnt keine Client-Datenverbindungen ab, die die Verarbeitung des Clients zur Wiederaufnahme der TLS-Sitzung nicht ausführen.
- `ENFORCED`: Der Server verarbeitet und akzeptiert Clients, die die Wiederaufnahme der TLS-Sitzung durchführen. Der Server lehnt Client-Datenverbindungen ab, die die Verarbeitung des Clients zur Wiederaufnahme der TLS-Sitzung nicht ausführen. Bevor Sie den Wert auf `ENFORCED` festlegen, testen Sie Ihre Kunden.

> **Note**
>
> Nicht alle FTPS-Clients führen eine TLS-Sitzungs-Wiederaufnahme durch. Wenn Sie also die Wiederaufnahme der TLS-Sitzung erzwingen, verhindern Sie jegliche Verbindungen von FTPS-Clients, die die Protokoll-Aushandlung nicht durchführen. Um festzustellen, ob Sie den `ENFORCED`-Wert verwenden können oder nicht, müssen Sie Ihre Clients testen.

Typ: Zeichenfolge

Zulässige Werte: `DISABLED | ENABLED | ENFORCED`

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# S3FileLocation

Gibt die Details für den Dateispeicherort der Datei an, die im Workflow verwendet wird. Gilt nur, wenn Sie S3-Speicher verwenden.

## Inhalt

Bucket

Gibt den S3-Bucket an, der die verwendete Datei enthält.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 3. Maximale Länge beträgt 63 Zeichen.

Pattern: `[a-z0-9][\.\-a-z0-9]{1,61}[a-z0-9]`

Erforderlich: Nein

Etag

Der Entitäts-Tag ist ein Hashwert des Objekts. Das ETag gibt nur Änderungen am Inhalt eines Objekts wieder, nicht an seinen Metadaten.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Die maximale Länge beträgt 65536.

Pattern: `.+`

Erforderlich: Nein

Key

Der Name, der der Datei zugewiesen wurde, als sie in Amazon S3 erstellt wurde. Zum Abrufen des Objekts verwenden Sie den Objektschlüssel.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 1024 Zeichen.

Pattern: `[\P{M}\p{M}]*`

Erforderlich: Nein

VersionId

Gibt die Dateiversion an.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 1. Maximale Länge beträgt 1024 Zeichen.

Pattern: `.+`

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# S3InputFileLocation

Gibt den Speicherort der vom Kunden eingegebenen Amazon S3 S3-Datei an. Wenn es intern verwendet wird`copyStepDetails.DestinationFileLocation`, sollte es das Ziel für die S3-Kopie sein.

Sie müssen den Bucket und den Schlüssel angeben. Der Schlüssel kann entweder einen Pfad oder eine Datei darstellen. Dies hängt davon ab, ob Sie den Schlüsselwert mit dem Schrägstrich (/) beenden oder nicht. Wenn das letzte Zeichen „/“ ist, wird Ihre Datei in den Ordner kopiert, und ihr Name ändert sich nicht. Wenn das letzte Zeichen stattdessen alphanumerisch ist, wird Ihre hochgeladene Datei in den Pfadwert umbenannt. In diesem Fall wird eine Datei mit diesem Namen überschrieben, wenn sie bereits existiert.

Wenn Ihr Pfad beispielsweise lautet`shared-files/bob/`, werden Ihre hochgeladenen Dateien in den Ordner`shared-files/bob/`, kopiert. Wenn Ihr Pfad lautet`shared-files/today`, wird jede hochgeladene Datei in den `shared-files` Ordner kopiert und benannt`today`: Jeder Upload überschreibt die vorherige Version der Bob-Datei.

## Inhalt

Bucket

Gibt den S3-Bucket für die Kundeneingabedatei an.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 3. Maximale Länge beträgt 63 Zeichen.

Pattern: `[a-z0-9][\.\-a-z0-9]{1,61}[a-z0-9]`

Erforderlich: Nein

Key

Der Name, der der Datei zugewiesen wurde, als sie in Amazon S3 erstellt wurde. Zum Abrufen des Objekts verwenden Sie den Objektschlüssel.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 1024 Zeichen.

Pattern: `[\P{M}\p{M}]*`

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# S3StorageOptions

Die Amazon S3 S3-Speicheroptionen, die für Ihren Server konfiguriert sind.

## Inhalt

DirectoryListingOptimization

Gibt an, ob die Leistung für Ihre Amazon S3 S3-Verzeichnisse optimiert ist oder nicht. Diese ist standardmäßig deaktiviert.

Standardmäßig haben Zuordnungen von Home-Verzeichnissen einen `TYPE` Wert von. `DIRECTORY` Wenn Sie diese Option aktivieren, müssten Sie den Wert dann explizit auf setzen, `FILE` wenn Sie möchten `HomeDirectoryMapEntryType`, dass eine Zuordnung ein Dateiziel hat.

Typ: Zeichenfolge

Zulässige Werte: `ENABLED | DISABLED`

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# S3Tag

Gibt das Schlüssel-Wert-Paar an, das einer Datei während der Ausführung eines Tagging-Schritts zugewiesen wird.

## Inhalt

Key

Der Name, der dem von Ihnen erstellten Tag zugewiesen wurde.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 128 Zeichen.

Pattern: `([\p{L}\p{Z}\p{N}_.:/=+\-@]*)`

Erforderlich: Ja

Value

Der Wert, der dem Schlüssel entspricht.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 256 Zeichen.

Pattern: `([\p{L}\p{Z}\p{N}_.:/=+\-@]*)`

Erforderlich: Ja

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# ServiceMetadata

Ein Container-Objekt für die Sitzungsdetails, die einem Workflow zugeordnet sind.

## Inhalt

UserDetails

Die Server-ID (`ServerId`), die Sitzungs-ID (`SessionId`) und der Benutzer (`UserName`) bilden den`UserDetails`.

Typ: UserDetails Objekt

Erforderlich: Ja

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# SftpConnectorConfig

Enthält die Details für ein SFTP-Connector-Objekt. Das Connector-Objekt wird für die Übertragung von Dateien zum und vom SFTP-Server eines Partners verwendet.

> **Note**
>
> Da der `SftpConnectorConfig` Datentyp sowohl für die Erstellung als auch für die Aktualisierung von SFTP-Konnektoren verwendet wird, `UserSecretId` sind seine Parameter `TrustedHostKeys` und Parameter als nicht erforderlich gekennzeichnet. Dies ist etwas irreführend, da sie nicht erforderlich sind, wenn Sie einen vorhandenen SFTP-Connector aktualisieren, sondern erforderlich sind, wenn Sie einen neuen SFTP-Connector erstellen.

## Inhalt

TrustedHostKeys

Der öffentliche Teil des Hostschlüssels oder der Schlüssel, mit denen der externe Server identifiziert wird, zu dem Sie eine Verbindung herstellen. Sie können den `ssh-keyscan` Befehl für den SFTP-Server verwenden, um den erforderlichen Schlüssel abzurufen.

Die drei Standardelemente im SSH-Format für öffentliche Schlüssel sind `<key type><body base64>`, und ein optionales `<comment>` Element mit Leerzeichen zwischen den einzelnen Elementen. Geben Sie nur das `<key type>` und an`<body base64>`: Geben Sie nicht den `<comment>` Teil des Schlüssels ein.

AWS Transfer Family Akzeptiert für den vertrauenswürdigen Host-Schlüssel RSA- und ECDSA-Schlüssel.

- Für RSA-Schlüssel lautet die Zeichenfolge. `<key type>` `ssh-rsa`
- Bei ECSA-Schlüsseln ist die `<key type>` Zeichenfolge entweder, oder `ecdsa-sha2-nistp256` `ecdsa-sha2-nistp384ecdsa-sha2-nistp521`, abhängig von der Größe des von Ihnen generierten Schlüssels.

Führen Sie diesen Befehl aus, um den Hostschlüssel des SFTP-Servers abzurufen, in dem sich Ihr SFTP-Servername befindet. `ftp.host.com`

```
ssh-keyscan ftp.host.com
```

Dadurch wird der öffentliche Hostschlüssel auf die Standardausgabe gedruckt.

```
ftp.host.com ssh-rsa AAAAB3Nza...<long-string-for-public-key
```

Kopieren Sie diese Zeichenfolge und fügen Sie sie in das `TrustedHostKeys` Feld für den `create-connector` Befehl oder in das Feld Vertrauenswürdige Hostschlüssel in der Konsole ein.

Typ: Zeichenfolgen-Array

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Die maximale Anzahl beträgt 10 Elemente.

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 2048 Zeichen.

Erforderlich: Nein

UserSecretId

Der Bezeichner für das Geheimnis (in AWS Secrets Manager), das den privaten Schlüssel, das Passwort oder beides des SFTP-Benutzers enthält. Die Kennung muss der Amazon-Ressourcenname (ARN) des Geheimnisses sein.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge beträgt 1 Zeichen. Maximale Länge beträgt 2048 Zeichen.

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# SshPublicKey

Stellt Informationen über den öffentlichen Secure Shell (SSH) -Schlüssel bereit, der einem Transfer Family Family-Benutzer für den spezifischen Server zugeordnet ist, für den das File Transfer Protocol aktiviert ist (wie durch identifiziert). `ServerId` Die zurückgegebenen Informationen umfassen das Datum, an dem der Schlüssel importiert wurde, den Inhalt des öffentlichen Schlüssels und die ID des öffentlichen Schlüssels. Ein Benutzer kann mehr als einen öffentlichen SSH-Schlüssel, der seinem Benutzernamen zugeordnet ist, auf einem bestimmten Server speichern.

## Inhalt

DateImported

Gibt das Datum an, an dem der öffentliche Schlüssel dem Transfer Family Family-Benutzer hinzugefügt wurde.

Typ: Zeitstempel

Erforderlich: Ja

SshPublicKeyBody

Gibt den Inhalt des öffentlichen SSH-Schlüssels an, wie in der `PublicKeyId` angegeben.

AWS Transfer Family akzeptiert RSA-, ECDSA- und ED25519-Schlüssel.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 2048 Zeichen.

Erforderlich: Ja

SshPublicKeyId

Gibt an, dass der `SshPublicKeyId` Parameter den Bezeichner des öffentlichen Schlüssels enthält.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 21.

Pattern: `key-[0-9a-f]{17}`

Erforderlich: Ja

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# Tag

Erzeugt ein Schlüssel-Wert-Paar für eine bestimmte Ressource. Tags sind Metadaten, die Sie verwenden können, um nach einer Ressource für verschiedene Zwecke zu suchen und sie zu gruppieren. Sie können Tags auf Server, Benutzer und Rollen anwenden. Ein Tag-Schlüssel kann mehr als einen Wert annehmen. Um beispielsweise Server für Abrechnungszwecke zu gruppieren, können Sie ein Tag mit dem Namen `Group` und erstellen `Research` und `Accounting` die Werte dieser Gruppe zuweisen.

## Inhalt

Key

Der Name, der dem Tag zugewiesen wurde, den Sie erstellen.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 128 Zeichen.

Erforderlich: Ja

Value

Enthält einen oder mehrere Werte, die Sie dem von Ihnen erstellten Schlüsselnamen zugewiesen haben.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 256 Zeichen.

Erforderlich: Ja

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# TagStepDetails

Jeder Schritttyp hat seine eigene `StepDetails` Struktur.

Die Schlüssel/Wert-Paare, die verwendet werden, um eine Datei während der Ausführung eines Workflow-Schritts zu kennzeichnen.

## Inhalt

Name

Der Name des Schritts, der als Kennung verwendet wird.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Die maximale Länge beträgt 30.

Pattern: `[\w-]*`

Erforderlich: Nein

SourceFileLocation

Gibt an, welche Datei als Eingabe für den Workflow-Schritt verwendet werden soll: entweder die Ausgabe des vorherigen Schritts oder die ursprünglich hochgeladene Datei für den Workflow.

- Um die vorherige Datei als Eingabe zu verwenden, geben Sie ein`${previous.file}`. In diesem Fall verwendet dieser Workflow-Schritt die Ausgabedatei aus dem vorherigen Workflow-Schritt als Eingabe. Dies ist der Standardwert.
- Um den Speicherort der ursprünglich hochgeladenen Datei als Eingabe für diesen Schritt zu verwenden, geben Sie ein`${original.file}`.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 0. Maximale Länge beträgt 256 Zeichen.

Pattern: `\$\{(\w+.)+\w+\}`

Erforderlich: Nein

Tags

Array, das 1 bis 10 Schlüssel/Wert-Paare enthält.

Typ: Array von S3Tag-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 1 Element. Die maximale Anzahl beträgt 10 Elemente.

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# UserDetails

Gibt den Benutzernamen, die Server-ID und die Sitzungs-ID für einen Workflow an.

## Inhalt

ServerId

Der vom System zugewiesene eindeutige Bezeichner für eine Transfer-Server-Instanz.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `s-([0-9a-f]{17})`

Erforderlich: Ja

UserName

Eine eindeutige Zeichenfolge, die einen Transfer Family Family-Benutzer identifiziert, der einem Server zugeordnet ist.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 3. Maximale Länge beträgt 100 Zeichen.

Pattern: `[\w][\w@.-]{2,99}`

Erforderlich: Ja

SessionId

Die vom System zugewiesene eindeutige Kennung für eine Sitzung, die dem Workflow entspricht.

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 3. Maximale Länge beträgt 32 Zeichen.

Pattern: `[\w-]*`

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# WorkflowDetail

Gibt die Workflow-ID für den zuzuweisenden Workflow und die für die Ausführung des Workflows verwendete Ausführungsrolle an.

Zusätzlich zu einem Workflow, der ausgeführt wird, wenn eine Datei vollständig hochgeladen wurde, kann `WorkflowDetails` auch eine Workflow-ID (und Ausführungsrolle) für einen Workflow enthalten, der beim teilweisen Upload ausgeführt werden soll. Ein teilweiser Upload erfolgt, wenn die Serversitzung unterbrochen wird, während die Datei noch hochgeladen wird.

## Inhalt

ExecutionRole

Umfasst die erforderlichen Berechtigungen für S3-, EFS- und Lambda-Vorgänge, die Transfer annehmen kann, damit alle Workflow-Schritte mit den erforderlichen Ressourcen ausgeführt werden können

Typ: Zeichenfolge

Längenbeschränkungen: Minimale Länge von 20. Maximale Länge beträgt 2048 Zeichen.

Pattern: `arn:.*role/\S+`

Erforderlich: Ja

WorkflowId

Eine eindeutige Kennung für den Workflow.

Typ: Zeichenfolge

Längenbeschränkungen: Feste Länge von 19.

Pattern: `w-([a-z0-9]{17})`

Erforderlich: Ja

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# WorkflowDetails

Container für den `WorkflowDetail`-Datentyp. Sie wird von Aktionen verwendet, die einen Workflow dazu veranlassen, mit der Ausführung zu beginnen.

## Inhalt

OnPartialUpload

Ein Auslöser, der einen Workflow startet, wenn eine Datei nur teilweise hochgeladen wird. Sie können einen Workflow an einen Server anhängen, der immer dann ausgeführt wird, wenn ein teilweiser Upload erfolgt.

Ein teilweiser Upload erfolgt, wenn eine Datei geöffnet ist, wenn die Sitzung getrennt wird.

> **Note**
>
> `OnPartialUpload`kann maximal ein `WorkflowDetail` Objekt enthalten.

Typ: Array von WorkflowDetail-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 0 Elemente. Die maximale Anzahl beträgt 1 Element.

Erforderlich: Nein

OnUpload

Ein Auslöser, der einen Workflow startet: Der Workflow wird ausgeführt, nachdem eine Datei hochgeladen wurde.

Um einen zugeordneten Workflow von einem Server zu entfernen, können Sie einen leeren `OnUpload`-Datentyp bereitstellen, wie im folgenden Beispiel.

```
aws transfer update-server --server-id s-01234567890abcdef --workflow-details '{"OnUpload":[]}'
```

> **Note**
>
> `OnUpload`kann maximal ein `WorkflowDetail` Objekt enthalten.

Typ: Array von WorkflowDetail-Objekten

Array-Mitglieder: Die Mindestanzahl beträgt 0 Elemente. Die maximale Anzahl beträgt 1 Element.

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# WorkflowStep

Der Grundbaustein für einen Workflow.

## Inhalt

CopyStepDetails

Details für einen Schritt, der eine Dateikopie durchführt.

Besteht aus folgenden Werten:

- Eine Beschreibung
- Ein Amazon-S3-Speicherort für das Ziel der Dateikopie.
- Ein Flag, das angibt, ob eine vorhandene Datei mit demselben Namen überschrieben werden. Der Standardwert ist `FALSE`.

Typ: CopyStepDetails Objekt

Erforderlich: Nein

CustomStepDetails

Details für einen Schritt, der eine AWS Lambda Funktion aufruft.

Besteht aus dem Namen der Lambda-Funktion, dem Ziel und dem Timeout (in Sekunden).

Typ: CustomStepDetails Objekt

Erforderlich: Nein

DecryptStepDetails

Details für einen Schritt, der eine verschlüsselte Datei entschlüsselt.

Besteht aus folgenden Werten:

- Ein beschreibender Name
- Ein Amazon S3- oder Amazon Elastic File System (Amazon EFS) -Speicherort für die zu entschlüsselnde Quelldatei.
- Ein S3- oder Amazon EFS-Speicherort für das Ziel der Dateientschlüsselung.
- Ein Flag, das angibt, ob eine vorhandene Datei mit demselben Namen überschrieben werden. Der Standardwert ist `FALSE`.
- Die Art der verwendeten Verschlüsselung. Derzeit wird nur PGP-Verschlüsselung unterstützt.

Typ: DecryptStepDetails Objekt

Erforderlich: Nein

DeleteStepDetails

Details für einen Schritt, der die Datei löscht.

Typ: DeleteStepDetails Objekt

Erforderlich: Nein

TagStepDetails

Details für einen Schritt, der ein oder mehrere Tags erstellt.

Sie legen einen oder mehrere Markierungen fest. Jedes Tag enthält ein Schlüssel-Wert-Paar.

Typ: TagStepDetails Objekt

Erforderlich: Nein

Type

Derzeit werden die folgenden Schritttypen unterstützt.

- **`COPY`** – Die Datei an einen anderen Ort kopieren.
- **`CUSTOM`**- Führen Sie einen benutzerdefinierten Schritt mit einem AWS Lambda Funktionsziel aus.
- **`DECRYPT`** – Eine Datei entschlüsseln, die vor dem Hochladen verschlüsselt wurde.
- **`DELETE`** – Die Datei löschen.
- **`TAG`** – Der Datei ein Tag hinzufügen.

Typ: Zeichenfolge

Zulässige Werte: `COPY | CUSTOM | TAG | DELETE | DECRYPT`

Erforderlich: Nein

## Weitere Informationen finden Sie unter:

Weitere Informationen zur Verwendung dieser API in einem der sprachspezifischen AWS SDKs finden Sie im Folgenden:

- AWS SDK for C++
- AWS SDK for Java V2
- AWS SDK for Ruby V3

# API-Anfragen stellen

Sie können nicht nur die Konsole verwenden, sondern auch die AWS Transfer Family API verwenden, um Ihre Server programmgesteuert zu konfigurieren und zu verwalten. In diesem Abschnitt werden die AWS Transfer Family-Operationen, das Anfordern des Signierens für die Authentifizierung und die Fehlerbehandlung beschrieben. Informationen zu den Regionen und Endpunkten, die für Transfer Family verfügbar sind, finden Sie unter AWS Transfer FamilyEndpunkte und Kontingente in der Allgemeine AWS-Referenz

> **Note**
>
> Sie können die AWS SDKs auch bei der Entwicklung von Anwendungen mit Transfer Family; verwenden. Die AWS SDKs für Java, .NET und PHP umschließen die zugrunde liegende Transfer Family Family-API und vereinfachen so Ihre Programmieraufgaben. Informationen zum Herunterladen der SDK-Bibliotheken finden Sie unter Beispielcodebibliotheken.

Themen

- Erforderliche Anforderungsheader für Transfer Family
- Eingaben und Unterschreiben von Familienanfragen übertragen
- Fehlermeldungen
- Verfügbare Bibliotheken

## Erforderliche Anforderungsheader für Transfer Family

In diesem Abschnitt werden die erforderlichen Header beschrieben, die Sie mit jeder POST-Anfrage an senden müssen. AWS Transfer Family In HTTP-Headern geben Sie wichtige Informationen über die Abfrage an, z. B, die Operation, die aufgerufen werden soll, das Datum der Abfrage und Informationen zur Ihrer Autorisierung als Sender der Abfrage. In Headern muss Groß- und Kleinschreibung beachtet werden; die Reihenfolge der Header ist nicht wichtig.

Das folgende Beispiel zeigt Header, die in der ListServersOperation verwendet werden.

```
POST / HTTP/1.1
Host: transfer.us-east-1.amazonaws.com
x-amz-target: TransferService.ListServers
x-amz-date: 20220507T012034Z
Authorization: AWS4-HMAC-SHA256 Credential=AKIDEXAMPLE/20220507/us-east-1/transfer/
aws4_request,
    SignedHeaders=content-type;host;x-amz-date;x-amz-target,
    Signature=13550350a8681c84c861aac2e5b440161c2b33a3e4f302ac680ca5b686de48de
Content-Type: application/x-amz-json-1.1
Content-Length: 17

{"MaxResults":10}
```

Im Folgenden sind die Header aufgeführt, die in Ihren POST-Anfragen an Transfer Family enthalten sein müssen. Die unten aufgeführten Header, die mit „x-amz“ beginnen, sind spezifisch für. AWS Alle anderen aufgeführten Header sind allgemeine Header für HTTP-Transaktionen.

| Header | Beschreibung |
| --- | --- |
| `Authorization` | Der Autorisierungsheader ist erforderlich. Das Format ist die standardm äßige Sigv4-Anforderungssignatur, die unter AWSAPI-Anfragen signieren dokumentiert ist. |
| `Content-Type` | Verwenden Sie ihn `application/x-amz-json-1.1` als Inhaltstyp für alle Anfragen an Transfer Family.<br><br>`Content-Type: application/x-amz-json-1.1` |
| `Host` | Verwenden Sie den Host-Header, um den Transfer Family Family-En dpunkt anzugeben, an den Sie Ihre Anfrage senden. Dies `transfer.us-east-1.amazonaws.com` ist beispielsweise der Endpunkt für die Region USA Ost (Ohio). Weitere Informationen zu den für Transfer Family verfügbaren Endpunkten finden Sie unter AWS Transfer FamilyEndpunkte und Kontingente in der. Allgemeine AWS-Referenz<br><br>`Host: transfer. region.amazonaws.com` |

| Header | Beschreibung |
| --- | --- |
| `x-amz-date` | Sie müssen den Zeitstempel entweder im `Date` HTTP-Header oder im AWS `x-amz-date` Header angeben. (Einige HTTP-Client-Bibliotheken lassen den Header `Date` nicht zu.) Wenn ein `x-amz-date` Header vorhanden ist, ignoriert die Transfer Family jeden `Date` Header während der Anforderungsauthentifizierung. Das `x-amz-date` Format muss ISO8601 sein, im Format YYYYMMDD'T 'HHMMSS'Z'.<br><br>`x-amz-date: YYYYMMDD'T'HHMMSS'Z'` |
| `x-amz-target` | In diesem Header werden die Version der API und die angefragte Operation angegeben. Die Werte des Ziel-Headers werden durch Verknüpfung der API-Version mit dem API-Namen gebildet und haben folgendes Format.<br><br>`x-amz-target: TransferService. operationName`<br><br>Der OperationName-Wert (zum Beispiel`ListServers` ) kann in der API-Liste gefunden werden. ListServers |
| `x-amz-security-token` | Dieser Header ist erforderlich, wenn es sich bei den zum Signieren der Anfrage verwendeten Anmeldeinformationen um temporäre Anmeldein formationen oder um Sitzungsanmeldedaten handelt (Einzelheiten finden Sie unter Verwenden temporärer Anmeldeinformationen mit AWS Ressourcen im IAM-Benutzerhandbuch). Weitere Informationen finden Sie unter Hinzufügen der Signatur Allgemeine Amazon Web Services-Referenz zur HTTP-Anfrage in. |

## Eingaben und Unterschreiben von Familienanfragen übertragen

Alle Anforderungseingaben müssen als Teil der JSON-Nutzlast im Anfragetext gesendet werden. Für Aktionen, bei denen alle Anforderungsfelder optional sind, müssen Sie beispielsweise `ListServers` immer noch ein leeres JSON-Objekt im Anforderungstext angeben, z. B. `{}` Die Struktur der

Payload-Anfrage/Antwort von Transfer Family ist beispielsweise in der vorhandenen API-Referenz dokumentiert. DescribeServer

Transfer Family unterstützt die Authentifizierung mit AWS Signature Version 4. Einzelheiten finden Sie unter AWSAPI-Anfragen signieren.

## Fehlermeldungen

Bei einem Fehler enthalten die Informationen im Antwort-Header:

- Inhaltstyp: `application/x-amz-json-1.1`
- Einen passenden `4xx`- oder `5xx`-HTTP-Statuscode

Der Textkörper einer Fehlermeldung enthält Informationen zu dem aufgetretenen Fehler. Das folgende Beispiel zeigt eine Fehlerantwort mit der Ausgabesyntax von Antwortelementen für alle Fehlermeldungen.

```
{
    "__type": "String",
    "Message": "String", <!-- Message is lowercase in some instances -->
    "Resource": String,
    "ResourceType": String
    "RetryAfterSeconds": String
}
```

In der folgenden Tabellen werden die Felder der JSON-Fehlerantwort in dieser Syntax erläutert.

__type

Eine der Ausnahmen bei einem Transfer Family Family-API-Aufruf.

Typ: Zeichenfolge

Nachricht oder Nachricht

Eine der Operationsfehlercode-Nachrichten .

> **Note**
>
> Einige Ausnahmen verwendenmessage, andere verwendenMessage. Sie können den Code für Ihre Schnittstelle überprüfen, um den richtigen Fall zu ermitteln. Alternativ können Sie jede Option testen, um zu sehen, welche funktioniert.

Typ: Zeichenfolge

Resource

Die Ressource, für die der Fehler ausgelöst wurde. Wenn Sie beispielsweise versuchen, einen Benutzer zu erstellen, der bereits existiert, `Resource` ist dies der Benutzername für den vorhandenen Benutzer.

Typ: Zeichenfolge

ResourceType

Der Ressourcentyp, für den der Fehler ausgelöst wird. Wenn Sie beispielsweise versuchen, einen Benutzer zu erstellen, der bereits existiert, `ResourceType` ist `User` der.

Typ: Zeichenfolge

RetryAfterSeconds

Die Anzahl der Sekunden, die gewartet werden muss, bevor der Befehl erneut ausgeführt wird.

Typ: Zeichenfolge

## Beispiele für Antworten auf Fehler

Der folgende JSON-Hauptteil wird zurückgegeben, wenn Sie die `DescribeServer` API aufrufen und einen Server angeben, der nicht existiert.

```
{
  "__type": "ResourceNotFoundException",
  "Message": "Unknown server",
  "Resource": "s-11112222333344444",
  "ResourceType": "Server"
}
```

Der folgende JSON-Hauptteil wird zurückgegeben, wenn die Ausführung einer API zu einer Drosselung führt.

```
{
   "__type":"ThrottlingException",
   "RetryAfterSeconds":"1"
}
```

Der folgende JSON-Hauptteil wird zurückgegeben, wenn Sie die `CreateServer` API verwenden und nicht über ausreichende Berechtigungen verfügen, um einen Transfer Family Family-Server zu erstellen.

```
{
  "__type": "AccessDeniedException",
  "Message": "You do not have sufficient access to perform this action."
}
```

Der folgende JSON-Hauptteil wird zurückgegeben, wenn Sie die `CreateUser` API verwenden und einen Benutzer angeben, der bereits existiert.

```
{
  "__type": "ResourceExistsException",
  "Message": "User already exists",
  "Resource": "Alejandro-Rosalez",
  "ResourceType": "User"
}
```

## Verfügbare Bibliotheken

AWSbietet Bibliotheken, Beispielcode, Tutorials und andere Ressourcen für Softwareentwickler, die es vorziehen, Anwendungen mithilfe sprachspezifischer APIs anstelle der Befehlszeilentools und der Abfrage-API zu erstellen. Diese Bibliotheken bieten grundlegende Funktionen (nicht in den APIs enthalten) wie Anforderungsauthentifizierung, Wiederholungen von Anfragen und Fehlerbehandlung, sodass der Einstieg erleichtert wird. Siehe Tools, auf denen Sie aufbauen können AWS

Bibliotheken und Beispielcode in allen Sprachen finden Sie unter Beispielcode und Bibliotheken.

# Geläufige Parameter

Die folgende Liste enthält die Parameter, die alle Aktionen zum Signieren von Signature-Version-4-Anforderungen mit einer Abfragezeichenfolge verwenden. Alle aktionsspezifischen Parameter werden im Thema für diese Aktion aufgelistet. Weitere Informationen zur Verwendung von Signature Version 4 finden Sie unter Signieren von AWS-API-Anforderungen im IAM-Benutzerhandbuch.

Action

Die auszuführende Aktion.

Typ: Zeichenfolge

Erforderlich: Ja

Version

Die API-Version, für die die Anforderung geschrieben wurde, ausgedrückt im Format JJJJ-MM-TT.

Typ: Zeichenfolge

Erforderlich: Ja

X-Amz-Algorithm

Der Hashalgorithmus, den Sie zum Erstellen der Anforderungssignatur verwendet haben.

Bedingung: Geben Sie diesen Parameter an, wenn Sie Authentifizierungsinformationen in eine Abfragezeichenfolge anstatt in den HTTP-Autorisierungsheader aufnehmen.

Typ: Zeichenfolge

Zulässige Werte: `AWS4-HMAC-SHA256`

Required: Conditional

X-Amz-Credential

Der Wert des Anmeldeinformationsumfangs. Dabei handelt es sich um eine Zeichenfolge, die Ihren Zugriffsschlüssel, das Datum, die gewünschte Region und eine Zeichenfolge zur Beendigung („aws4_request“) beinhaltet. Der Wert wird im folgenden Format ausgedrückt: Zugriffsschlüssel/JJJJMMTT/Region/Service/aws4_request.

Weitere Informationen finden Sie unter Erstellen einer signierten AWS-API-Anfrage im IAM-Benutzerhandbuch.

Bedingung: Geben Sie diesen Parameter an, wenn Sie Authentifizierungsinformationen in eine Abfragezeichenfolge anstatt in den HTTP-Autorisierungsheader aufnehmen.

Typ: Zeichenfolge

Required: Conditional

X-Amz-Date

Das Datum, das zum Erstellen der Signatur verwendet wird. Das Format muss das ISO 8601-Basisformat (JJJJMMTT'T'SSMMSS'Z') sein. Die folgende Datumszeit ist beispielsweise ein gültiger X-Amz-Date-Wert: `20120325T120000Z`.

Bedingung: X-Amz-Date ist bei allen Anforderungen optional. Damit kann das Datum überschrieben werden, das zum Signieren von Anforderungen verwendet wird. Wenn der Date-Header im ISO 8601-Basisformat angegeben ist, ist X-Amz-Date nicht erforderlich. Wenn X-Amz-Date verwendet wird, überschreibt es immer den Wert des Date-Headers. Weitere Informationen finden Sie unter Elemente einer AWS-API-Anfragesignatur im IAM-Benutzerhandbuch.

Typ: Zeichenfolge

Required: Conditional

X-Amz-Security-Token

Das temporäre Sicherheitstoken, das durch einen Anruf von AWS Security Token Service (AWS STS) abgerufen wurde. Eine Liste der Services, die temporäre Sicherheits-Anmeldeinformationen von AWS STS unterstützen, finden Sie im IAM-Benutzerhandbuch unter AWS-Services, die mit IAM funktionieren.

Bedingung: Wenn Sie temporäre Sicherheits-Anmeldeinformationen von AWS STS nutzen, müssen Sie das Sicherheitstoken einschließen.

Typ: Zeichenfolge

Required: Conditional

X-Amz-Signature

Gibt die hex-codierte Signatur an, die aus der zu signierenden Zeichenfolge und dem abgeleiteten Signaturschlüssel berechnet wurde.

Bedingung: Geben Sie diesen Parameter an, wenn Sie Authentifizierungsinformationen in eine Abfragezeichenfolge anstatt in den HTTP-Autorisierungsheader aufnehmen.

Typ: Zeichenfolge

Required: Conditional

X-Amz-SignedHeaders

Gibt alle HTTP-Header an, die als Teil der kanonischen Anforderung enthalten waren. Weitere Informationen zur Angabe signierter Header finden Sie unter Erstellen einer signierten AWS-API-Anfrage im IAM-Benutzerhandbuch.

Bedingung: Geben Sie diesen Parameter an, wenn Sie Authentifizierungsinformationen in eine Abfragezeichenfolge anstatt in den HTTP-Autorisierungsheader aufnehmen.

Typ: Zeichenfolge

Required: Conditional

# Häufige Fehler

In diesem Abschnitt sind Fehler aufgeführt, die häufig bei den API-Aktionen aller AWS-Services auftreten. Informationen zu Fehlern, die spezifisch für eine API-Aktion für diesen Service sind, finden Sie unter dem Thema für diese API-Aktion.

AccessDeniedException

Sie haben keinen ausreichenden Zugriff zum Durchführen dieser Aktion.

HTTP Status Code: 400

IncompleteSignature

Die Anforderungssignatur entspricht nicht den AWS-Standards.

HTTP Status Code: 400

InternalFailure

Die Anforderungsverarbeitung ist fehlgeschlagen, da ein unbekannter Fehler, eine Ausnahme oder ein Fehler aufgetreten ist.

HTTP Status Code: 500

InvalidAction

Die angeforderte Aktion oder Operation ist ungültig. Überprüfen Sie, ob die Aktion ordnungsgemäß eingegeben wurde.

HTTP Status Code: 400

InvalidClientTokenId

Das angegebene X.509-Zertifikat oder die AWS-Zugriffsschlüssel-ID ist nicht in unseren Datensätzen vorhanden.

HTTP Status Code: 403

NotAuthorized

Sie haben keine Berechtigung zum Ausführen dieser Aktion.

HTTP Status Code: 400

OptInRequired

Die AWS-Zugriffsschlüssel-ID benötigt ein Abonnement für den Service.

HTTP Status Code: 403

RequestExpired

Die Anforderung hat den Service mehr als 15 Minuten nach dem Datumsstempel oder mehr als 15 Minuten nach dem Ablaufdatum der Anforderung erreicht (z. B. für vorsignierte URLs) oder der Datumsstempel auf der Anforderung liegt mehr als 15 Minuten in der Zukunft.

HTTP Status Code: 400

ServiceUnavailable

Die Anforderung ist aufgrund eines temporären Fehlers des Servers fehlgeschlagen.

HTTP Status Code: 503

ThrottlingException

Die Anforderung wurde aufgrund der Drosselung von Anforderungen abgelehnt.

HTTP Status Code: 400

ValidationError

Die Eingabe erfüllt nicht die von einem AWS-Service definierten Einschränkungen.

HTTP Status Code: 400

# Dokumenthistorie für AWS Transfer Family

In der folgenden Tabelle wird die Dokumentation für diese Version von beschrieben AWS Transfer Family.

- API-Version: transfer-2018-11-05
- Letzte Aktualisierung der Dokumentation: 23. April 2024

| Änderung | Beschreibung | Datum |
| --- | --- | --- |
| Möglichkeit für SFTP-Konnektoren, entfernte Dateien und Verzeichnisse aufzulisten | Transfer Family bietet unseren Kunden die Möglichkeit, SFTP-Konnektoren zu verwenden, um Dateien aufzulisten, die auf Remote-SFTP-Servern gespeichert sind. Details hierzu finden Sie unter Inhalt eines Remote-Verzeichnisses auflisten | 23. April 2024 |
| Möglichkeit, das selbstsignierte TLS-Zertifikat eines Handelspartners für den AS2-Nachrichtenaustausch zu verwenden | AWS Transfer Family hat die Option hinzugefügt, das öffentliche, selbstsignierte TLS-Zertifikat eines Handelspartners zu importieren und zu verwenden, um Applicability Statement 2-Nachrichten (AS2) über HTTPS an seinen Server zu senden. | 12. April 2024 |
| Hinzufügung von Sicherheitsrichtlinien für SFTP-Konnektoren | AWS Transfer Family hat Sicherheitsrichtlinien für die Verwendung mit SFTP-Konnektoren hinzugefügt. Details hierzu finden Sie unter AWS | 5. April 2024 |

| Änderung | Beschreibung | Datum |
| --- | --- | --- |
| | Transfer Family Sicherheitsrichtlinien für SFTP-Konnektoren. | |
| Integrieren Sie mit Amazon EventBridge | AWS Transfer Family veröffentlicht jetzt automatisch Ereignisse EventBridge für alle Dateiübertragungsvorgänge auf Amazon. Details hierzu finden Sie unter Transfer Family Ereignisse verwalten mit Amazon EventBridge. | 8. Februar 2024 |
| Hinzufügung neuer Sicherheitsrichtlinien | AWS Transfer Family hat neue FIPS- und Nicht-FIPS-Sicherheitsrichtlinien hinzugefügt. Außerdem ist die Standardsicherheitsrichtlinie, die Servern zugewiesen ist, immer die neueste Sicherheitsrichtlinie. Details hierzu finden Sie unter Sicherheitsrichtlinien für AWS Transfer Family Server. | 5. Februar 2024 |
| Support für statische IP-Adressen für SFTP-Anschlüsse und AS2 | Transfer Family bietet jetzt statische IP-Adressen für SFTP-Anschlüsse und AS2. Dies ermöglicht die Verbindung mit Remote-SFTP-Servern, die durch IP-Allowlisting-Kontrollen gesichert sind. Für AS2 führen wir statische IP-Adressen für asynchrone MDN-Antworten von AS2-Servern ein. | 16. Januar 2024 |

| Änderung | Beschreibung | Datum |
| --- | --- | --- |
| Das Benutzerhandbuch wurde neu organisiert, um es besser an die neueste Version von AWS Transfer Family anzupassen. | Transfer Family hat seit der Entstehung des Leitfadens mehrere Funktionen hinzugefügt, was eine Umstrukturierung des Leitfadens erforderlich machte. | 3. Januar 2024 |
| Verbesserungen bei den logischen Verzeichniszuordnungen<br><br>Leistungsoptimierung der Amazon S3 S3-Liste | Transfer Family unterstützt jetzt logische Verzeichniszuordnungen bis zu 2,1 MB. Sie können jetzt auch angeben, ob eine Benutzerzuordnung zu einer Datei erfolgt. Weitere Informationen finden Sie unter Regeln für die Verwendung logischer Verzeichnisse.<br><br>Wenn Sie einen Server erstellen oder aktualisieren, der Amazon S3 als Speicher verwendet, können Sie jetzt die Leistung beim Auflisten Ihrer S3-Verzeichnisse (oder Ordner) optimieren. Weitere Informationen finden Sie unter Konfiguration eines SFTP-, FTPS- oder FTP-Serverendpunkts. | 17. November 2023 |

| Änderung | Beschreibung | Datum |
| --- | --- | --- |
| Alternativer Port für SFTP-Server mit Virtual Private Cloud (VPC) -Endpunkten | Sie können jetzt einen alternativen, nicht standardmäßigen Port für Ihre SFTP Transfer Family Family-Server mit VPC-Endpunkten aktivieren. Weitere Informationen finden Sie unter Erstellen Sie einen Server in einer virtuellen privaten Cloud. | 17. November 2023 |
| Support für SFTP-Anschlüsse | SFTP-Konnektoren erweitern die Möglichkeiten AWS Transfer Family zur Kommunikation mit Remoteservern sowohl in der Cloud als auch vor Ort. Weitere Informationen finden Sie unter Senden und Abrufen von Dateien mithilfe eines SFTP-Connectors. | 25. Juli 2023 |
| Support für AS2 Basic-Authentifizierung | Transfer Family unterstützt jetzt die Verwendung der Standardauthentifizierung für Server, die das Applicability Statement 2 (AS2)-Protokoll verwenden. Weitere Informationen finden Sie unter Standardauthentifizierung für AS2-Konnektoren. | 30. Juni 2023 |

| Änderung | Beschreibung | Datum |
| --- | --- | --- |
| Support für strukturiertes JSON-Logging | Transfer Family unterstützt jetzt die Bereitstellung strukturierter JSON-Protokolle an Amazon CloudWatch, die Gruppierung von Protokollströmen in benutzerdefinierte Protokollgruppen und die Ausführung gängiger Protokollabfragen über Protokolle hinweg. Weitere Informationen finden Sie unter Amazon CloudWatch loggt sich ein für AWS Transfer Family. | 24. Juni 2023 |
| Support für mehrere Authentifizierungsmethoden | Transfer Family unterstützt die Authentifizierung mit einem Passwort, einem öffentlichen/privaten key pair oder beidem. Dies ist für Server verfügbar, die das SFTP-Protokoll und einen benutzerdefinierten Identitätsanbieter verwenden. Weitere Informationen finden Sie unter Erstellen Sie einen SFTP-fähigen Server. | 17. Mai 2023 |

| Änderung | Beschreibung | Datum |
| --- | --- | --- |
| Support für die Pretty Good Privacy (PGP) -Entschlüsselung mit Dateien, die Transfer Family mit Workflows verarbeitet | Transfer Family bietet integrierte Unterstützung für die Pretty Good Privacy (PGP) -Entschlüsselung. Sie können die PGP-Entschlüsselung für Dateien verwenden, die über SFTP, FTPS oder FTP auf Amazon Simple Storage Service (Amazon S3) oder Amazon Elastic File System (Amazon EFS) hochgeladen werden. Weitere Informationen finden Sie unter Generieren und verwalten Sie PGP-Schlüssel und Verwenden Sie die PGP-Entschlüsselung in Ihrem Workflow. | 21. Dezember 2022 |
| Vollständig verwaltete Unterstützung für das AS2-Dateiübertragungsprotokoll Applicability Statement 2 (Applicability Statement 2) mit Transfer Family Family-Servern | Sie können Server einrichten, die das AS2-Protokoll zum Senden und Empfangen von Informationen an und von Handelspartnern innerhalb oder außerhalb der Umgebung verwenden. AWS Weitere Informationen finden Sie unter AS2 konfigurieren. | 25. Juli 2022 |

| Änderung | Beschreibung | Datum |
| --- | --- | --- |
| Support für Display-Banner beim Erstellen eines Servers | Sie können benutzerdefinierte Nachrichten hinzufügen, wenn Sie Server erstellen. Sie können eine Nachricht vor der Authentifizierung (alle Protokolle) und eine Nachricht nach der Authentifizierung (für FTP- und FTPS-Server) anzeigen. Weitere Informationen finden Sie unter Erstellen Sie einen SFTP-fähigen Server, Erstellen Sie einen FTPS-fähigen Server oder Erstellen Sie einen FTP-fähigen Server. | 17. Februar 2022 |
| Support für AWS Lambda als Identitätsanbieter | Sie können jetzt eine Verbindung zu einem benutzerdefinierten Identitätsanbieter herstellen, indem Sie AWS Lambda dessen Transfer Family Family-Server verwenden. Bisher mussten Sie eine Amazon API Gateway URL angeben, um einen benutzerdefinierten Identitätsanbieter zu integrieren. Weitere Informationen finden Sie unter Wird AWS Lambda zur Integration Ihres Identitätsanbieters verwendet. | 16. November 2021 |

| Änderung | Beschreibung | Datum |
| --- | --- | --- |
| Support für Workflows zur verwalteten Dateiübertragung | Workflows für verwaltete Dateiübertragungen bieten Ihnen nach dem Upload Abstraktionen für die allgemeinen Aufgaben, die Sie derzeit manuell ausführen. Weitere Informationen finden Sie unter AWS Transfer Family verwaltete Workflows. | 2. September 2021 |
| Support für AWS Directory Service for Microsoft Active Directory | Zusätzlich zu dienstverwalteten und benutzerdefinierten Identitätsanbietern können Sie jetzt auch AWS Directory Service for Microsoft Active Directory den Benutzerzugriff für die Authentifizierung und Autorisierung verwalten. Weitere Informationen finden Sie unter Verwenden des AWS Directory Service-Identitätsanbieters. | 24. Mai 2021 |
| Neu AWS-Regionen | AWS Transfer Family ist jetzt in der Region Afrika (Kapstadt) verfügbar. Weitere Informationen zu Transfer Family Family-Endpunkten finden Sie unter AWS Transfer Family Endpunkte und Kontingente in der. Allgemeine AWS-Referenz | 24. Februar 2021 |

| Änderung | Beschreibung | Datum |
| --- | --- | --- |
| Neu AWS-Regionen | AWS Transfer Family ist jetzt in den Regionen Asien-Pazifik (Hongkong) und Naher Osten (Bahrain) verfügbar. Weitere Informationen zu Transfer Family Family-Endpunkten finden Sie unter AWS Transfer Family Endpunkte und Kontingente in der. Allgemeine AWS-Referenz | 17. Februar 2021 |
| Support für Amazon EFS als Datenspeicher | Transfer Family unterstützt jetzt Dateiübertragungen in und aus dem Amazon Elastic File System (Amazon EFS). Amazon EFS ist ein einfaches , skalierbares, vollständig verwaltetes elastisches NFS-Dateisystem. Weitere Informationen finden Sie unter Ein Amazon EFS-Dateisystem konfigurieren. | 06. Januar 2021 |
| Support für AWS WAF | Transfer Family unterstützt AWS WAF jetzt eine Firewall für Webanwendungen, die Webanwendungen und API-Operationen vor Angriffen schützt. Weitere Informationen finden Sie unter Fügen Sie eine Firewall für Webanwendungen hinzu. | 24. November 2020 |

| Änderung | Beschreibung | Datum |
| --- | --- | --- |
| Support für mehrere Sicherheitsgruppen in einer Virtual Private Cloud (VPC) | Sie können jetzt mehrere Sicherheitsgruppen an einen Server in einer VPC anhängen. Weitere Informationen finden Sie unter Erstellen Sie einen Server in einer virtuellen privaten Cloud. | 15. Oktober 2020 |
| Neu AWS-Regionen | Transfer Family ist jetzt in den AWS GovCloud (US) Regionen verfügbar. Weitere Informationen zu Transfer Family Family-Endpunkten für AWS GovCloud (US) Regionen finden Sie unter AWS Transfer Family Endpunkte und Kontingente in der. Allgemeine AWS-Referenz Informationen zur Verwendung von Transfer Family in den AWS GovCloud (US) Regionen finden Sie AWS Transfer Familyim AWS GovCloud (US) Benutzerhandbuch. | 30. September 2020 |
| Eine Sicherheitsrichtlinie mit unterstützten kryptografischen Algorithmen kann jetzt an Ihren Server angehängt werden | Sie können Ihrem Server jetzt eine Sicherheitsrichtlinie hinzufügen, die eine Reihe unterstützter kryptografischer Algorithmen enthält. Weitere Informationen finden Sie unter Sicherheitsrichtlinien für AWS Transfer Family Server. | 12. August 2020 |

| Änderung | Beschreibung | Datum |
| --- | --- | --- |
| Unterstützung von Endpunkten für den Federal Information Processing Standard (FIPS) | FIPS-fähige Endpunkte sind jetzt in Nordamerika verfügbar. AWS-Regionen Informationen zu den verfügbaren Regionen finden Sie unter AWS Transfer Family Endpunkte und Kontingente in der. Allgemeine AWS-Referenz Informationen zur Aktivierung von FIPS für einen SFTP-fähigen Serverendpunkt finden Sie unter. Erstellen Sie einen SFTP-fähigen Server Informationen zur Aktivierung von FIPS für einen FTPS-fähigen Serverendpunkt finden Sie unter. Erstellen Sie einen FTPS-fähigen Server Informationen zur Aktivierung von FIPS für einen FTP-fähigen Serverendpunkt finden Sie unter. Erstellen Sie einen FTP-fähigen Server | 12. August 2020 |
| Erhöhung der Zeichenlänge des Benutzernamens und zusätzliche zulässige Zeichen | Benutzernamen können jetzt At-Zeichen (@) und Punkte (.) enthalten und dürfen eine maximale Länge von 100 Zeichen haben. Informationen zum Hinzufügen eines Benutzers finden Sie unterVerwalten von Benutzern für Serverendpunkte. | 12. August 2020 |

| Änderung | Beschreibung | Datum |
| --- | --- | --- |
| Support für die automatische Erstellung von Amazon CloudWatch Logging AWS Identity and Access Management (IAM) -Rollen | Transfer Family unterstützt jetzt die automatische Erstellung einer CloudWatch Protokollierungs-IAM-Rolle zur Anzeige der Endbenutzeraktivitäten. Weitere Informationen finden Sie unter Erstellen Sie einen SFTP-fähigen Server, Erstellen Sie einen FTPS-fähigen Server oder Erstellen Sie einen FTP-fähigen Server. | 30. Juli 2020 |
| AWS Transfer Family unterstützt jetzt Quell-IP als Autorisierungsfaktor. | Transfer Family bietet Unterstützung für die Verwendung der Quell-IP-Adressen von Endbenutzern als Autorisierungsfaktor, sodass Sie bei der Autorisierung des Zugriffs über Secure File Transfer Protocol (SFTP), File Transfer Protocol over SSL (FTPS) oder File Transfer Protocol (FTP) eine zusätzliche Sicherheitsebene einrichten können. Weitere Informationen finden Sie unter Mit Anbietern benutzerdefinierter Identitäten arbeiten. | 9. Juni 2020 |

| Änderung | Beschreibung | Datum |
| --- | --- | --- |
| AWS Transfer for SFTP ist jetzt verfügbar AWS Transfer Family und bietet Unterstützung für FTP und FTPS. | Sie können jetzt zwei zusätzliche Protokolle für die Dateiübertragungen Ihrer Benutzer verwenden: File Transfer Protocol Secure (FTPS) und File Transfer Protocol (FTP). Benutzer können zusätzlich zur bestehenden Unterstützung des Secure File Transfer Protocol (SFTP) auch FTP über SSL (FTPS) und FTP-basierte Klartext-Workflows verschieben AWS, ausführen, sichern und integrieren. | 23. April 2020 |
| Support für Virtual Private Cloud (VPC) -Sicherheitsgruppen und Elastic IP-Adressen | Sie können jetzt mithilfe von Sicherheitsgruppen eine Zulassungsliste für eingehende IP-Adressen erstellen, wodurch eine zusätzliche Sicherheitsebene für Server bereitgestellt wird. Sie können Elastic IP-Adressen auch dem Endpunkt Ihres Servers zuordnen. Auf diese Weise können Sie Benutzern hinter Firewalls den Zugriff auf diesen Endpunkt ermöglichen. Weitere Informationen finden Sie unter Erstellen Sie einen Server in einer virtuellen privaten Cloud. | 10. Januar 2020 |

| Änderung | Beschreibung | Datum |
| --- | --- | --- |
| Support für die Arbeit in einer VPC | Sie können jetzt einen Server in einer VPC erstellen. Sie können Ihren Server verwenden, um Daten über Ihren Client zu und von einem Amazon S3 S3-Bucket zu übertragen, ohne das öffentliche Internet nutzen zu müssen. Weitere Informationen finden Sie unter Erstellen Sie einen Server in einer virtuellen privaten Cloud. | 27. März 2019 |
| Erste Version von AWS Transfer Family veröffentlicht. | Diese Erstversion umfasst das Einrichten von Verzeichnissen, beschreibt die ersten Schritte und stellt Informationen zur Client-Konfiguration, zur Benutzerkonfiguration und zur Überwachung bereit. | 25. November 2018 |

# AWS-Glossar

Die neueste AWS-Terminologie finden Sie im AWS-Glossar in der AWS-Glossar-Referenz.

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.