



Panduan Pengguna

AWS Transfer Family



AWS Transfer Family: Panduan Pengguna

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Merek dagang dan tampilan dagang Amazon tidak boleh digunakan sehubungan dengan produk atau layanan apa pun yang bukan milik Amazon, dengan cara apa pun yang dapat menyebabkan kebingungan di antara pelanggan, atau dengan cara apa pun yang merendahkan atau mendiskreditkan Amazon. Semua merek dagang lain yang tidak dimiliki oleh Amazon merupakan hak milik masing-masing pemiliknya, yang mungkin atau tidak terafiliasi, terkait dengan, atau disponsori oleh Amazon.

Table of Contents

Apa itu AWS Transfer Family?	1
Bagaimana cara AWS Transfer Family kerja	3
Posting blog yang relevan untuk Transfer Family	4
Prasyarat	7
Wilayah, titik akhir, dan kuota	7
Mendaftar untuk AWS	7
Mengonfigurasi penyimpanan	8
Konfigurasi bucket Amazon S3	9
Konfigurasi sistem EFS file Amazon	13
Buat IAM peran dan kebijakan	16
Membuat peran pengguna	18
Bagaimana kebijakan sesi bekerja	21
Contoh kebijakan akses baca/tulis	24
Tutorial Transfer Family	28
Memulai dengan titik akhir server	28
Prasyarat	29
Masuk ke konsol	30
Buat server SFTP yang diaktifkan	30
Menambahkan pengguna terkelola layanan	31
Transfer file menggunakan klien	33
Buat Alur Kerja dekripsi	35
Langkah 1: Konfigurasi peran eksekusi	35
Langkah 2: Buat alur kerja terkelola	37
Langkah 3: Tambahkan alur kerja ke server dan buat pengguna	38
Langkah 4: Buat PGP key pair	39
Langkah 5: Simpan kunci PGP pribadi di AWS Secrets Manager	40
Langkah 6: Enkripsi file	42
Langkah 7: Jalankan alur kerja dan lihat hasilnya	42
Buat dan gunakan SFTP konektor	43
Langkah 1: Buat sumber daya pendukung yang diperlukan	44
Langkah 2: Buat dan uji SFTP konektor	49
Langkah 3: Kirim dan ambil file menggunakan konektor SFTP	53
Prosedur untuk membuat server Transfer Family untuk digunakan sebagai SFTP server jarak jauh Anda	56

Menggunakan penyedia identitas kustom	59
Prasyarat	59
Langkah 1: Buat CloudFormation tumpukan	60
Langkah 2: Periksa konfigurasi metode API Gateway untuk server Anda	61
Langkah 3: Lihat detail server Transfer Family	61
Langkah 4: Uji apakah pengguna Anda dapat terhubung ke server	63
Langkah 5: Uji SFTP koneksi dan transfer file	63
Langkah 6: Batasi akses ke ember	64
Perbarui Lambda jika menggunakan Amazon EFS	66
Siapkan AS2 konfigurasi	67
Langkah 1: Buat sertifikat untuk AS2	69
Langkah 2: Buat server Transfer Family yang menggunakan AS2 protokol	72
Langkah 3: Impor sertifikat sebagai sumber sertifikat Transfer Family	76
Langkah 4: Buat profil untuk Anda dan mitra dagang Anda	77
Langkah 5: Buat kesepakatan antara Anda dan pasangan	78
Langkah 6: Buat konektor antara Anda dan pasangan	79
Langkah 7: Uji pertukaran file AS2 dengan menggunakan Transfer Family	80
Transfer Family untukSFTP,FTPS, FTP	83
Opsi penyedia identitas	83
AWS Transfer Family matriks tipe titik akhir	85
Mengkonfigurasi titik akhir server Transfer Family	89
Buat server SFTP yang diaktifkan	91
Buat server FTPS yang diaktifkan	103
Buat server FTP yang diaktifkan	115
Buat server di VPC	126
Bekerja dengan nama host khusus	147
Mentransfer file melalui titik akhir server	151
SFTP/FTPS/FTPPerintah yang Tersedia	154
Temukan titik VPC akhir Amazon Anda	155
Hindari setstat kesalahan	157
Gunakan Buka SSH	34
Gunakan Win SCP	159
Gunakan Cyberduck	33
Gunakan FileZilla	162
Gunakan klien Perl	164
Pemrosesan unggahan pasca	164

Mengelola pengguna	165
Pengguna yang dikelola layanan	167
Pengguna layanan direktori	177
Pengguna penyedia identitas khusus	194
Gunakan direktori logis	224
Aturan untuk menggunakan direktori logis	225
Menerapkan direktori logis dan chroot	227
Konfigurasi contoh direktori logis	229
Konfigurasi direktori logis untuk Amazon EFS	230
AWS Lambda Tanggapan khusus	231
SFTPkonektor	232
Konfigurasi SFTP konektor	232
Buat SFTP konektor	233
SFTP algoritma konektor	241
Simpan rahasia untuk digunakan dengan SFTP konektor	242
Menghasilkan dan memformat kunci pribadi SFTP konektor	243
Uji SFTP konektor	246
Transfer file dengan SFTP konektor	248
Kelola SFTP konektor	250
Perbarui SFTP konektor	250
Lihat detail SFTP konektor	250
Kuota untuk konektor SFTP	252
Transfer Family untuk AS2	254
Kasus penggunaan AS2	255
Konfigurasi AS2	259
Membuat AS2 server menggunakan konsol Transfer Family	260
Buat AS2 server menggunakan template	263
AS2 konfigurasi	266
AS2 fitur dan kemampuan	272
Konfigurasi AS2 konektor	273
Buat AS2 konektor	274
AS2 algoritma konektor	277
Otentikasi dasar untuk konektor AS2	278
Aktifkan otentikasi dasar untuk konektor AS2	280
Lihat detail konektor	284
Kelola AS2 mitra	285

Impor AS2 sertifikat	285
AS2rotasi sertifikat	287
Buat AS2 profil	289
Buat AS2 perjanjian	289
Transfer AS2 pesan	290
Kirim AS2 pesan	291
Menerima AS2 pesan	292
Konfigurasikan HTTPS untuk AS2	293
Transfer file dengan AS2 konektor	297
Nama dan lokasi file	298
Kode status	300
Contoh JSON file	301
Monitor AS2	303
Kode Status AS2	304
Kode kesalahan AS2	305
Mengelola alur kerja pemrosesan file	319
Buat alur kerja	321
Konfigurasikan dan jalankan alur kerja	322
Lihat detail alur kerja	325
Gunakan langkah-langkah yang telah ditentukan	327
Salin berkas	328
Dekripsi file	333
Berkas tag	339
Hapus berkas	340
Variabel bernama untuk alur kerja	341
Contoh tag dan hapus alur kerja	341
Gunakan langkah-langkah pemrosesan file khusus	346
Menggunakan beberapa fungsi Lambda secara berurutan	348
Mengakses file setelah pemrosesan kustom	348
Contoh peristiwa dikirim ke AWS Lambda saat file upload	349
Contoh fungsi Lambda untuk langkah alur kerja khusus	350
IAMizin untuk langkah khusus	351
IAMkebijakan untuk alur kerja	352
Hubungan kepercayaan alur kerja	354
Contoh peran eksekusi: Dekripsi, salin, dan tag	354
Contoh peran eksekusi: Jalankan fungsi dan hapus	356

Penanganan pengecualian untuk alur kerja	357
Pantau eksekusi alur kerja	358
CloudWatch logging untuk alur kerja	358
CloudWatch metrik untuk alur kerja	361
Buat alur kerja dari template	361
Menghapus alur kerja dari server Transfer Family	365
Pembatasan dan batasan	366
Mengelola server	369
Lihat daftar server	369
Hapus server	369
Lihat detail SFTP server	371
Lihat detail AS2 server	372
Edit detail server	374
Edit protokol transfer file	377
Edit parameter penyedia identitas khusus	379
Edit titik akhir server	382
Edit pencatatan	383
Edit kebijakan keamanan	383
Mengubah alur kerja terkelola	385
Ubah spanduk tampilan untuk server Anda	386
Menempatkan server Anda secara online atau offline	387
Kelola kunci host server	388
Tambahkan kunci host server tambahan	389
Hapus kunci host server	390
Putar kunci host server	391
Informasi kunci host server tambahan	392
Pantau penggunaan dalam konsol	393
Mengelola kontrol akses	397
Membuat kebijakan akses bucket S3	398
Membuat kebijakan sesi	399
Mencegah pengguna berjalan <code>mkdir</code> di bucket S3	403
Pencatatan log	404
CloudTrail penebangan	404
Mengaktifkan pencatatan CloudTrail	406
Contoh entri log untuk membuat server	406
CloudWatch penebangan	408

Membuat logging untuk server	409
Mengelola logging untuk alur kerja	417
Mengkonfigurasi peran untuk CloudWatch	420
Melihat aliran log Transfer Family	422
Membuat CloudWatch alarm Amazon	426
Mencatat API panggilan S3 ke log akses S3	426
Contoh untuk membatasi masalah wakil yang membingungkan	427
CloudWatch struktur log untuk Transfer Family	429
Contoh entri CloudWatch log	434
Menggunakan CloudWatch metrik	438
Notifikasi pengguna	441
Mengelola acara menggunakan EventBridge	442
Transfer Family acara	443
SFTP,FTPS, dan peristiwa FTP server	443
SFTPaacara konektor	444
Acara A2S	444
Mengirim Transfer Family acara	445
Membuat pola acara	446
Menguji pola acara untuk Transfer Family acara	447
Izin	447
Sumber daya tambahan	447
Referensi detail acara	448
Acara server	449
Acara konektor	453
AS2acara	457
Keamanan	464
Bekerja dengan kebijakan keamanan	466
Algoritma kriptografi	467
TransferSecurityPolicy-2024-01	476
TransferSecurityPolicy-2023-05	477
TransferSecurityPolicy-2022-03	478
TransferSecurityPolicy-2020-06	479
TransferSecurityPolicy-2018-11	480
TransferSecurityPolicy- FIPS -2024-01	481
TransferSecurityPolicy- FIPS -2023-05	482
TransferSecurityPolicy- FIPS -2020-06	484

Pasca kebijakan keamanan Quantum	485
Kebijakan keamanan pasca-Quantum	490
Tentang pertukaran kunci hibrida pasca-kuantum di SSH	491
Cara menggunakannya	491
Bagaimana cara mengujinya	493
Perlindungan data	496
Enkripsi data	498
Manajemen kunci	499
Pengelolaan identitas dan akses	515
Audiens	515
Mengautentikasi dengan identitas	516
Mengelola akses menggunakan kebijakan	520
Bagaimana AWS Transfer Family bekerja dengan IAM	522
Contoh kebijakan berbasis identitas	527
Contoh kebijakan berbasis tanda	530
Pemecahan masalah identitas dan akses	534
Validasi kepatuhan	536
Ketangguhan	537
Keamanan infrastruktur	537
Firewall aplikasi web	538
Pencegahan confused deputy lintas layanan	539
Peran pengguna Transfer Family	541
Peran alur kerja Transfer Family	543
Peran logging/doa Transfer Family	544
AWS kebijakan terkelola	546
AWSTransferConsoleFullAccess	546
AWSTransferFullAccess	548
AWSTransferLoggingAccess	550
AWSTransferReadOnlyAccess	550
Pembaruan kebijakan	551
Pemecahan Masalah Transfer Family	553
Memecahkan masalah pengguna yang dikelola layanan	553
Memecahkan masalah pengguna yang dikelola layanan Amazon EFS	554
Memecahkan masalah badan kunci publik terlalu lama	554
Pemecahan masalah gagal menambahkan kunci publik SSH	555
Memecahkan masalah Amazon API Gateway	555

Terlalu banyak kegagalan otentikasi	555
Koneksi ditutup	557
Memecahkan masalah kebijakan untuk bucket Amazon S3 terenkripsi	557
Memecahkan masalah otentikasi	558
Kegagalan otentikasi—/SSHSFTP	558
Masalah alam tidak cocok AD terkelola	559
Masalah otentikasi lain-lain	559
Memecahkan masalah alur kerja terkelola	559
Memecahkan masalah kesalahan terkait alur kerja menggunakan Amazon CloudWatch	560
Memecahkan masalah kesalahan penyalinan alur kerja	562
Memecahkan masalah dekripsi alur kerja	562
Memecahkan masalah Amazon EFS	564
Memecahkan masalah profil yang hilang POSIX	565
Memecahkan masalah direktori logis dengan Amazon EFS	566
Memecahkan masalah pengujian penyedia identitas Anda	566
Memecahkan masalah menambahkan kunci host tepercaya untuk konektor Anda SFTP	567
Memecahkan masalah pengunggahan file	568
Memecahkan masalah kesalahan unggahan file Amazon S3	568
Memecahkan masalah nama file yang tidak dapat dibaca	568
Memecahkan masalah pengecualian ResourceNotFound	569
Memecahkan masalah SFTP konektor	569
Negosiasi kunci gagal	570
Masalah konektor lain-lain SFTP	570
Memecahkan masalah AS2	571
APIreferensi	572
Selamat datang	572
Tindakan	575
CreateAccess	578
CreateAgreement	585
CreateConnector	591
CreateProfile	598
CreateServer	602
CreateUser	615
CreateWorkflow	624
DeleteAccess	633
DeleteAgreement	636

DeleteCertificate	639
DeleteConnector	641
DeleteHostKey	643
DeleteProfile	646
DeleteServer	648
DeleteSshPublicKey	651
DeleteUser	654
DeleteWorkflow	657
DescribeAccess	659
DescribeAgreement	663
DescribeCertificate	666
DescribeConnector	669
DescribeExecution	672
DescribeHostKey	677
DescribeProfile	680
DescribeSecurityPolicy	683
DescribeServer	687
DescribeUser	692
DescribeWorkflow	697
ImportCertificate	702
ImportHostKey	707
ImportSshPublicKey	711
ListAccesses	716
ListAgreements	720
ListCertificates	724
ListConnectors	728
ListExecutions	731
ListHostKeys	736
ListProfiles	740
ListSecurityPolicies	744
ListServers	748
ListTagsForResource	752
ListUsers	757
ListWorkflows	762
SendWorkflowStepState	765
StartFileTransfer	769

StartServer	775
StopServer	778
TagResource	781
TestConnection	785
TestIdentityProvider	789
UntagResource	796
UpdateAccess	799
UpdateAgreement	806
UpdateCertificate	812
UpdateConnector	816
UpdateHostKey	821
UpdateProfile	825
UpdateServer	828
UpdateUser	841
Tipe Data	848
As2ConnectorConfig	851
CopyStepDetails	855
CustomStepDetails	858
DecryptStepDetails	860
DeleteStepDetails	863
DescribedAccess	865
DescribedAgreement	869
DescribedCertificate	873
DescribedConnector	877
DescribedExecution	881
DescribedHostKey	884
DescribedProfile	887
DescribedSecurityPolicy	890
DescribedServer	892
DescribedUser	901
DescribedWorkflow	905
EfsFileLocation	907
EndpointDetails	909
ExecutionError	913
ExecutionResults	915
ExecutionStepResult	916

FileLocation	918
HomeDirectoryMapEntry	919
IdentityProviderDetails	921
InputFileLocation	924
ListedAccess	925
ListedAgreement	928
ListedCertificate	931
ListedConnector	934
ListedExecution	936
ListedHostKey	938
ListedProfile	940
ListedServer	942
ListedUser	945
ListedWorkflow	948
LoggingConfiguration	950
PosixProfile	952
ProtocolDetails	954
S3FileLocation	958
S3InputFileLocation	960
S3StorageOptions	962
S3Tag	963
ServiceMetadata	964
SftpConnectorConfig	965
SshPublicKey	967
Tag	969
TagStepDetails	970
UserDetails	972
WorkflowDetail	974
WorkflowDetails	976
WorkflowStep	978
Membuat permintaan API	980
Transfer Family membutuhkan header permintaan	980
Transfer Family meminta masukan dan penandatanganan	982
Tanggapan kesalahan	983
Pustaka yang tersedia	985
Parameter Umum	985

Kesalahan Umum	988
Riwayat dokumen	990
Daftar istilah AWS	1003
.....	miv

Apa itu AWS Transfer Family?

AWS Transfer Family adalah layanan transfer aman yang memungkinkan Anda mentransfer file masuk dan keluar dari layanan AWS penyimpanan. Transfer Family adalah bagian dari AWS Cloud platform. AWS Transfer Family menawarkan dukungan yang dikelola sepenuhnya untuk transfer file melalui SFTP, AS2FTPS, dan FTP langsung masuk dan keluar dari Amazon S3 atau Amazon EFS. Anda dapat memigrasi, mengotomatisasi, dan memantau alur kerja transfer file dengan mempertahankan konfigurasi sisi klien yang ada untuk autentikasi, akses, dan firewall—sehingga tidak ada perubahan bagi pelanggan, mitra, dan tim internal Anda, atau aplikasi mereka.

Lihat [Memulai AWS](#) untuk mempelajari lebih lanjut dan mulai membangun aplikasi cloud dengan Amazon Web Services.

AWS Transfer Family mendukung transfer data dari atau ke layanan AWS penyimpanan berikut.

- Penyimpanan Amazon Simple Storage Service (Amazon S3). Untuk informasi tentang Amazon S3, lihat [Memulai Layanan Penyimpanan Sederhana Amazon](#).
- Amazon Elastic File System (AmazonEFS) Sistem File Jaringan (NFS) sistem file. Untuk informasi tentang AmazonEFS, lihat [Apa itu Amazon Elastic File System?](#)

AWS Transfer Family mendukung transfer data melalui protokol berikut:

- Secure Shell (SSH) Protokol Transfer File (SFTP): versi 3
- Protokol Transfer File Aman (FTPS)
- Protokol Transfer File (FTP)
- Pernyataan Penerapan 2 () AS2

Note

Untuk FTP dan koneksi FTPS data, rentang port yang digunakan Transfer Family untuk membuat saluran data adalah 8192-8200.

Protokol transfer file digunakan dalam alur kerja pertukaran data di berbagai industri seperti layanan keuangan, perawatan kesehatan, periklanan, dan ritel, antara lain. Transfer Family menyederhanakan migrasi alur kerja transfer file ke AWS.

Berikut ini adalah beberapa kasus penggunaan umum untuk menggunakan Transfer Family dengan Amazon S3:

- Data digunakan AWS untuk diunggah dari pihak ketiga seperti vendor dan mitra.
- Distribusi data berbasis langganan dengan pelanggan Anda.
- Transfer internal dalam organisasi Anda.

Berikut ini adalah beberapa kasus penggunaan umum untuk menggunakan Transfer Family dengan AmazonEFS:

- Distribusi data
- Rantai pasokan
- Manajemen konten
- Aplikasi penyajian web

Berikut ini adalah beberapa kasus penggunaan umum untuk menggunakan Transfer Family dengan AS2:

- Alur kerja dengan persyaratan kepatuhan yang bergantung pada perlindungan data dan fitur keamanan yang dibangun ke dalam protokol
- Logistik rantai pasokan
- Alur kerja pembayaran
- Business-to-business (B2B) transaksi
- Integrasi dengan sistem perencanaan sumber daya perusahaan (ERP) dan manajemen hubungan pelanggan (CRM)

Dengan Transfer Family, Anda mendapatkan akses ke server berkemampuan protokol transfer file AWS tanpa perlu menjalankan infrastruktur server apa pun. Anda dapat menggunakan layanan ini untuk memigrasikan alur kerja berbasis transfer file AWS sambil mempertahankan klien dan konfigurasi pengguna akhir Anda apa adanya. Pertama-tama Anda mengaitkan nama host Anda dengan titik akhir server, lalu menambahkan pengguna Anda dan menyediakannya dengan tingkat akses yang tepat. Setelah Anda melakukan ini, permintaan transfer pengguna Anda dilayani langsung dari titik akhir server Transfer Family Anda.

Transfer Family memberikan manfaat sebagai berikut:

- Layanan terkelola penuh yang menskalakan secara real time untuk memenuhi kebutuhan Anda.
- Anda tidak perlu memodifikasi aplikasi Anda atau menjalankan infrastruktur protokol transfer file apa pun.
- Dengan data Anda dalam penyimpanan Amazon S3 yang tahan lama, Anda dapat menggunakan native Layanan AWS untuk fungsi pemrosesan, analitik, pelaporan, audit, dan arsip.
- Dengan Amazon EFS sebagai penyimpanan data, Anda mendapatkan sistem file elastis yang dikelola sepenuhnya untuk digunakan dengan AWS Cloud layanan dan sumber daya lokal. Amazon EFS dibangun untuk menskalakan sesuai permintaan ke petabyte tanpa mengganggu aplikasi, tumbuh dan menyusut secara otomatis saat Anda menambahkan dan menghapus file. Ini membantu menghilangkan kebutuhan untuk menyediakan dan mengelola kapasitas untuk mengakomodasi pertumbuhan.
- Layanan Alur Kerja Transfer File tanpa server yang dikelola sepenuhnya yang memudahkan untuk mengatur, menjalankan, mengotomatisasi, dan memantau pemrosesan file yang diunggah menggunakan. AWS Transfer Family
- Tidak ada biaya di muka, dan Anda hanya membayar untuk penggunaan layanan.

Di bagian berikut, Anda dapat menemukan deskripsi tentang berbagai fitur Transfer Family, tutorial memulai, petunjuk terperinci tentang cara mengatur berbagai server yang diaktifkan protokol, cara menggunakan berbagai jenis penyedia identitas, dan API referensi layanan.

Untuk memulai Transfer Family, lihat berikut ini:

- [Bagaimana cara AWS Transfer Family kerja](#)
- [Prasyarat](#)
- [Memulai dengan AWS Transfer Family titik akhir server](#)

Bagaimana cara AWS Transfer Family kerja

AWS Transfer Family adalah AWS layanan terkelola penuh yang dapat Anda gunakan untuk mentransfer file masuk dan keluar dari penyimpanan Amazon Simple Storage Service (Amazon S3) atau sistem file Amazon Elastic File System (EFSAmazon) melalui protokol berikut:

- Secure Shell (SSH) Protokol Transfer File (SFTP): versi 3
- Protokol Transfer File Aman (FTPS)
- Protokol Transfer File (FTP)

- Pernyataan Penerapan 2 () AS2

AWS Transfer Family mendukung hingga 3 Availability Zones dan didukung oleh auto scaling, armada redundan untuk koneksi Anda dan permintaan transfer. Untuk contoh tentang cara membangun redundansi yang lebih tinggi dan meminimalkan latensi jaringan dengan menggunakan perutean berbasis Latensi, lihat posting blog [Minimalkan latensi jaringan](#) dengan transfer Anda untuk server. AWS SFTP

Transfer Family Managed File Transfer Workflows (MFTW) adalah layanan Alur Kerja Transfer File tanpa server yang dikelola sepenuhnya yang memudahkan untuk mengatur, menjalankan, mengotomatisasi, dan memantau pemrosesan file yang diunggah menggunakan. AWS Transfer Family Pelanggan dapat menggunakan MFTW untuk mengotomatisasi berbagai langkah pemrosesan seperti menyalin, menandai, memindai, memfilter, compressing/decompressing, and encrypting/decrypting data yang ditransfer menggunakan Transfer Family. Ini memberikan visibilitas ujung ke ujung untuk pelacakan dan auditabilitas. Untuk detail selengkapnya, lihat [AWS Transfer Family alur kerja terkelola](#).

AWS Transfer Family mendukung klien protokol transfer file standar apa pun. Beberapa klien yang umum digunakan adalah sebagai berikut:

- [Buka SSH](#) — Utilitas baris perintah Macintosh dan Linux.
- [Menang SCP](#) - Klien grafis khusus Windows.
- [Cyberduck](#) — Klien grafis Linux, Macintosh, dan Microsoft Windows.
- [FileZilla](#)— Klien grafis Linux, Macintosh, dan Windows.

Posting blog yang relevan untuk Transfer Family

Tabel berikut mencantumkan posting blog yang berisi informasi berguna bagi pelanggan Transfer Family. Tabel diurutkan dalam urutan kronologis terbalik, sehingga posting terbaru ada di awal tabel.

Judul dan tautan posting blog	Tanggal
Bagaimana Transfer Family dapat membantu Anda membangun solusi transfer file terkelola yang aman dan sesuai	Januari 3, 2024

Judul dan tautan posting blog	Tanggal
Mendeteksi ancaman malware menggunakan AWS Transfer Family	Juli 20, 2023
Memperluas beban SAP kerja dengan AWS Transfer Family	13 Juli 2023
Mengenkripsi dan mendekripsi file dengan dan PGP AWS Transfer Family	Juni 21, 2023
Mengautentikasi AWS Transfer Family dengan Azure Active Directory dan AWS Lambda	15 Desember 2022
Sesuaikan pemberitahuan pengiriman file menggunakan alur kerja AWS Transfer Family terkelola	14 Oktober 2022
Membangun platform transfer file cloud-native menggunakan alur kerja AWS Transfer Family	5 Januari 2022
Mengaktifkan manajemen kunci swalayan pengguna dengan A AWS Transfer Family dan. AWS Lambda	Desember 17, 2021
Tingkatkan kontrol akses data dengan AWS Transfer Family dan Amazon S3	5 Oktober 2021
Meningkatkan throughput untuk pengguna n AWS Global Accelerator dan AWS Transfer Family layanan transfer file yang dihadapi internet	7 Juni 2021
Mengamankan AWS Transfer Family dengan Firewall Aplikasi AWS Web dan Amazon API Gateway	5 Mei 2021

Judul dan tautan posting blog	Tanggal
Mengamankan AWS Transfer Family dengan Firewall Aplikasi AWS Web dan Amazon API Gateway	15 Januari 2021
AWS Transfer Family dukungan untuk Amazon Elastic File System	7 Januari 2021
Aktifkan otentikasi kata sandi untuk menggunakan AWS Transfer FamilyAWS Secrets Manager	5 November 2020
Memusatkan akses data menggunakan AWS Transfer Family dan AWS Storage Gateway	22 Juni 2020
Menggunakan Amazon EFS untuk AWS Lambda aplikasi tanpa server Anda	18 Juni 2020
Gunakan daftar izin IP untuk mengamankan AWS Transfer Family server Anda	8 April 2020
Minimalkan latensi jaringan dengan AWS transfer Anda untuk server SFTP	19 Februari 2020
Angkat dan Pergeseran migrasi SFTP server ke AWS	12 Februari 2020
Sederhanakan AWS SFTP Struktur Anda dengan direktori chroot dan logis	26 September 2019
Menggunakan Okta sebagai penyedia identitas dengan AWS Transfer Family	30 Mei 2019

Prasyarat

Bagian berikut menjelaskan prasyarat yang diperlukan untuk menggunakan layanan ini. AWS Transfer Family Minimal, Anda harus membuat bucket Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) dan menyediakan akses ke bucket tersebut melalui AWS Identity and Access Management peran IAM (). Peran Anda juga perlu membangun hubungan kepercayaan. Hubungan kepercayaan ini memungkinkan Transfer Family untuk mengambil IAM peran untuk mengakses bucket Anda sehingga dapat melayani permintaan transfer file pengguna Anda.

Topik

- [AWS Wilayah, titik akhir, dan kuota yang didukung](#)
- [Mendaftar untuk AWS](#)
- [Konfigurasi penyimpanan untuk digunakan AWS Transfer Family](#)
- [Buat IAM peran dan kebijakan](#)

AWS Wilayah, titik akhir, dan kuota yang didukung

Untuk terhubung secara terprogram ke AWS layanan, Anda menggunakan titik akhir. Misalnya, titik akhir untuk pelanggan di wilayah AS Timur (Ohio) (us-east-2), adalah `transfer.us-east-2.amazonaws.com`. Kuota layanan, juga disebut sebagai batas, adalah jumlah maksimum sumber daya layanan atau operasi untuk Akun AWS. Dalam panduan ini, Anda dapat menemukan kuota di [AS2kuota](#) dan [Kuota untuk konektor SFTP](#).

Untuk informasi selengkapnya tentang AWS Wilayah, titik akhir, dan kuota layanan yang didukung, lihat [AWS Transfer Family titik akhir dan kuota](#) di Referensi Umum Amazon Web

Mendaftar untuk AWS

Saat Anda mendaftar ke Amazon Web Services (AWS), AWS akun Anda secara otomatis mendaftar untuk semua layanan AWS, termasuk AWS Transfer Family. Anda hanya membayar biaya layanan yang Anda gunakan.

Jika Anda sudah memiliki AWS akun, lompat ke tugas berikutnya. Jika Anda belum memiliki akun AWS, gunakan prosedur berikut untuk membuatnya.

Jika Anda tidak memiliki Akun AWS, selesaikan langkah-langkah berikut untuk membuatnya.

Untuk mendaftar untuk Akun AWS

1. Buka <https://portal.aws.amazon.com/billing/pendaftaran>.
2. Ikuti petunjuk online.

Bagian dari prosedur pendaftaran melibatkan tindakan menerima panggilan telepon dan memasukkan kode verifikasi di keypad telepon.

Saat Anda mendaftar untuk sebuah Akun AWS, sebuah Pengguna root akun AWS dibuat. Pengguna root memiliki akses ke semua Layanan AWS dan sumber daya di akun. Sebagai praktik terbaik keamanan, [tetapkan akses administratif ke pengguna administratif](#), dan hanya gunakan pengguna root untuk melakukan [tugas-tugas yang memerlukan akses pengguna root](#).

Untuk informasi tentang harga dan penggunaan AWS Kalkulator Harga untuk mendapatkan perkiraan biaya penggunaan Transfer Family, lihat [AWS Transfer Family harga](#).

Untuk informasi tentang ketersediaan AWS Wilayah, lihat [AWS Transfer Family titik akhir dan kuota](#) di. Referensi Umum AWS

Konfigurasikan penyimpanan untuk digunakan AWS Transfer Family

Topik ini menjelaskan opsi penyimpanan yang dapat Anda gunakan AWS Transfer Family. Anda dapat menggunakan Amazon S3 atau Amazon EFS sebagai penyimpanan untuk server Transfer Family Anda.

Daftar Isi

- [Konfigurasikan bucket Amazon S3](#)
 - [Titik akses Amazon S3](#)
 - [Perilaku Amazon S3 HeadObject](#)
 - [Memberikan kemampuan untuk hanya menulis dan membuat daftar file](#)
 - [Sejumlah besar objek nol-byte menyebabkan masalah latensi](#)
- [Konfigurasikan sistem EFS file Amazon](#)
 - [Kepemilikan EFS file Amazon](#)
 - [Mengatur EFS pengguna Amazon untuk Transfer Family](#)

- [Konfigurasi pengguna Transfer Family di Amazon EFS](#)
- [Buat pengguna EFS root Amazon](#)
- [EFS Perintah Amazon yang didukung](#)

Konfigurasi bucket Amazon S3

AWS Transfer Family mengakses bucket Amazon S3 untuk melayani permintaan transfer pengguna, jadi Anda perlu menyediakan bucket Amazon S3 sebagai bagian dari pengaturan server yang mendukung protokol transfer file Anda. Anda dapat menggunakan bucket yang sudah ada, atau Anda dapat membuat yang baru.

Note

Anda tidak harus menggunakan server dan bucket Amazon S3 yang berada di AWS Wilayah yang sama, tetapi kami merekomendasikan ini sebagai praktik terbaik.

Saat Anda mengatur pengguna, Anda menetapkan masing-masing IAM peran kepada mereka. Peran ini menentukan tingkat akses yang mereka miliki ke bucket Amazon S3 Anda.

Untuk informasi tentang cara membuat bucket baru, lihat [Bagaimana cara membuat bucket S3?](#) di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

Note

Anda dapat menggunakan Amazon S3 Object Lock untuk mencegah objek ditimpa untuk jangka waktu yang tetap atau tanpa batas waktu. Ini bekerja dengan cara yang sama dengan Transfer Family seperti layanan lainnya. Jika suatu objek ada dan dilindungi, menulis ke file itu atau menghapusnya tidak diperbolehkan. Untuk detail selengkapnya tentang Kunci Objek Amazon S3, lihat [Menggunakan Kunci Objek Amazon S3 di Panduan](#) Pengguna Layanan Penyimpanan Sederhana Amazon.

Titik akses Amazon S3

AWS Transfer Family mendukung [Poin Akses Amazon S3](#), fitur Amazon S3 yang memungkinkan Anda mengelola akses granular ke kumpulan data bersama dengan mudah. Anda dapat

menggunakan alias S3 Access Point di mana pun Anda menggunakan nama bucket S3. Anda dapat membuat ratusan titik akses di Amazon S3 untuk pengguna yang memiliki izin berbeda untuk mengakses data bersama di bucket Amazon S3.

Misalnya, Anda dapat menggunakan titik akses untuk memungkinkan tiga tim berbeda memiliki akses ke kumpulan data bersama yang sama di mana satu tim dapat membaca data dari S3, tim kedua dapat menulis data ke S3, dan tim ketiga dapat membaca, menulis, dan menghapus data dari S3. Untuk menerapkan kontrol akses granular seperti yang disebutkan di atas, Anda dapat membuat titik akses S3 yang berisi kebijakan yang memberikan akses asimetris ke tim yang berbeda. Anda dapat menggunakan titik akses S3 dengan server Transfer Family Anda untuk mencapai kontrol akses yang halus, tanpa membuat kebijakan bucket S3 kompleks yang mencakup ratusan kasus penggunaan. Untuk mempelajari lebih lanjut tentang cara menggunakan titik akses S3 dengan server Transfer Family, lihat [Tingkatkan kontrol akses data dengan AWS Transfer Family dan posting blog Amazon S3](#).

 Note

AWS Transfer Family saat ini tidak mendukung Titik Akses Multi-Wilayah Amazon S3.

Perilaku Amazon S3 HeadObject

 Note

Saat membuat atau memperbarui server Transfer Family, Anda dapat mengoptimalkan kinerja untuk direktori Amazon S3, yang menghilangkan panggilan. `HeadObject`

Di Amazon S3, bucket dan objek adalah sumber daya utama, dan objek disimpan dalam bucket. Amazon S3 dapat meniru sistem file hierarkis, tetapi terkadang dapat berperilaku berbeda dari sistem file biasa. Misalnya, direktori bukan konsep kelas satu di Amazon S3 tetapi didasarkan pada kunci objek. AWS Transfer Family menyimpulkan jalur direktori dengan memisahkan kunci objek dengan karakter garis miring maju (/), memperlakukan elemen terakhir sebagai nama file, lalu mengelompokkan nama file yang memiliki awalan yang sama bersama-sama di bawah jalur yang sama. Objek nol-byte dibuat untuk mewakili jalur folder saat Anda membuat direktori kosong menggunakan `mkdir` atau dengan menggunakan konsol Amazon S3. Kunci untuk benda-benda ini berakhir dengan garis miring ke depan. Objek nol-byte ini dijelaskan dalam [Mengatur objek di konsol Amazon S3 menggunakan folder di Panduan Pengguna Amazon S3](#).

Saat Anda menjalankan `ls` perintah, dan beberapa hasilnya adalah objek zero-byte Amazon S3 (objek ini memiliki kunci yang diakhiri dengan karakter garis miring ke depan), Transfer Family mengeluarkan `HeadObject` permintaan untuk masing-masing objek ini (lihat di Referensi Layanan Penyimpanan Sederhana Amazon untuk [HeadObject](#) detailnya). API Hal ini dapat mengakibatkan masalah berikut saat menggunakan Amazon S3 sebagai penyimpanan Anda dengan Transfer Family.

Memberikan kemampuan untuk hanya menulis dan membuat daftar file

Dalam beberapa kasus, Anda mungkin hanya ingin menawarkan akses tulis ke objek Amazon S3 Anda. Misalnya, Anda mungkin ingin menyediakan akses untuk menulis (atau mengunggah) dan mencantumkan objek dalam ember, tetapi tidak untuk membaca (mengunduh) objek. Untuk melakukan `ls` dan `mkdir` memerintahkan dengan menggunakan klien transfer file, Anda harus memiliki Amazon S3 `ListObjects` dan `PutObject` izin. Namun, ketika Transfer Family perlu melakukan `HeadObject` panggilan untuk menulis atau membuat daftar file, panggilan gagal dengan kesalahan Akses ditolak, karena panggilan ini memerlukan `GetObject` izin.

 Note

Saat membuat atau memperbarui server Transfer Family, Anda dapat mengoptimalkan kinerja untuk direktori Amazon S3, yang menghilangkan panggilan. `HeadObject`

Dalam hal ini, Anda dapat memberikan akses dengan menambahkan kondisi kebijakan AWS Identity and Access Management (IAM) yang menambahkan `GetObject` izin hanya untuk objek yang diakhiri dengan garis miring (/). Kondisi ini mencegah `GetObject` panggilan pada file (sehingga tidak dapat dibaca), tetapi memungkinkan pengguna untuk membuat daftar dan melintasi folder. Kebijakan contoh berikut hanya menawarkan akses tulis dan daftar ke bucket Amazon S3 Anda. Untuk menggunakan kebijakan ini, ganti ***DOC-EXAMPLE-BUCKET*** dengan nama bucket Anda.

```
{
  "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

Kebijakan ini tidak mengizinkan pengguna untuk menambahkan file. Dengan kata lain, pengguna yang diberi kebijakan ini tidak dapat membuka file untuk menambahkan konten ke dalamnya, atau memodifikasinya. Selain itu, jika kasus penggunaan Anda memerlukan HeadObject panggilan sebelum mengunggah file, kebijakan ini tidak akan berfungsi untuk Anda.

Sejumlah besar objek nol-byte menyebabkan masalah latensi

Jika bucket Amazon S3 Anda berisi sejumlah besar objek nol-byte ini, Transfer Family mengeluarkan banyak panggilan, yang dapat mengakibatkan HeadObject penundaan pemrosesan.

Salah satu solusi yang mungkin untuk masalah ini adalah menghapus semua objek zero-byte Anda. Perhatikan hal berikut:

- Direktori kosong tidak akan ada lagi. Direktori hanya ada karena nama mereka berada di kunci suatu objek.
- Tidak mencegah seseorang menelepon `mkdir` dan merusak semuanya lagi. Anda dapat mengurangi ini dengan membuat kebijakan yang mencegah pembuatan direktori.
- Beberapa skenario menggunakan objek 0-byte ini. Misalnya, Anda memiliki struktur seperti `/inboxes/customer1000` dan direktori kotak masuk dibersihkan setiap hari.

Solusi lain yang mungkin adalah membatasi jumlah objek yang terlihat melalui kondisi kebijakan untuk mengurangi jumlah `HeadObject` panggilan. Agar ini menjadi solusi yang bisa diterapkan, Anda harus menerima bahwa Anda mungkin hanya dapat melihat sekumpulan terbatas semua sub-direktori Anda.

Konfigurasi sistem EFS file Amazon

AWS Transfer Family mengakses Amazon Elastic File System (AmazonEFS) untuk melayani permintaan transfer pengguna Anda. Jadi, Anda harus menyediakan sistem EFS file Amazon sebagai bagian dari pengaturan server yang mendukung protokol transfer file Anda. Anda dapat menggunakan sistem file yang ada, atau Anda dapat membuat yang baru.

Perhatikan hal berikut:

- Saat Anda menggunakan server Transfer Family dan sistem EFS file Amazon, server dan sistem file harus sama Wilayah AWS.
- Server dan sistem file tidak perlu berada di akun yang sama. Jika server dan sistem file tidak berada dalam akun yang sama, kebijakan sistem file harus memberikan izin eksplisit untuk peran pengguna.

Untuk informasi tentang cara menyiapkan beberapa akun, lihat [Mengelola AWS akun di organisasi Anda](#) di Panduan AWS Organizations Pengguna.

- Saat Anda mengatur pengguna, Anda menetapkan masing-masing IAM peran kepada mereka. Peran ini menentukan tingkat akses yang mereka miliki ke sistem EFS file Amazon Anda.
- Untuk detail tentang pemasangan sistem EFS file Amazon, lihat [Memasang sistem EFS file Amazon](#).

Untuk detail selengkapnya tentang cara AWS Transfer Family dan Amazon EFS bekerja sama, lihat [Menggunakan AWS Transfer Family untuk mengakses file di sistem EFS file Amazon Anda](#) di Panduan Pengguna Amazon Elastic File System.

Kepemilikan EFS file Amazon

Amazon EFS menggunakan model izin file Antarmuka Sistem Operasi Portabel (POSIX) untuk mewakili kepemilikan file.

Dalam POSIX, pengguna dalam sistem dikategorikan ke dalam tiga kelas izin yang berbeda: Ketika Anda mengizinkan pengguna untuk mengakses file yang disimpan dalam sistem EFS file Amazon menggunakan AWS Transfer Family, Anda harus menetapkan mereka “POSIXprofil.” Profil ini digunakan untuk menentukan akses mereka ke file dan direktori di sistem EFS file Amazon.

- User (u): Pemilik file atau direktori. Biasanya, pencipta file atau direktori juga pemiliknya.
- Grup (g): Kumpulan pengguna yang membutuhkan akses identik ke file dan direktori yang mereka bagikan.
- Lainnya (o): Semua pengguna lain yang memiliki akses ke sistem kecuali pemilik dan anggota grup. Kelas izin ini juga disebut sebagai “Publik.”

Dalam model POSIX izin, setiap objek sistem file (file, direktori, tautan simbolis, pipa bernama, dan soket) dikaitkan dengan tiga set izin yang disebutkan sebelumnya. EFS Objek Amazon memiliki mode gaya Unix yang terkait dengannya. Nilai mode ini mendefinisikan izin untuk melakukan tindakan pada objek tersebut.

Selain itu, pada sistem bergaya Unix, pengguna dan grup dipetakan ke pengidentifikasi numerik, yang digunakan EFS Amazon untuk mewakili kepemilikan file. Untuk AmazonEFS, objek dimiliki oleh satu pemilik dan satu grup. Amazon EFS menggunakan numerik yang IDs dipetakan untuk memeriksa izin saat pengguna mencoba mengakses objek sistem file.

Mengatur EFS pengguna Amazon untuk Transfer Family

Sebelum mengatur EFS pengguna Amazon, Anda dapat melakukan salah satu hal berikut:

- Anda dapat membuat pengguna dan mengatur folder rumah mereka di AmazonEFS. Lihat [Konfigurasi pengguna Transfer Family di Amazon EFS](#) untuk detail.
- Jika Anda merasa nyaman menambahkan pengguna root, Anda bisa [Buat pengguna EFS root Amazon](#).

 Note

Server Transfer Family tidak mendukung EFS jalur akses Amazon untuk menetapkan POSIX izin. POSIXProfil pengguna Transfer Family (dijelaskan di bagian sebelumnya) menawarkan kemampuan untuk mengatur izin. POSIX Izin ini ditetapkan pada tingkat pengguna, untuk akses granular, berdasarkan UIDGID, dan sekunder. GIDs

Konfigurasi pengguna Transfer Family di Amazon EFS

Transfer Family memetakan pengguna ke yang UID/GID and directories you specify. If the UID/GID/ directories belum adaEFS, maka Anda harus membuatnya sebelum menetapkan di Transfer ke pengguna. Detail untuk membuat EFS pengguna Amazon dijelaskan dalam [Bekerja dengan pengguna, grup, dan izin di Tingkat Sistem File Jaringan \(NFS\)](#) di Panduan Pengguna Amazon Elastic File System.

Langkah-langkah untuk mengatur EFS pengguna Amazon di Transfer Family

1. Petakan EFS UID dan GID untuk pengguna Anda di Transfer Family menggunakan [PosixProfile](#)bidang.
2. Jika Anda ingin pengguna memulai di folder tertentu saat login, Anda dapat menentukan EFS direktori di bawah [HomeDirectory](#)bidang.

Anda dapat mengotomatiskan proses, dengan menggunakan CloudWatch aturan dan fungsi Lambda. Sebagai contoh, fungsi Lambda yang berinteraksi dengannyaEFS, lihat Menggunakan [Amazon EFS untuk AWS Lambda aplikasi tanpa server Anda](#).

Selain itu, Anda dapat mengonfigurasi direktori logis untuk pengguna Transfer Family Anda. Untuk detailnya, lihat [Konfigurasi direktori logis untuk Amazon EFS](#) bagian dalam [Menggunakan direktori logis untuk menyederhanakan struktur direktori Transfer Family](#) topik.

Buat pengguna EFS root Amazon

Jika organisasi Anda merasa nyaman bagi Anda untuk mengaktifkan akses pengguna root viaSFTP/ FTPSuntuk konfigurasi pengguna Anda, Anda dapat membuat pengguna GID yang 0 (pengguna root), lalu gunakan pengguna root itu untuk membuat folder dan menetapkan pemilik POSIX ID untuk pengguna lainnya. UID Keuntungan dari opsi ini adalah tidak perlu memasang sistem EFS file Amazon.

Lakukan langkah-langkah yang dijelaskan dalam [Menambahkan pengguna yang EFS dikelola layanan Amazon](#), dan untuk ID Pengguna dan ID Grup, masukkan 0 (nol).

EFSPerintah Amazon yang didukung

Perintah berikut didukung untuk Amazon EFS untuk AWS Transfer Family.

- `cd`
- `ls/dir`
- `pwd`
- `put`
- `get`
- `rename`
- `chown`: Hanya root (yaitu, pengguna dengan uid=0) yang dapat mengubah kepemilikan dan izin file dan direktori.
- `chmod`: Hanya root yang dapat mengubah kepemilikan dan izin file dan direktori.
- `chgrp`: Didukung baik untuk root atau untuk pemilik file yang hanya dapat mengubah grup file menjadi salah satu grup sekunder mereka.
- `ln -s/symlink`
- `mkdir`
- `rm/delete`
- `rmdir`
- `chmtime`

Buat IAM peran dan kebijakan

Topik ini menjelaskan jenis kebijakan dan peran yang dapat digunakan AWS Transfer Family, dan berjalan melalui proses pembuatan peran pengguna. Ini juga menjelaskan cara kerja kebijakan sesi dan memberikan contoh peran pengguna.

AWS Transfer Family menggunakan jenis peran berikut:

- Peran pengguna — Memungkinkan pengguna yang dikelola layanan mengakses sumber daya Transfer Family yang diperlukan. AWS Transfer Family mengambil peran ini dalam konteks pengguna ARN Transfer Family.

- Peran akses - Menyediakan akses hanya ke file Amazon S3 yang sedang ditransfer. Untuk AS2 transfer masuk, peran akses menggunakan Amazon Resource Name (ARN) untuk perjanjian. Untuk AS2 transfer keluar, peran akses menggunakan konektor ARN untuk.
- Peran pemanggilan — Untuk digunakan dengan Amazon API Gateway sebagai penyedia identitas kustom server. Transfer Family mengambil peran ini dalam konteks server ARN Transfer Family.
- Peran logging - Digunakan untuk log entri ke Amazon CloudWatch. Transfer Family menggunakan peran ini untuk mencatat detail keberhasilan dan kegagalan bersama dengan informasi tentang transfer file. Transfer Family mengambil peran ini dalam konteks server ARN Transfer Family. Untuk AS2 transfer keluar, peran logging menggunakan konektor ARN.
- Peran eksekusi — Memungkinkan pengguna Transfer Family memanggil dan meluncurkan alur kerja. Transfer Family mengasumsikan peran ini dalam konteks alur kerja ARN Transfer Family.

Selain peran ini, Anda juga dapat menggunakan kebijakan sesi. Kebijakan sesi digunakan untuk membatasi akses bila diperlukan. Perhatikan bahwa kebijakan ini berdiri sendiri: artinya, Anda tidak menambahkan kebijakan ini ke peran. Sebaliknya, Anda menambahkan kebijakan sesi langsung ke pengguna Transfer Family.

Note

Saat membuat pengguna Transfer Family yang dikelola layanan, Anda dapat memilih Kebijakan buat otomatis berdasarkan folder beranda. Ini adalah pintasan yang berguna jika Anda ingin membatasi akses pengguna ke folder mereka sendiri. Selain itu, Anda dapat melihat detail tentang kebijakan sesi dan contoh di [Bagaimana kebijakan sesi bekerja](#). Anda juga dapat menemukan informasi selengkapnya tentang kebijakan [sesi dalam kebijakan Sesi](#) di Panduan IAM Pengguna.

Topik

- [Membuat peran pengguna](#)
- [Bagaimana kebijakan sesi bekerja](#)
- [Contoh kebijakan akses baca/tulis](#)

Membuat peran pengguna

Saat Anda membuat pengguna, Anda membuat sejumlah keputusan tentang akses pengguna. Keputusan ini termasuk bucket Amazon S3 atau sistem EFS file Amazon mana yang dapat diakses pengguna, bagian mana dari setiap bucket Amazon S3 dan file mana dalam sistem file yang dapat diakses, dan izin apa yang dimiliki pengguna (misalnya, `PUT` `GET`).

Untuk menetapkan akses, Anda membuat kebijakan dan peran berbasis identitas AWS Identity and Access Management (IAM) yang menyediakan informasi akses tersebut. Sebagai bagian dari proses ini, Anda menyediakan akses bagi pengguna Anda ke bucket Amazon S3 atau sistem EFS file Amazon yang merupakan target atau sumber untuk operasi file. Untuk melakukan ini, ambil langkah-langkah tingkat tinggi berikut, yang dijelaskan secara rinci nanti:

Membuat peran pengguna

1. Buat IAM kebijakan untuk AWS Transfer Family. Ini dijelaskan dalam [Untuk membuat IAM kebijakan untuk AWS Transfer Family](#).
2. Buat IAM peran dan lampirkan IAM kebijakan baru. Sebagai contoh, lihat [Contoh kebijakan akses baca/tulis](#).
3. Membangun hubungan kepercayaan antara AWS Transfer Family dan IAM peran. Ini dijelaskan dalam [Untuk membangun hubungan kepercayaan](#).

Prosedur berikut menjelaskan cara membuat IAM kebijakan dan peran.

Untuk membuat IAM kebijakan untuk AWS Transfer Family

1. Buka IAM konsol di <https://console.aws.amazon.com/iam/>.
2. Di panel navigasi, pilih Kebijakan dan kemudian pilih Buat kebijakan.
3. Pada halaman Buat Kebijakan, pilih JSON tab.
4. Di editor yang muncul, ganti konten editor dengan IAM kebijakan yang ingin Anda lampirkan ke IAM peran.

Anda dapat memberikan akses baca/tulis atau membatasi pengguna ke direktori home mereka. Untuk informasi selengkapnya, lihat [Contoh kebijakan akses baca/tulis](#).

5. Pilih Kebijakan tinjauan dan berikan nama dan deskripsi untuk kebijakan Anda, lalu pilih Buat kebijakan.

Selanjutnya, Anda membuat IAM peran dan melampirkan IAM kebijakan baru padanya.

Untuk membuat IAM peran untuk AWS Transfer Family

1. Di panel navigasi, pilih Peran, lalu pilih Buat peran.

Pada halaman Buat peran, pastikan bahwa AWS layanan dipilih.

2. Pilih Transfer dari daftar layanan, lalu pilih Berikutnya: Izin. Ini membangun hubungan kepercayaan antara AWS Transfer Family dan AWS.
3. Di bagian Lampirkan kebijakan izin, cari dan pilih kebijakan yang baru saja Anda buat, lalu pilih Berikutnya: Tag.
4. (Opsional) Masukkan kunci dan nilai untuk tag, dan pilih Berikutnya: Tinjau.
5. Pada halaman Tinjauan, masukkan nama dan deskripsi untuk peran baru Anda, lalu pilih Buat peran.

Selanjutnya, Anda membangun hubungan kepercayaan antara AWS Transfer Family dan AWS.

Untuk membangun hubungan kepercayaan

Note

Dalam contoh kami, kami menggunakan keduanya `ArnLike` dan `ArnEquals`. Mereka identik secara fungsional, dan oleh karena itu Anda dapat menggunakan keduanya ketika Anda membuat kebijakan Anda. Dokumentasi Transfer Family digunakan `ArnLike` ketika kondisi berisi karakter wildcard, dan `ArnEquals` untuk menunjukkan kondisi kecocokan yang tepat.

1. Di IAM konsol, pilih peran yang baru saja Anda buat.
2. Pada halaman Ringkasan, pilih Trust relationship, lalu pilih Edit trust relationship.
3. Di editor Edit Trust Relationship, pastikan layanannya `"transfer.amazonaws.com"`. Kebijakan akses ditampilkan sebagai berikut.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
```

```

    "Service": "transfer.amazonaws.com"
  },
  "Action": "sts:AssumeRole"
}
]
}

```

Kami menyarankan Anda menggunakan kunci `aws:SourceAccount` dan `aws:SourceArn` kondisi untuk melindungi diri Anda dari masalah wakil yang membingungkan. Akun sumber adalah pemilik server dan sumbernya ARN ARN adalah pengguna. Sebagai contoh:

```

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

```

Anda juga dapat menggunakan `ArnLike` kondisi ini jika Anda ingin membatasi ke server tertentu, bukan server apa pun di akun pengguna. Sebagai contoh:

```

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

```

Note

Dalam contoh di atas, ganti masing-masing *user input placeholder* dengan informasi Anda sendiri.

Untuk detail tentang masalah wakil yang membingungkan dan lebih banyak contoh, lihat [Pencegahan confused deputy lintas layanan](#).

4. Pilih Perbarui Kebijakan Kepercayaan untuk memperbarui kebijakan akses.

Anda sekarang telah menciptakan IAM peran yang memungkinkan AWS Transfer Family untuk memanggil AWS layanan atas nama Anda. Anda melampirkan peran IAM kebijakan yang Anda buat untuk memberikan akses ke pengguna Anda. Di [Memulai dengan AWS Transfer Family titik akhir server](#) bagian ini, peran dan kebijakan ini ditetapkan untuk pengguna atau pengguna Anda.

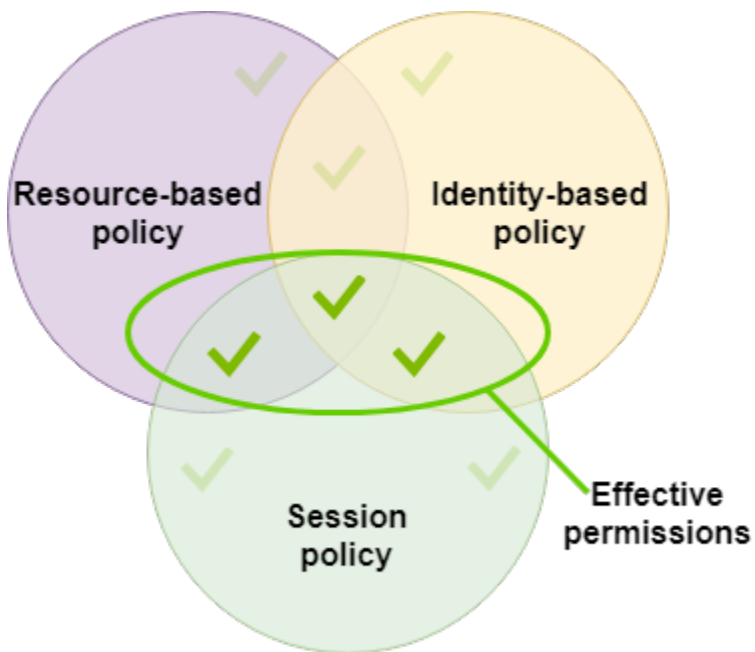
Lihat juga

- Untuk informasi umum selengkapnya tentang IAM peran, lihat [Membuat peran untuk mendelegasikan izin ke AWS layanan](#) di IAMPanduan Pengguna.
- Untuk mempelajari selengkapnya tentang kebijakan berbasis identitas untuk sumber daya Amazon S3, lihat Manajemen [identitas dan akses di Amazon S3 di](#) Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.
- Untuk mempelajari lebih lanjut tentang kebijakan berbasis identitas untuk EFS sumber daya Amazon, lihat [Menggunakan IAM untuk mengontrol akses data sistem file](#) di Panduan Pengguna Amazon Elastic File System.

Bagaimana kebijakan sesi bekerja

Saat administrator membuat peran, peran tersebut sering menyertakan izin luas untuk mencakup beberapa kasus penggunaan atau anggota tim. Jika administrator mengonfigurasi [konsol URL](#), mereka dapat mengurangi izin untuk sesi yang dihasilkan dengan menggunakan kebijakan sesi. Misalnya, jika Anda membuat peran dengan [akses baca/tulis](#), Anda dapat mengatur URL yang membatasi akses pengguna hanya ke direktori beranda mereka.

Kebijakan sesi adalah kebijakan lanjutan yang Anda berikan sebagai parameter saat Anda membuat sesi sementara untuk peran atau pengguna secara terprogram. Kebijakan sesi berguna untuk mengunci pengguna sehingga mereka hanya memiliki akses ke bagian bucket Anda di mana awalan objek berisi nama pengguna mereka. Diagram berikut menunjukkan bahwa izin kebijakan sesi adalah persimpangan kebijakan sesi dan kebijakan berbasis sumber daya ditambah persimpangan kebijakan sesi dan kebijakan berbasis identitas.



Untuk detail selengkapnya, lihat [Kebijakan sesi](#) di Panduan IAM Pengguna.

Di AWS Transfer Family, kebijakan sesi hanya didukung saat Anda mentransfer ke atau dari Amazon S3. Contoh kebijakan berikut adalah kebijakan sesi yang membatasi akses pengguna ke home direktori mereka saja. Perhatikan hal berikut:

- PutObjectACL Pernyataan GetObjectACL dan pernyataan hanya diperlukan jika Anda perlu mengaktifkan Akses Lintas Akun. Artinya, server Transfer Family Anda perlu mengakses bucket di akun yang berbeda.
- Panjang maksimum kebijakan sesi adalah 2048 karakter. Untuk detail selengkapnya, lihat [parameter Permintaan kebijakan](#) untuk CreateUser tindakan dalam API referensi.
- Jika bucket Amazon S3 dienkripsi menggunakan AWS Key Management Service (AWS KMS), Anda harus menentukan izin tambahan dalam kebijakan Anda. Untuk detailnya, lihat [Enkripsi data di 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

Contoh kebijakan sebelumnya mengasumsikan bahwa pengguna memiliki direktori home mereka diatur untuk menyertakan garis miring, untuk menandakan bahwa itu adalah direktori. Jika, di sisi lain, Anda menetapkan pengguna HomeDirectory tanpa garis miring, maka Anda harus memasukkannya sebagai bagian dari kebijakan Anda.

Dalam contoh kebijakan sebelumnya, perhatikan penggunaan parameter `transfer:HomeFolder`, `transfer:HomeBucket`, dan `transfer:HomeDirectory` kebijakan. Parameter ini diatur untuk HomeDirectory yang dikonfigurasi untuk pengguna, seperti

yang dijelaskan dalam [HomeDirectory](#) dan [Menerapkan metode API Gateway Anda](#). Parameter ini memiliki definisi berikut:

- `transfer:HomeBucketParameter` diganti dengan komponen pertama dari `HomeDirectory`.
- `transfer:HomeFolderParameter` diganti dengan bagian `HomeDirectory` parameter yang tersisa.
- `transfer:HomeDirectoryParameter` memiliki garis miring depan (/) dihapus sehingga dapat digunakan sebagai bagian dari S3 Amazon Resource Name (ARN) dalam sebuah Resource pernyataan.

Note

Jika Anda menggunakan direktori logik—yaitu, pengguna adalah LOGICAL —parameter kebijakan ini (`HomeBucket`, `HomeDirectory`, dan `HomeFolder`) tidak didukung.

`homeDirectoryType`

Misalnya, asumsikan bahwa `HomeDirectory` parameter yang dikonfigurasi untuk pengguna Transfer Family adalah `/home/bob/amazon/stuff/`.

- `transfer:HomeBucket` diatur ke `/home`.
- `transfer:HomeFolder` diatur ke `/bob/amazon/stuff/`.
- `transfer:HomeDirectory` menjadi `home/bob/amazon/stuff/`.

Yang pertama "Sid" memungkinkan pengguna untuk membuat daftar semua direktori mulai dari `/home/bob/amazon/stuff/`.

Yang kedua "Sid" membatasi pengguna put dan get akses ke jalur yang sama, `/home/bob/amazon/stuff/`.

Contoh kebijakan akses baca/tulis

Berikan akses baca/tulis ke bucket Amazon S3

Contoh kebijakan berikut untuk AWS Transfer Family memberikan akses baca/tulis ke objek di bucket Amazon S3 Anda.

Perhatikan hal berikut:

- Ganti **DOC-EXAMPLE-BUCKET** dengan nama keranjang Amazon S3.
- PutObjectACL Pernyataan GetObjectACL dan pernyataan hanya diperlukan jika Anda perlu mengaktifkan Akses Lintas Akun. Artinya, server Transfer Family Anda perlu mengakses bucket di akun yang berbeda.
- DeleteObjectVersion Pernyataan GetObjectVersion dan hanya diperlukan jika pembuatan versi diaktifkan di bucket Amazon S3 yang sedang diakses.

 Note

Jika Anda pernah mengaktifkan pembuatan versi untuk bucket, Anda memerlukan izin ini, karena Anda hanya dapat menanggukkan pembuatan versi di Amazon S3, dan tidak mematikannya sepenuhnya. Untuk detailnya, lihat Bucket [yang tidak berversi, berkemampuan versi, dan ditanggukkan versi](#).

```
{
  "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/*"
  }
]
}

```

Berikan akses sistem file ke file dalam sistem EFS file Amazon

Note

Selain kebijakan, Anda juga harus memastikan izin POSIX file Anda memberikan akses yang sesuai. Untuk informasi selengkapnya, lihat [Bekerja dengan pengguna, grup, dan izin di Tingkat Sistem File Jaringan \(NFS\)](#) di Panduan Pengguna Amazon Elastic File System.

Contoh kebijakan berikut memberikan akses sistem file root ke file di sistem EFS file Amazon Anda.

Note

Dalam contoh berikut, ganti *region* dengan wilayah Anda, *account-id* dengan akun file tersebut masuk, dan *file-system-id* dengan ID Amazon Elastic File System (AmazonEFS) Anda.

```

{
  "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"
    }
  ]
}

```

Contoh kebijakan berikut memberikan akses sistem file pengguna ke file di sistem EFS file Amazon Anda.

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

Tutorial Transfer Family

Panduan AWS Transfer Family pengguna menyediakan penelusuran terperinci untuk beberapa kasus penggunaan.

- [Memulai dengan AWS Transfer Family titik akhir server](#): tutorial ini memandu Anda melalui pembuatan server SFTP Transfer Family dan pengguna yang dikelola layanan, lalu menunjukkan cara mentransfer file menggunakan klien.
- [Menyiapkan dan menggunakan SFTP konektor](#): tutorial ini menggambarkan cara mengatur SFTP konektor, dan kemudian mentransfer file antara penyimpanan Amazon S3 dan SFTP server.
- [Menyiapkan metode Amazon API Gateway sebagai penyedia identitas khusus](#) : tutorial ini menggambarkan cara mengatur metode Amazon API Gateway dan menggunakannya sebagai penyedia identitas khusus untuk mengunggah file ke AWS Transfer Family server.
- [Menyiapkan alur kerja terkelola untuk mendekripsi file](#): tutorial ini menggambarkan cara mengatur alur kerja terkelola yang berisi langkah dekripsi, dan cara mengunggah file terenkripsi ke bucket Amazon S3 dan kemudian melihat file yang didekripsi.
- [Menyiapkan AS2 konfigurasi](#): tutorial ini berjalan melalui langkah-langkah yang diperlukan untuk mengkonfigurasi server AS2 Transfer Family. Ada instruksi untuk mengimpor sertifikat, membuat profil dan perjanjian, secara opsional membuat AS2 konektor, dan kemudian menguji konfigurasi.

Topik

- [Memulai dengan AWS Transfer Family titik akhir server](#)
- [Menyiapkan alur kerja terkelola untuk mendekripsi file](#)
- [Menyiapkan dan menggunakan SFTP konektor](#)
- [Menyiapkan metode Amazon API Gateway sebagai penyedia identitas khusus](#)
- [Menyiapkan AS2 konfigurasi](#)

Memulai dengan AWS Transfer Family titik akhir server

Gunakan tutorial ini untuk memulai AWS Transfer Family (Transfer Family). Anda akan mempelajari cara membuat server SFTP berkemampuan dengan titik akhir yang dapat diakses publik menggunakan penyimpanan Amazon S3, menambahkan pengguna dengan otentikasi yang dikelola layanan, dan mentransfer file dengan Cyberduck.

Topik

- [Prasyarat](#)
- [Langkah 1: Masuk ke AWS Transfer Family konsol](#)
- [Langkah 2: Buat server SFTP yang diaktifkan](#)
- [Langkah 3: Tambahkan pengguna yang dikelola layanan](#)
- [Langkah 4: Transfer file menggunakan klien](#)

Prasyarat

Sebelum Anda mulai, pastikan untuk melengkapi persyaratan di[Prasyarat](#). Sebagai bagian dari penyiapan ini, Anda membuat bucket Amazon Simple Storage Service (Amazon S3) dan AWS Identity and Access Management peran pengguna IAM ().

Ada izin yang diperlukan untuk menggunakan AWS Transfer Family konsol, dan ada izin yang diperlukan untuk mengonfigurasi AWS layanan lain yang digunakan Transfer Family, seperti Amazon Simple Storage Service, Amazon Elastic File System AWS Certificate Manager, dan Amazon Route 53. Misalnya, untuk pengguna yang mentransfer file masuk dan keluar AWS menggunakan Transfer Family, AmazonS3 FullAccess memberikan izin untuk menyiapkan dan menggunakan bucket Amazon S3. Beberapa izin dalam kebijakan ini diperlukan untuk membuat bucket Amazon S3.

Untuk menggunakan konsol Transfer Family, Anda memerlukan yang berikut:

- AWSTransferConsoleFullAccess memberikan izin bagi SFTP pengguna Anda untuk membuat sumber daya Transfer Family.
- IAMFullAccess(atau khususnya kebijakan yang memungkinkan pembuatan IAM peran) hanya diperlukan jika Anda ingin Transfer Family secara otomatis membuat peran logging untuk server Anda di CloudWatch Log Amazon atau peran pengguna untuk pengguna yang masuk ke server.
- Untuk membuat dan menghapus jenis VPC server, Anda perlu menambahkan tindakan ec2: CreateVpcEndpoint dan ec2: DeleteVpcEndpoints ke kebijakan Anda.

Note

AmazonS3 FullAccess dan IAMFullAccesskebijakan, sendiri, tidak diperlukan untuk penggunaan umum. AWS Transfer Family Mereka disajikan di sini sebagai cara sederhana untuk memastikan bahwa semua izin yang Anda butuhkan tercakup. Selain itu, ini adalah

kebijakan AWS terkelola, yang merupakan kebijakan standar yang tersedia untuk semua AWS pelanggan. Anda dapat melihat izin individual dalam kebijakan ini dan menentukan set minimal yang Anda perlukan untuk tujuan Anda.

Langkah 1: Masuk ke AWS Transfer Family konsol

Untuk masuk ke Transfer Family

1. Masuk ke AWS Management Console dan buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Untuk ID Akun atau alias, masukkan ID untuk Anda Akun AWS.
3. Untuk nama IAM pengguna, masukkan nama peran pengguna yang Anda buat untuk Transfer Family.
4. Untuk Kata Sandi, masukkan kata sandi AWS akun Anda.
5. Pilih Masuk.

Langkah 2: Buat server SFTP yang diaktifkan

Secure Shell (SSH) File Transfer Protocol (SFTP) adalah protokol jaringan yang digunakan untuk transfer data yang aman melalui internet. Protokol ini mendukung fungsionalitas keamanan dan otentikasi penuh. SSH ini banyak digunakan untuk bertukar data, termasuk informasi sensitif antara mitra bisnis di berbagai industri seperti layanan keuangan, perawatan kesehatan, ritel, dan periklanan.

Untuk membuat server SFTP yang diaktifkan

1. Pilih Server dari panel Navigasi lalu pilih Buat server.
2. Di Pilih protokol, pilih SFTP, lalu pilih Berikutnya.
3. Di Pilih penyedia identitas, pilih Layanan yang dikelola untuk menyimpan identitas dan kunci pengguna di Transfer Family, lalu pilih Berikutnya.
4. Di Pilih titik akhir, lakukan hal berikut:
 - a. Untuk tipe Endpoint, pilih tipe titik akhir yang dapat diakses publik.
 - b. Untuk nama host Kustom, pilih Tidak Ada.
 - c. Pilih Berikutnya.

5. Di Pilih domain, pilih Amazon S3.
6. Di Konfigurasikan detail tambahan, lakukan hal berikut:
 - a. Untuk CloudWatch logging, pilih Buat peran baru agar Transfer Family membuat IAM peran secara otomatis, selama Anda memiliki izin yang tepat untuk membuat peran baru. IAMPeran yang diciptakan disebut `AWSTransferLoggingAccess`.
 - b. Untuk opsi algoritma kriptografi, pilih kebijakan keamanan yang berisi algoritma kriptografi yang diaktifkan untuk digunakan oleh server Anda. Kebijakan keamanan default adalah `TransferSecurityPolicy-2020-06`.
 - c. Pilih Berikutnya.
7. Di Tinjau dan buat, pilih Buat server. Anda dibawa ke halaman Server.

Diperlukan beberapa menit sebelum status server baru Anda berubah menjadi Online. Pada saat itu, server Anda dapat melakukan operasi file, tetapi Anda harus membuat pengguna terlebih dahulu.

Langkah 3: Tambahkan pengguna yang dikelola layanan

Untuk menambahkan pengguna ke server SFTP yang diaktifkan

1. Pada halaman Server, pilih kotak centang server yang ingin Anda tambahkan pengguna.
2. Pilih Tambahkan pengguna.
3. Di bagian Konfigurasi pengguna, untuk Nama Pengguna, masukkan nama pengguna. Nama pengguna ini harus minimal 3 dan maksimal 100 karakter. Anda dapat menggunakan karakter berikut dalam nama pengguna: a—z, A-Z, 0-9, garis bawah '_', tanda hubung '-', periode '.', dan pada tanda "@". Nama pengguna tidak dapat dimulai dengan tanda hubung, titik, atau tanda saat.
4. Untuk Access, pilih IAM peran yang sebelumnya Anda buat yang menyediakan akses ke bucket Amazon S3 Anda.

Anda membuat IAM peran ini menggunakan prosedur di [Buat IAM peran dan kebijakan](#).

IAMPeran tersebut mencakup IAM kebijakan yang menyediakan akses ke bucket Amazon S3 Anda. Ini juga mencakup hubungan kepercayaan dengan AWS Transfer Family layanan, yang didefinisikan dalam IAM kebijakan lain.

 Note

IAMPeran untuk pengguna yang dikelola layanan harus berisi izin untuk mengakses bucket yang diinginkan. Izin untuk mengakses bucket yang diinginkan tercakup dalam S3 FullAccess yang memberikan izin tingkat administrator ke sumber daya S3.

5. Untuk Kebijakan, pilih Tidak Ada.
6. Untuk direktori Home, pilih bucket Amazon S3 untuk menyimpan data yang akan ditransfer. AWS Transfer Family Masukkan jalur ke home direktori tempat pengguna Anda mendarat saat mereka masuk menggunakan klien mereka.

Jika Anda membiarkan parameter ini kosong, root direktori bucket Amazon S3 Anda digunakan. Dalam hal ini, pastikan bahwa IAM peran Anda menyediakan akses ke root direktori ini.

 Note

Kami menyarankan Anda memilih jalur direktori yang berisi nama pengguna pengguna, yang memungkinkan Anda menggunakan kebijakan sesi secara efektif. Kebijakan sesi membatasi akses pengguna di bucket Amazon S3 ke direktori pengguna tersebut. home

7. Untuk Dibatasi, pilih kotak centang agar pengguna Anda tidak dapat mengakses apa pun di luar folder itu dan tidak dapat melihat bucket atau nama folder Amazon S3.

 Note

Saat menugaskan pengguna direktori home dan membatasi pengguna ke direktori home itu, ini harus cukup untuk mengunci akses pengguna ke folder yang ditunjuk. Gunakan kebijakan sesi saat Anda perlu menerapkan kontrol lebih lanjut.

8. Untuk SSHpublic key, masukkan bagian SSH kunci publik dari SSH key pair.

Kunci Anda divalidasi oleh layanan sebelum Anda dapat menambahkan pengguna baru Anda.

 Important

Format kunci SSH publik adalah `ssh-rsa <string>`. Untuk petunjuk tentang cara membuat SSH key pair, lihat [Buat SSH kunci untuk pengguna yang dikelola layanan](#).

- (Opsional) Untuk Kunci dan Nilai, masukkan satu atau beberapa tag sebagai pasangan nilai kunci, dan pilih Tambahkan tag.
- Pilih Tambah untuk menambahkan pengguna baru Anda ke server yang Anda pilih.

Pengguna baru muncul di bagian Pengguna pada halaman detail Server.

Langkah 4: Transfer file menggunakan klien

Anda mentransfer file melalui AWS Transfer Family layanan dengan menentukan operasi transfer di klien. AWS Transfer Family mendukung beberapa klien. Untuk detailnya, lihat [Mentransfer file melalui titik akhir server menggunakan klien](#)

Bagian ini berisi prosedur untuk menggunakan Cyberduck dan OpenSSH.

Topik

- [Gunakan Cyberduck](#)
- [Gunakan Terbuka SSH](#)

Gunakan Cyberduck

Untuk mentransfer file AWS Transfer Family menggunakan Cyberduck

- Buka klien [Cyberduck](#).
- Pilih Buka Koneksi.
- Dalam kotak dialog Open Connection, pilih SFTP(SSHFile Transfer Protocol).
- Untuk Server, masukkan endpoint server Anda. Titik akhir server terletak di halaman detail Server, lihat [LihatSFTP,FTPS, dan detail FTP server](#).
- Untuk nomor Port, masukkan **22** untukSFTP.
- Untuk Nama Pengguna, masukkan nama untuk pengguna yang Anda buat[Mengelola pengguna untuk titik akhir server](#).

7. Untuk Kunci SSH Pribadi, pilih atau masukkan kunci SSH pribadi.
8. Pilih Hubungan.
9. Lakukan transfer file Anda.

Tergantung di mana file Anda berada, lakukan salah satu hal berikut:

- Di direktori lokal Anda (sumber), pilih file yang ingin Anda transfer, dan seret dan jatuhkan ke direktori Amazon S3 (target).
- Di direktori Amazon S3 (sumber), pilih file yang ingin Anda transfer, dan seret dan jatuhkan ke direktori lokal Anda (target).

Gunakan Terbuka SSH

Gunakan instruksi yang mengikuti untuk mentransfer file dari baris perintah menggunakan BukaSSH.

Note

Klien ini hanya berfungsi dengan SFTP server yang diaktifkan.

Untuk mentransfer file AWS Transfer Family menggunakan utilitas SSH baris perintah Open

1. Di Linux atau Macintosh, buka terminal perintah.
2. Pada prompt, masukkan perintah berikut: `% sftp -i transfer-key sftp_user@service_endpoint`

Dalam perintah sebelumnya, `sftp_user` adalah nama pengguna dan `transfer-key` merupakan kunci SSH pribadi. Di sini, `service_endpoint` adalah titik akhir server seperti yang ditunjukkan di AWS Transfer Family konsol untuk server yang dipilih.

`sftpPrompt` akan muncul.

3. (Opsional) Untuk melihat direktori home pengguna, masukkan perintah berikut pada `sftp` prompt: `sftp> pwd`
4. Pada baris berikutnya, masukkan teks berikut: `sftp> cd /mybucket/home/sftp_user`

Dalam latihan memulai ini, bucket Amazon S3 ini adalah target transfer file.

5. Pada baris berikutnya, masukkan perintah berikut: `sftp> put filename.txt`

putPerintah mentransfer file ke bucket Amazon S3.

Pesan seperti berikut ini muncul, menunjukkan bahwa transfer file sedang berlangsung, atau selesai.

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

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

Menyiapkan alur kerja terkelola untuk mendekripsi file

Tutorial ini menggambarkan cara mengatur alur kerja terkelola yang berisi langkah dekripsi. Tutorial ini juga menunjukkan cara mengunggah file terenkripsi ke bucket Amazon S3 dan kemudian melihat file yang didekripsi di bucket yang sama.

Note

Blog AWS penyimpanan memiliki posting yang menjelaskan cara mengenkripsi dan mendekripsi file, mengenkripsi dan mendekripsi file dengan [dan](#). PGP AWS Transfer Family

Topik

- [Langkah 1: Konfigurasi peran eksekusi](#)
- [Langkah 2: Buat alur kerja terkelola](#)
- [Langkah 3: Tambahkan alur kerja ke server dan buat pengguna](#)
- [Langkah 4: Buat PGP key pair](#)
- [Langkah 5: Simpan kunci PGP pribadi di AWS Secrets Manager](#)
- [Langkah 6: Enkripsi file](#)
- [Langkah 7: Jalankan alur kerja dan lihat hasilnya](#)

Langkah 1: Konfigurasi peran eksekusi

Buat peran eksekusi AWS Identity and Access Management (IAM) yang dapat digunakan Transfer Family untuk meluncurkan alur kerja. Proses pembuatan peran eksekusi dijelaskan dalam [IAM kebijakan untuk alur kerja](#).

Note

Sebagai bagian dari menciptakan peran eksekusi, pastikan untuk membangun hubungan kepercayaan antara peran eksekusi dan Transfer Family, seperti yang dijelaskan dalam [Untuk membangun hubungan kepercayaan](#).

Kebijakan peran eksekusi berikut berisi semua izin yang diperlukan untuk berhasil menjalankan alur kerja yang akan Anda buat dalam tutorial ini. Untuk menggunakan kebijakan contoh ini, ganti *user input placeholders* dengan informasi Anda sendiri. Ganti *DOC-EXAMPLE-BUCKET* dengan nama bucket Amazon S3 tempat Anda akan mengunggah file terenkripsi Anda.

Note

Tidak setiap alur kerja memerlukan setiap izin yang tercantum dalam contoh ini. Anda dapat membatasi izin berdasarkan jenis langkah dalam alur kerja spesifik Anda. Izin yang diperlukan untuk setiap jenis langkah yang telah ditentukan dijelaskan dalam [Gunakan langkah-langkah yang telah ditentukan](#). Izin yang diperlukan untuk langkah kustom dijelaskan di [AMizin untuk langkah khusus](#).

```
{
  "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/
* "
    }
  ]
}

```

Langkah 2: Buat alur kerja terkelola

Sekarang Anda perlu membuat alur kerja yang berisi langkah dekripsi.

Untuk membuat alur kerja yang berisi langkah dekripsi

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Di panel navigasi kiri, pilih Alur kerja, lalu pilih Buat alur kerja.
3. Masukkan detail berikut:
 - Masukkan deskripsi, misalnya **Decrypt workflow example**.
 - Di bagian Langkah nominal, pilih Tambah langkah.
4. Untuk Pilih jenis langkah, pilih Dekripsi file, lalu pilih Berikutnya.
5. Dalam kotak dialog Konfigurasi parameter, tentukan yang berikut ini:
 - Masukkan nama langkah deskriptif, misalnya, **decrypt-step**. Spasi tidak diperbolehkan dalam nama langkah.
 - Untuk Tujuan untuk file yang didekripsi, pilih Amazon S3.
 - Untuk nama bucket Tujuan, pilih bucket Amazon S3 yang sama dengan yang Anda tentukan seperti *DOC-EXAMPLE-BUCKET* dalam IAM kebijakan yang Anda buat di Langkah 1.

- Untuk awalan kunci Destination, masukkan nama awalan (folder) tempat Anda ingin menyimpan file yang didekripsi di bucket tujuan, misalnya, **decrypted-files/**

 Note

Pastikan untuk menambahkan trailing slash (/) ke awalan Anda.

- Untuk tutorial ini, biarkan Overwrite sudah dihapus. Saat pengaturan ini dihapus, jika Anda mencoba mendekripsi file dengan nama identik dari file yang ada, pemrosesan alur kerja berhenti, dan file baru tidak diproses.

Pilih Berikutnya untuk pindah ke layar ulasan.

6. Tinjau detail untuk langkahnya. Jika semuanya benar, pilih Buat langkah.
7. Alur kerja Anda hanya membutuhkan satu langkah dekripsi, jadi tidak ada langkah tambahan untuk mengonfigurasi. Pilih Buat alur kerja untuk membuat alur kerja baru.

Perhatikan ID alur kerja untuk alur kerja baru Anda. Anda akan memerlukan ID ini untuk langkah selanjutnya. Tutorial ini menggunakan *w-1234abcd5678efghi* sebagai contoh ID alur kerja.

Langkah 3: Tambahkan alur kerja ke server dan buat pengguna

Sekarang Anda memiliki alur kerja dengan langkah dekripsi, Anda harus mengaitkannya dengan server Transfer Family. Tutorial ini menunjukkan cara melampirkan alur kerja ke server Transfer Family yang ada. Atau, Anda dapat membuat server baru untuk digunakan dengan alur kerja Anda.

Setelah Anda melampirkan alur kerja ke server, Anda harus membuat pengguna yang dapat SFTP masuk ke server dan memicu alur kerja untuk dijalankan.

Untuk mengonfigurasi server Transfer Family untuk menjalankan alur kerja

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Di panel navigasi kiri, pilih Server, lalu pilih server dari daftar. Pastikan server ini mendukung SFTP protokol.
3. Pada halaman detail untuk server, gulir ke bawah ke bagian Detail tambahan, lalu pilih Edit.
4. Pada halaman Edit detail tambahan, di bagian Alur kerja terkelola, pilih alur kerja Anda, dan pilih peran eksekusi yang sesuai.

- Untuk Alur Kerja untuk upload file lengkap, pilih alur kerja yang Anda buat [Langkah 2: Buat alur kerja terkelola](#), misalnya, **w-1234abcd5678efghi**
 - Untuk peran eksekusi alur kerja terkelola, pilih IAM peran yang Anda buat. [Langkah 1: Konfigurasi peran eksekusi](#)
5. Gulir ke bagian bawah halaman, dan pilih Simpan untuk menyimpan perubahan Anda.

Catat ID untuk server yang Anda gunakan. Nama AWS Secrets Manager rahasia yang Anda gunakan untuk menyimpan PGP kunci Anda sebagian didasarkan pada ID server.

Untuk menambahkan pengguna yang dapat memicu alur kerja

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Di panel navigasi kiri, pilih Server, lalu pilih server yang Anda gunakan untuk mendekripsi alur kerja.
3. Pada halaman detail server, gulir ke bawah ke bagian Pengguna, dan pilih Tambah pengguna.
4. Untuk pengguna baru Anda, masukkan detail berikut:
 - Untuk Nama Pengguna, masukkan **decrypt-user**.
 - Untuk Peran, pilih peran pengguna yang dapat mengakses server Anda.
 - Untuk direktori Home, pilih bucket Amazon S3 yang Anda gunakan sebelumnya, misalnya, **DOC-EXAMPLE-BUCKET**
 - Untuk kunci SSH publik, tempel kunci publik yang sesuai dengan kunci pribadi yang Anda miliki. Untuk detailnya, lihat [Buat SSH kunci untuk pengguna yang dikelola layanan](#).
5. Pilih Tambah untuk menyimpan pengguna baru Anda.

Catat nama pengguna Transfer Family Anda untuk server ini. Rahasiannya sebagian didasarkan pada nama pengguna. Untuk kesederhanaan, tutorial ini menggunakan rahasia default yang dapat digunakan oleh setiap pengguna server.

Langkah 4: Buat PGP key pair

Gunakan salah satu [PGPklien yang didukung](#) untuk menghasilkan PGP key pair. Proses ini dijelaskan secara rinci dalam [Hasilkan PGP kunci](#).

Untuk menghasilkan PGP key pair

1. Untuk tutorial ini, Anda dapat menggunakan gpg (GnuPG) versi 2.0.22 klien untuk menghasilkan PGP key pair yang digunakan RSA sebagai algoritma enkripsi. Untuk klien ini, jalankan perintah berikut, dan berikan alamat email dan frasa sandi. Anda dapat menggunakan nama atau alamat email apa pun yang Anda suka. Pastikan Anda mengingat nilai yang Anda gunakan, karena Anda harus memasukkannya nanti dalam tutorial.

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

Note

Jika Anda menggunakan GnuPG versi 2.3.0 atau yang lebih baru, Anda harus menjalankannya. `gpg --full-gen-key` Ketika diminta untuk jenis kunci yang akan dibuat, pilih RSA atau ECC. Namun, jika Anda memilih ECC, pastikan untuk memilih salah satu NIST atau BrainPool untuk kurva elips. Jangan memilih Curve 25519.

2. Ekspor kunci pribadi dengan menjalankan perintah berikut. Ganti `user@example.com` dengan alamat email yang Anda gunakan saat membuat kunci.

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

Perintah ini mengekspor kunci pribadi ke **workflow-tutorial-key.pgp** file. Anda dapat memberi nama file output apa pun yang Anda suka. Anda juga dapat menghapus file kunci pribadi setelah Anda menambahkannya AWS Secrets Manager.

Langkah 5: Simpan kunci PGP pribadi di AWS Secrets Manager

Anda perlu menyimpan kunci pribadi di Secrets Manager, dengan cara yang sangat spesifik, sehingga alur kerja dapat menemukan kunci pribadi saat alur kerja menjalankan langkah dekripsi pada file yang diunggah.

Note

Ketika Anda menyimpan rahasia di Secrets Manager, Anda Akun AWS dikenakan biaya. Untuk informasi tentang harga, lihat [AWS Secrets Manager Harga](#).

Untuk menyimpan kunci PGP pribadi di Secrets Manager

1. Masuk ke AWS Management Console dan buka AWS Secrets Manager konsol di <https://console.aws.amazon.com/secretsmanager/>.
 2. Pada panel navigasi kiri, pilih Rahasia.
 3. Pada halaman Rahasia, pilih Simpan rahasia baru.
 4. Pada halaman Pilih jenis rahasia, untuk tipe Rahasia, pilih Jenis rahasia lainnya.
 5. Di bagian pasangan kunci/Nilai, pilih tab kunci/Nilai.
 - Kunci — Masukkan **PGPPrivateKey**.
 - nilai — Tempelkan teks kunci pribadi Anda ke bidang nilai.
 6. Pilih Tambah baris, dan di bagian pasangan kunci/Nilai, pilih tab kunci/Nilai.
 - Kunci — Masukkan **PGPPassphrase**.
 - value — Masukkan kata sandi yang Anda gunakan saat membuat PGP key pair. [Langkah 4: Buat PGP key pair](#)
 7. Pilih Berikutnya.
 8. Pada halaman Konfigurasi rahasia, masukkan nama dan deskripsi untuk rahasia Anda. Untuk tutorial ini, Anda dapat membuat rahasia default yang dapat digunakan semua pengguna. Dengan asumsi bahwa ID server adalah **s-11112222333344445**, sebutkan rahasianya **aws/transfer/s-11112222333344445/pggp-default**. Ganti **s-11112222333344445** dengan ID server Transfer Family Anda. Masukkan deskripsi untuk rahasia Anda.
- 
- ### Note
- Untuk membuat rahasia hanya untuk pengguna yang Anda buat sebelumnya, beri nama rahasianya **aws/transfer/s-11112222333344445/decrypt-user**.
9. Pilih Berikutnya, dan kemudian terima default pada halaman Konfigurasi rotasi. Lalu pilih Berikutnya.
 10. Pada halaman Review, pilih Store untuk membuat dan menyimpan rahasia.

Untuk informasi selengkapnya tentang menambahkan kunci PGP pribadi ke Secrets Manager, lihat [Menggunakan AWS Secrets Manager untuk menyimpan PGP kunci Anda](#).

Langkah 6: Enkripsi file

Gunakan gpg program untuk mengenkripsi file untuk digunakan dalam alur kerja Anda. Jalankan perintah berikut untuk mengenkripsi file:

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

Sebelum menjalankan perintah ini, perhatikan hal berikut:

- Untuk `-r` argumennya, ganti *marymajor@example.com* dengan alamat email yang Anda gunakan saat membuat PGP key pair.
- `--openpgp` Bendera adalah opsional. Bendera ini membuat file terenkripsi sesuai dengan standar [Open PGP RFC4880](#).
- Perintah ini membuat file bernama **testfile.txt.gpg** di lokasi yang sama dengan **testfile.txt**.

Langkah 7: Jalankan alur kerja dan lihat hasilnya

Untuk menjalankan alur kerja, Anda terhubung ke server Transfer Family dengan pengguna yang Anda buat di Langkah 3. Kemudian Anda dapat melihat di bucket Amazon S3 yang Anda tentukan di [Langkah 2.5, konfigurasi parameter tujuan](#) untuk melihat file yang didekripsi.

Untuk menjalankan alur kerja dekripsi

1. Buka terminal perintah.
2. Jalankan perintah berikut, ganti *your-endpoint* dengan endpoint Anda yang sebenarnya, dan *transfer-key* dengan kunci SSH pribadi pengguna Anda:

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

Misalnya, jika kunci pribadi disimpan `~/ .ssh/decrypt-user`, dan titik akhir Anda, perintahnya adalah sebagai berikut: `s-11112222333344445.server.transfer.us-east-2.amazonaws.com`

```
sftp -i ~/ .ssh/decrypt-user decrypt-user@s-11112222333344445.server.transfer.us-east-2.amazonaws.com
```

3. Jalankan perintah `pwd`. Jika berhasil, perintah ini akan mengembalikan yang berikut:

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

Direktori Anda mencerminkan nama bucket Amazon S3 Anda.

4. Jalankan perintah berikut untuk mengunggah file dan memicu alur kerja untuk dijalankan:

```
put testfile.txt.gpg
```

5. Untuk tujuan file yang didekripsi, Anda menentukan `decrypted-files/` folder saat Anda membuat alur kerja. Sekarang, Anda dapat menavigasi ke folder itu dan daftar isinya.

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

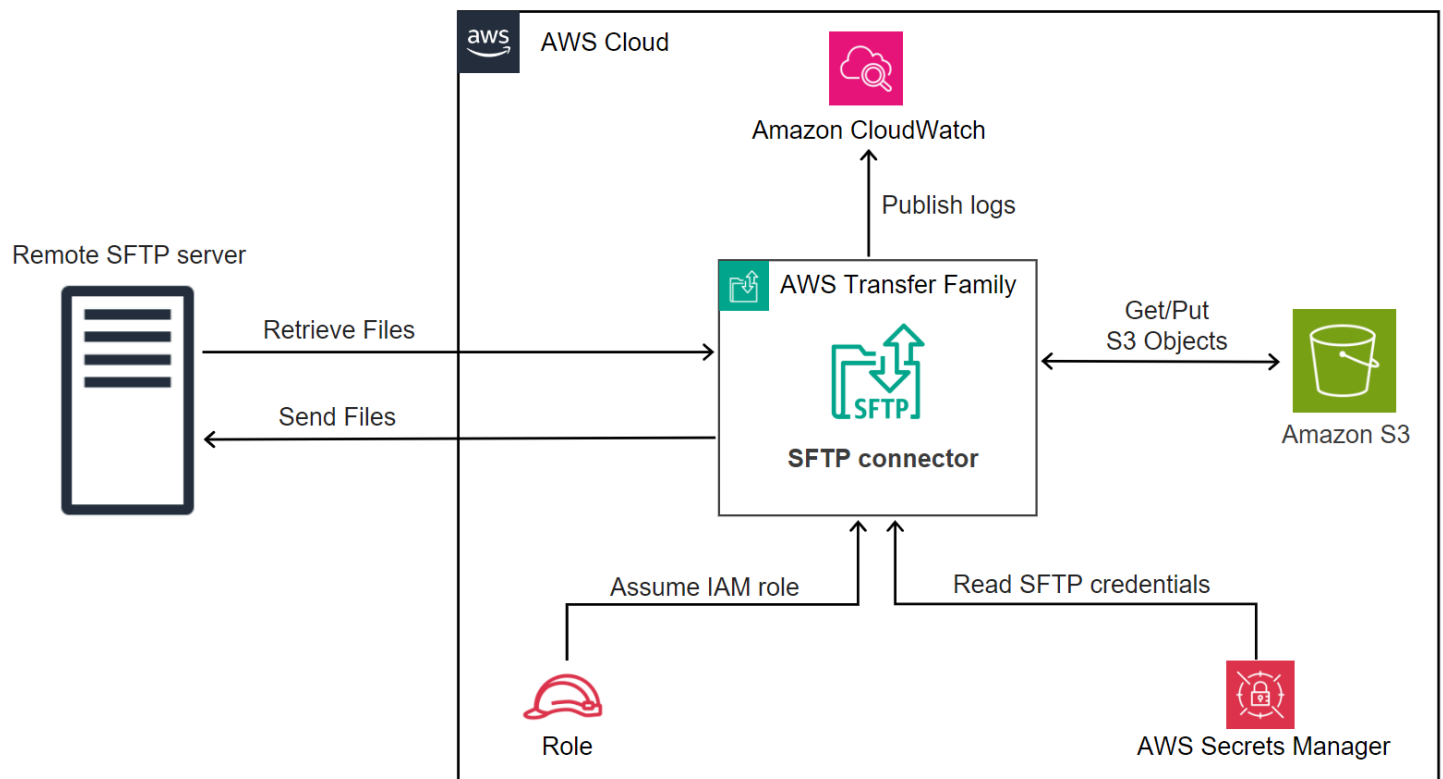
Jika berhasil, `ls` perintah mencantumkan `testfile.txt` file. Anda dapat mengunduh file ini dan memverifikasi bahwa itu sama dengan file asli yang Anda enkripsi sebelumnya.

Menyiapkan dan menggunakan SFTP konektor

Tujuan dari konektor adalah untuk membangun hubungan antara AWS penyimpanan Anda dan SFTP server mitra. Anda dapat mengirim file dari Amazon S3 ke tujuan eksternal milik mitra. Anda juga dapat menggunakan SFTP konektor untuk mengambil file dari SFTP server mitra.

Tutorial ini menggambarkan cara mengatur SFTP konektor, dan kemudian mentransfer file antara penyimpanan Amazon S3 dan SFTP server.

SFTPKonektor mengambil SFTP kredensi dari AWS Secrets Manager untuk mengautentikasi ke SFTP server jarak jauh dan membuat koneksi. Konektor mengirim file ke atau mengambil file dari server jarak jauh, dan menyimpan file di Amazon S3. IAM Peran digunakan untuk mengizinkan akses ke bucket Amazon S3 dan ke kredensial yang disimpan di Secrets Manager. Dan Anda dapat masuk ke Amazon CloudWatch.



Topik

- [Langkah 1: Buat sumber daya pendukung yang diperlukan](#)
 - [Langkah 2: Buat dan uji SFTP konektor](#)
 - [Langkah 3: Kirim dan ambil file menggunakan konektor SFTP](#)
 - [Prosedur untuk membuat server Transfer Family untuk digunakan sebagai SFTP server jarak jauh](#)
- [Anda](#)

Langkah 1: Buat sumber daya pendukung yang diperlukan

Anda dapat menggunakan SFTP konektor untuk menyalin file antara Amazon S3 dan server jarak jauh SFTP apa pun. Untuk tutorial ini, kami menggunakan AWS Transfer Family server sebagai SFTP server jarak jauh kami. Kita perlu membuat dan mengkonfigurasi sumber daya berikut:

- Buat bucket Amazon S3 untuk menyimpan file di AWS lingkungan Anda, dan untuk mengirim dan mengambil file dari server jarak jauh.: SFTP [Buat ember Amazon S3](#)
- Buat AWS Identity and Access Management peran untuk mengakses penyimpanan Amazon S3 dan rahasia kami di Secrets Manager.: [Buat IAM peran dengan izin yang diperlukan](#)

- Buat server Transfer Family yang menggunakan SFTP protokol, dan pengguna yang dikelola layanan yang menggunakan SFTP konektor untuk mentransfer file ke atau dari SFTP server: [Membuat SFTP server Transfer Family dan pengguna](#)
- Buat AWS Secrets Manager rahasia yang menyimpan kredensial yang digunakan oleh SFTP konektor untuk masuk ke SFTP server jarak jauh: [Buat dan simpan rahasia di AWS Secrets Manager](#)

Buat ember Amazon S3

Untuk membuat bucket Amazon S3

1. Masuk ke AWS Transfer Family konsol di <https://console.aws.amazon.com/s3/>.
2. Pilih Wilayah dan masukkan nama.

Untuk tutorial ini, ember kami ada di **US East (N. Virginia) us-east-1**, dan namanya **sftp-server-storage-east**.

3. Terima defaultnya dan pilih Buat bucket.

Untuk detail selengkapnya tentang membuat bucket Amazon S3, lihat [Bagaimana cara membuat bucket S3?](#) di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

Buat IAM peran dengan izin yang diperlukan

Untuk peran akses, buat kebijakan dengan izin berikut.

Contoh berikut memberikan izin yang diperlukan untuk mengakses **DOC-EXAMPLE-BUCKET** di Amazon S3, dan rahasia yang ditentukan disimpan di Secrets Manager.

```
{
  "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"
}
]
}

```

Ganti item sebagai berikut:

- Untuk *DOC-EXAMPLE-BUCKET*, tutorial menggunakan **s3-storage-east**.
- Untuk *region*, tutorial menggunakan **us-east-1**.
- Untuk *account-id*, gunakan Akun AWS ID Anda.
- Untuk *SecretName-6RandomCharacters*, kami **using sftp-connector1** untuk nama (Anda akan memiliki enam karakter acak Anda sendiri untuk rahasia Anda).

Anda juga harus memastikan bahwa peran ini berisi hubungan kepercayaan yang memungkinkan konektor mengakses sumber daya Anda saat melayani permintaan transfer pengguna Anda. Untuk detail tentang membangun hubungan kepercayaan, lihat [Untuk membangun hubungan kepercayaan](#).

Note

Untuk melihat detail untuk peran yang kita gunakan untuk tutorial, lihat [Gabungan peran pengguna dan akses](#).

Buat dan simpan rahasia di AWS Secrets Manager

Kita perlu menyimpan rahasia di Secrets Manager untuk menyimpan kredensial pengguna untuk konektor AndaSFTP. Anda dapat menggunakan kata sandi, kunci SSH pribadi, atau keduanya. Untuk tutorial, kita menggunakan kunci pribadi.

Note

Ketika Anda menyimpan rahasia di Secrets Manager, Anda Akun AWS dikenakan biaya. Untuk informasi tentang harga, lihat [AWS Secrets Manager Harga](#).

Sebelum Anda memulai prosedur untuk menyimpan rahasia, mengambil dan memformat kunci pribadi Anda. Kunci pribadi harus sesuai dengan kunci publik yang dikonfigurasi untuk pengguna di SFTP server jarak jauh. Untuk tutorial kami, kunci pribadi harus sesuai dengan kunci publik yang disimpan untuk pengguna uji kami di SFTP server Transfer Family yang kami gunakan sebagai server jarak jauh.

Untuk melakukan ini, jalankan perintah berikut:

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

Misalnya, jika file kunci pribadi Anda berada di `~/ .ssh/sftp-testuser-privatekey`, perintahnya adalah sebagai berikut.

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

Ini menghasilkan kunci dalam format yang benar (dengan karakter baris baru yang disematkan) ke output standar. Salin teks ini di suatu tempat, karena Anda harus menempelkannya dalam prosedur berikut (pada langkah 6).

Untuk menyimpan kredensi pengguna di Secrets Manager untuk konektor SFTP

1. Masuk ke AWS Management Console dan buka AWS Secrets Manager konsol di <https://console.aws.amazon.com/secretsmanager/>.
2. Pada panel navigasi kiri, pilih Rahasia.
3. Pada halaman Rahasia, pilih Simpan rahasia baru.
4. Pada halaman Pilih jenis rahasia, untuk tipe Rahasia, pilih Jenis rahasia lainnya.
5. Di bagian pasangan kunci/Nilai, pilih tab kunci/Nilai.
 - Kunci — Masukkan **Username**.
 - nilai — Masukkan nama pengguna kami, **sftp-testuser**.
6. Untuk memasukkan kunci, kami sarankan Anda menggunakan tab Plaintext.
 - a. Pilih Tambah baris, lalu masukkan **PrivateKey**.
 - b. Pilih tab Plaintext. Bidang sekarang berisi teks berikut:

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

- c. Tempelkan teks untuk kunci pribadi Anda (disimpan sebelumnya) di antara tanda kutip ganda kosong ("").

Layar Anda akan terlihat sebagai berikut (data kunci berwarna abu-abu).



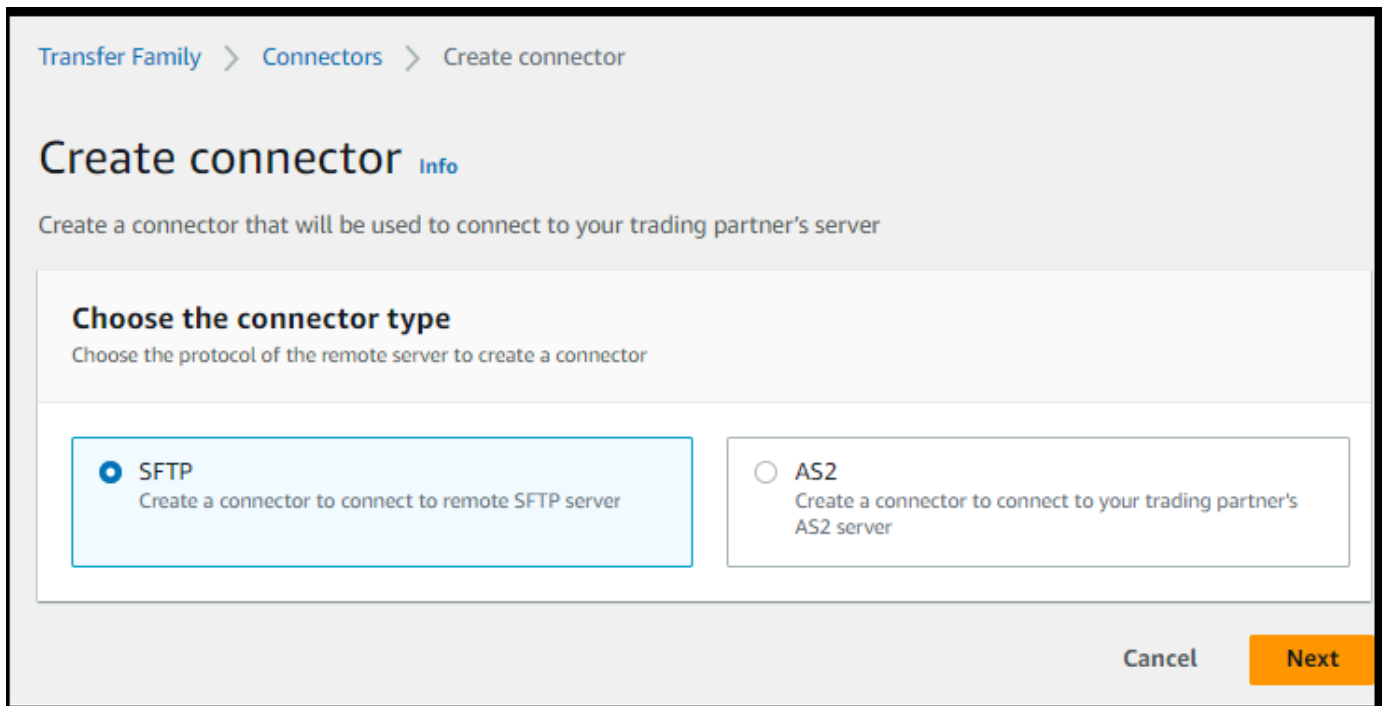
7. Pilih Berikutnya.
8. Pada halaman Konfigurasi rahasia, masukkan nama untuk rahasia Anda. Untuk tutorial ini, kami beri nama rahasianya **aws/transfer/sftp-connector1**.
9. Pilih Berikutnya, dan kemudian terima default pada halaman Konfigurasi rotasi. Lalu pilih Berikutnya.
10. Pada halaman Review, pilih Store untuk membuat dan menyimpan rahasia.

Langkah 2: Buat dan uji SFTP konektor

Di bagian ini, kami membuat SFTP konektor yang menggunakan semua sumber daya yang kami buat sebelumnya. Untuk detail selengkapnya, lihat [Konfigurasi SFTP konektor](#).

Untuk membuat SFTP konektor

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Di panel navigasi kiri, pilih Konektor, lalu pilih Buat konektor.
3. Pilih SFTP jenis konektor untuk membuat SFTP konektor, lalu pilih Berikutnya.



4. Di bagian Konfigurasi konektor, berikan informasi berikut:
 - Untuk URL, masukkan URL SFTP server jarak jauh. Untuk tutorialnya, kita masuk ke URL server Transfer Family yang kita gunakan sebagai SFTP server jarak jauh.

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

Ganti **1111aaaa2222bbbb3** dengan ID server Transfer Family Anda.

- Untuk peran Access, masukkan peran yang kita buat sebelumnya, **sftp-connector-role**.
- Untuk peran Logging, pilih **AWSTransferLoggingAccess**.

 Note

AWSTransferLoggingAccess adalah kebijakan yang AWS dikelola. Kebijakan ini dijelaskan secara rinci dalam [AWS kebijakan terkelola: AWSTransferLoggingAccess](#).

Connector configuration

URL

Specify the URL of remote server

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

Access role

IAM Role for Amazon S3 access and AWS Secrets Manager access

sftp-connector-role

Logging role - optional [Info](#)

IAM role for the connector to push events to your CloudWatch logs

AWSTransferLoggingAccess

5. Di bagian SFTP Konfigurasi, berikan informasi berikut:

- Untuk kredensial Connector, pilih nama sumber daya Secrets Manager yang berisi SFTP kredensial. Untuk tutorialnya, pilih **aws/transfer/sftp-connector1**.
- Untuk kunci host Tepercaya, tempel di bagian publik dari kunci host. Anda dapat mengambil kunci ini dengan menjalankan `ssh-keyscan` SFTP server Anda. Untuk detail tentang cara memformat dan menyimpan kunci host tepercaya, lihat [SftpConnectorConfig](#) dokumentasi tipe data.

SFTP configuration [Info](#)

Connector credentials
Select the username and password / SSH private key that will be used to connect to the remote server from AWS Secret Manager

aws/transfer/sftp-connector1 ↕ ↻ Store a new secret [↗](#)

Trusted host keys
Connector connects to the remote server only if the SSH public key matches one of the below

ssh-rsa AAA... Remove

Add trusted host key

- Setelah Anda mengkonfirmasi semua pengaturan Anda, pilih **Buat konektor** untuk membuat SFTP konektor.

Setelah Anda membuat SFTP konektor, kami sarankan Anda mengujinya sebelum mencoba mentransfer file apa pun menggunakan konektor baru Anda.

Test a connector using the console

Untuk menguji SFTP konektor

- Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
- Di panel navigasi kiri, pilih **Konektor**, dan pilih konektor.
- Dari menu **Tindakan**, pilih **Uji koneksi**.

AWS Transfer Family ×

Servers
Connectors
AS2 Trading Partners
Workflows
Feature Spotlight 4
What's New [↗](#)
Documentation [↗](#)

Introducing SFTP connectors
Use SFTP connector to connect to a remote SFTP server and transfer files to or from Amazon S3
[About connectors](#) | [Documentation](#) [↗](#) | [Pricing](#) [↗](#)

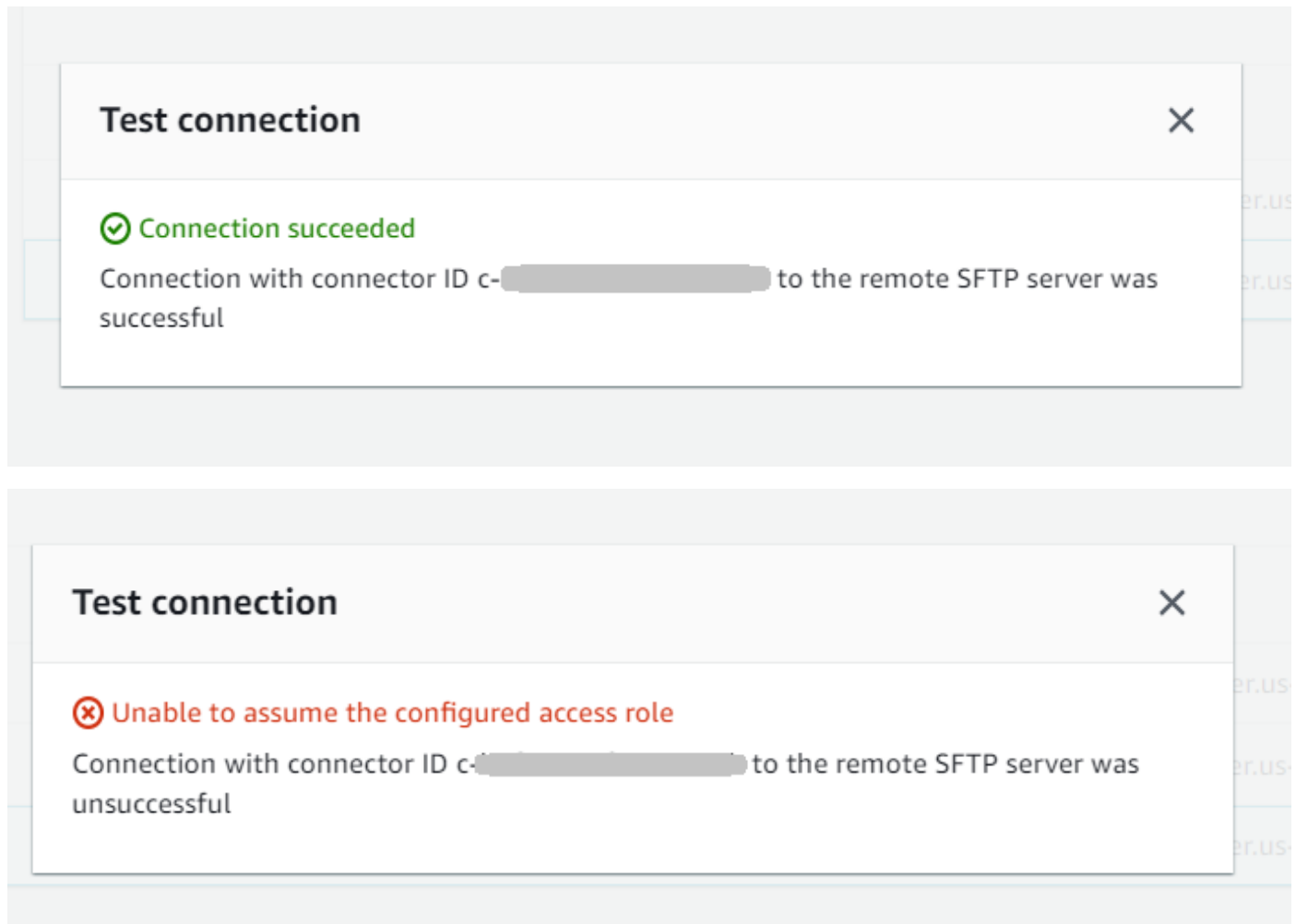
Connectors (2) [Info](#) ↻ Actions Create connector

Delete < 1 >

Test connection

	Connector ID	Type	URL
☐	c-...	AS2	http://s-...server.transfer.us-east-2.amazonaws.com:5080
☒	c-...	SFTP	sftp://s-...server.transfer.us-east-2.amazonaws.com

Sistem mengembalikan pesan, menunjukkan apakah tes lulus atau gagal. Jika tes gagal, sistem memberikan pesan kesalahan berdasarkan alasan pengujian gagal.



Test a connector using the CLI

Untuk menguji konektor menggunakan AWS Command Line Interface, jalankan perintah berikut pada prompt perintah (ganti *connector-id* dengan ID konektor Anda yang sebenarnya):

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

Jika tes berhasil, baris berikut dikembalikan:

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

Jika tes tidak berhasil, Anda menerima pesan kesalahan deskriptif, misalnya:

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

Langkah 3: Kirim dan ambil file menggunakan konektor SFTP

Untuk mempermudah, kami berasumsi bahwa Anda sudah memiliki file di bucket Amazon S3 Anda.

Note

Tutorial ini menggunakan bucket Amazon S3 untuk lokasi penyimpanan sumber dan tujuan. Jika SFTP server Anda tidak menggunakan penyimpanan Amazon S3, maka di mana pun Anda melihat `sftp-server-storage-east` dalam perintah berikut, Anda dapat mengganti jalur dengan jalur ke lokasi file yang dapat diakses dari server Anda SFTP.

- Kami mengirim file bernama `SEND-to-SERVER.txt` dari penyimpanan Amazon S3 ke server SFTP
- Kami mengambil file bernama `RETRIEVE-to-S3.txt` dari SFTP server ke penyimpanan Amazon S3.

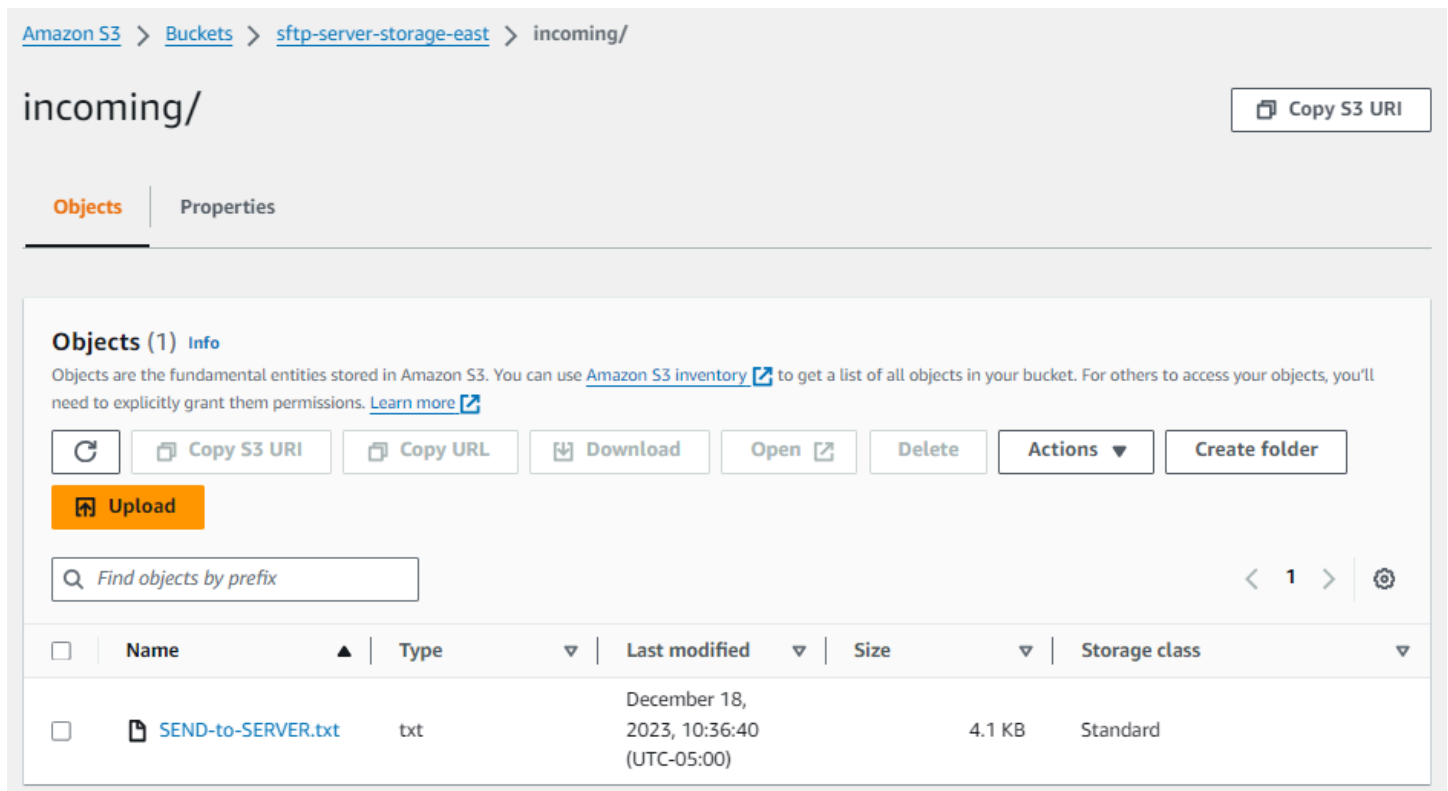
Note

Dalam perintah berikut, ganti *connector-id* dengan ID konektor Anda.

Pertama, kami mengirim file dari bucket Amazon S3 kami ke server jarak jauh SFTP. Dari command prompt, jalankan perintah berikut:

```
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"
```

`sftp-server-storage-east` Ember Anda sekarang akan terlihat seperti ini.



Jika Anda tidak melihat file seperti yang diharapkan, periksa CloudWatch log Anda.

Untuk memeriksa CloudWatch log Anda

1. Buka CloudWatch konsol Amazon di <https://console.aws.amazon.com/cloudwatch/>
2. Pilih Grup log dari menu navigasi kiri.
3. Masukkan ID konektor Anda di bilah pencarian untuk menemukan log Anda.
4. Pilih aliran Log yang dikembalikan dari pencarian.
5. Perluas entri log terbaru.

Jika berhasil, entri log terlihat seperti berikut:

```
{
  "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"
}

```

Jika transfer file gagal, entri log berisi pesan kesalahan yang menentukan masalah. Penyebab umum kesalahan adalah masalah dengan IAM izin dan jalur file yang salah.

Selanjutnya, kami mengambil file dari SFTP server ke bucket Amazon S3. Dari command prompt, jalankan perintah berikut:

```

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"

```

Jika transfer berhasil, bucket Amazon S3 Anda berisi file yang ditransfer, seperti yang ditunjukkan di sini.

Amazon S3 > Buckets > s3-storage-east > incoming/

incoming/ Copy S3 URI

Objects | Properties

Objects (1) [Info](#)

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

Refresh
Copy S3 URI
Copy URL
Download
Open
Delete
Actions
Create folder

Upload

Find objects by prefix

☐	Name	Type	Last modified	Size	Storage class
☐	RETRIEVE-to-S3.txt	txt	December 18, 2023, 10:26:58 (UTC-05:00)	4.1 KB	Standard

Jika berhasil, entri log terlihat seperti berikut:

```
{
  "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"
}
```

Prosedur untuk membuat server Transfer Family untuk digunakan sebagai SFTP server jarak jauh Anda

Berikut ini, kami menguraikan langkah-langkah untuk membuat server Transfer Family yang berfungsi sebagai SFTP server jarak jauh Anda untuk tutorial ini. Perhatikan hal berikut:

- Kami menggunakan server Transfer Family untuk mewakili SFTP server jarak jauh. Pengguna SFTP konektor tipikal memiliki SFTP server jarak jauh mereka sendiri. Lihat [Membuat SFTP server Transfer Family dan pengguna](#).
- Karena kami menggunakan server Transfer Family, kami juga menggunakan pengguna yang dikelola layanan SFTP. Dan, untuk kesederhanaan, kami menggabungkan izin yang dibutuhkan pengguna ini untuk mengakses server Transfer Family dengan izin yang mereka butuhkan untuk menggunakan konektor kami. Sekali lagi, sebagian besar kasus penggunaan SFTP konektor memiliki SFTP pengguna terpisah yang tidak terkait dengan server Transfer Family. Lihat [Membuat SFTP server Transfer Family dan pengguna](#).
- Untuk tutorial, karena kita menggunakan penyimpanan Amazon S3 untuk SFTP server jarak jauh kita, kita perlu membuat ember kedua, **s3-storage-east**, sehingga kita dapat mentransfer file dari satu ember ke ember lainnya.

Membuat SFTP server Transfer Family dan pengguna

Sebagian besar pengguna tidak perlu membuat SFTP server Transfer Family dan pengguna, karena Anda sudah memiliki SFTP server dengan pengguna, dan Anda dapat menggunakan server ini untuk mentransfer file ke dan dari. Namun, untuk tutorial ini, untuk kesederhanaan, kami menggunakan server Transfer Family untuk berfungsi sebagai SFTP server jarak jauh.

Ikuti prosedur yang dijelaskan dalam [Buat server SFTP yang diaktifkan](#) untuk membuat server, dan [Langkah 3: Tambahkan pengguna yang dikelola layanan](#) untuk menambahkan pengguna. Ini adalah detail pengguna yang kami gunakan untuk tutorial:

- Buat pengguna yang dikelola layanan Anda, `sftp-testuser`
 - Mengatur direktori home ke `/sftp-server-storage-east/sftp-testuser`
 - Saat Anda membuat pengguna, Anda menyimpan kunci publik. Kemudian, ketika Anda membuat rahasia di Secrets Manager, Anda perlu memberikan kunci pribadi yang sesuai.
- Peran: `sftp-connector-role`. Untuk tutorial, kami menggunakan IAM peran yang sama untuk SFTP pengguna kami dan untuk mengakses SFTP konektor. Saat membuat konektor untuk organisasi, Anda mungkin memiliki peran pengguna dan akses yang terpisah.
- Kunci host server: Anda perlu menggunakan kunci host server saat membuat konektor. Anda dapat mengambil kunci ini dengan menjalankan `ssh-keyscan` server Anda. Misalnya, jika ID server Anda `s-1111aaaa2222bbbb3`, dan titik akhirnya `us-east-1`, perintah berikut mengambil kunci host server:

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

Salin teks ini di suatu tempat, karena Anda harus menempelkannya dalam [Langkah 2: Buat dan uji SFTP konektor](#) prosedur.

Gabungan peran pengguna dan akses

Untuk tutorial, kita menggunakan peran tunggal gabungan. Kami menggunakan peran ini baik untuk SFTP pengguna kami, maupun untuk akses ke konektor. Contoh berikut berisi rincian untuk peran ini, jika Anda ingin melakukan tugas-tugas dalam tutorial.

Contoh berikut memberikan izin yang diperlukan untuk mengakses dua bucket kami di Amazon S3, dan rahasia bernama disimpan di `aws/transfer/sftp-connector1` Secrets Manager. Untuk tutorial, peran ini diberi nama `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"
    }
  ]
}

```

Untuk detail lengkap tentang membuat peran untuk Transfer Family, ikuti prosedur yang dijelaskan [Membuat peran pengguna](#) untuk membuat peran.

Menyiapkan metode Amazon API Gateway sebagai penyedia identitas khusus

Tutorial ini menggambarkan cara mengatur metode Amazon API Gateway dan menggunakannya sebagai penyedia identitas khusus untuk mengunggah file ke AWS Transfer Family server. Tutorial ini menggunakan [template stack Dasar](#), dan fungsi dasar lainnya sebagai contoh saja.

Topik

- [Prasyarat](#)
- [Langkah 1: Buat CloudFormation tumpukan](#)
- [Langkah 2: Periksa konfigurasi metode API Gateway untuk server Anda](#)
- [Langkah 3: Lihat detail server Transfer Family](#)
- [Langkah 4: Uji apakah pengguna Anda dapat terhubung ke server](#)
- [Langkah 5: Uji SFTP koneksi dan transfer file](#)
- [Langkah 6: Batasi akses ke ember](#)
- [Perbarui Lambda jika menggunakan Amazon EFS](#)

Prasyarat

Sebelum Anda membuat resource Transfer Family di AWS CloudFormation, buat penyimpanan dan peran pengguna Anda.

Untuk menentukan penyimpanan dan membuat peran pengguna

1. Bergantung pada penyimpanan yang Anda gunakan, lihat dokumentasi berikut:
 - Untuk membuat bucket Amazon S3, lihat [Bagaimana cara membuat bucket S3?](#) di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.
 - Untuk membuat sistem EFS file Amazon, lihat [Konfigurasi sistem EFS file Amazon](#).
2. Untuk membuat peran pengguna, lihat [Buat IAM peran dan kebijakan](#)

Anda memasukkan detail untuk penyimpanan dan peran pengguna saat membuat AWS CloudFormation tumpukan di bagian berikutnya.

Langkah 1: Buat CloudFormation tumpukan

Untuk membuat AWS CloudFormation tumpukan dari template yang disediakan

1. Buka AWS CloudFormation konsol di <https://console.aws.amazon.com/cloudformation>.
2. Pilih Buat tumpukan, dan pilih Dengan sumber daya baru (standar).
3. Di panel Prasyarat - Siapkan templat, pilih Template sudah siap.
4. Salin tautan ini, [template tumpukan dasar](#), dan tempel ke bidang Amazon S3 URL.
5. Klik Berikutnya.
6. Tentukan parameter, termasuk nama untuk tumpukan Anda. Pastikan untuk melakukan hal berikut:
 - Ganti nilai default untuk UserName dan UserPassword.
 - Untuk UserHomeDirectory, masukkan detail penyimpanan (baik bucket Amazon S3 atau EFS sistem file Amazon) yang Anda buat sebelumnya.
 - Ganti default UserRoleArn dengan peran pengguna yang Anda buat sebelumnya. Peran AWS Identity and Access Management (IAM) harus memiliki izin yang sesuai. Untuk contoh kebijakan IAM peran dan bucket, lihat [Langkah 6: Batasi akses ke ember](#).
 - Jika Anda ingin mengautentikasi menggunakan kunci publik alih-alih kata sandi, masukkan kunci publik Anda di bidang UserPublicKey1. Pertama kali Anda terhubung ke server menggunakan SFTP, Anda kemudian memberikan kunci pribadi alih-alih kata sandi.
7. Pilih Berikutnya, lalu pilih Berikutnya lagi di halaman Configure stack options.
8. Tinjau detail tumpukan yang Anda buat, lalu pilih Buat tumpukan.

Note

Di bagian bawah halaman, di bawah Kemampuan, Anda harus mengakui bahwa AWS CloudFormation mungkin membuat IAM sumber daya.

Langkah 2: Periksa konfigurasi metode API Gateway untuk server Anda

Note

Untuk meningkatkan keamanan, Anda dapat mengkonfigurasi firewall aplikasi web. AWS WAF adalah firewall aplikasi web yang memungkinkan Anda memantau HTTP dan HTTPS permintaan yang diteruskan ke Amazon API Gateway. Untuk detailnya, lihat [Tambahkan firewall aplikasi web](#).

Untuk memeriksa konfigurasi metode API Gateway untuk server Anda dan menerapkannya

1. Buka konsol API Gateway di <https://console.aws.amazon.com/apigateway/>.
2. Pilih template dasar Transfer Custom Identity Provider API yang dihasilkan AWS CloudFormation template.
3. Di panel Resources, pilih GET, lalu pilih Permintaan Metode.
4. Untuk Tindakan, pilih Terapkan API. Untuk tahap Deployment, pilih prod, lalu pilih Deploy.

Setelah metode API Gateway berhasil digunakan, lihat kinerjanya di bagian Editor Panggung.

Note

Salin URL alamat Invoke yang muncul di bagian atas halaman. Anda akan membutuhkannya untuk langkah selanjutnya.

Langkah 3: Lihat detail server Transfer Family

Saat Anda menggunakan template untuk membuat AWS CloudFormation tumpukan, server Transfer Family dibuat secara otomatis.

Untuk melihat detail server Transfer Family

1. Buka AWS CloudFormation konsol di <https://console.aws.amazon.com/cloudformation>.
2. Pilih tumpukan yang Anda buat.
3. Pilih tab Sumber Daya.

Resources (18)			
Search resources			
Logical ID	Physical ID	Type	
ApiCloudWatchLogsRole	-ApiCloudWatchLogsRole-	AWS::IAM::Role	
ApiDeployment202008		AWS::ApiGateway::Deployment	
ApiLoggingAccount		AWS::ApiGateway::Account	
ApiStage	prod	AWS::ApiGateway::Stage	
CloudWatchLoggingRole	-CloudWatchLoggingRole-	AWS::IAM::Role	
CustomIdentityProviderApi		AWS::ApiGateway::RestApi	
GetUserConfigLambda	-GetUserConfigLambda-	AWS::Lambda::Function	
GetUserConfigLambdaPermission	GetUserConfigLambdaPermission-	AWS::Lambda::Permission	
GetUserConfigRequest		AWS::ApiGateway::Method	
GetUserConfigResource		AWS::ApiGateway::Resource	
GetUserConfigResponseModel	UserConfigResponseModel	AWS::ApiGateway::Model	
LambdaExecutionRole	-LambdaExecutionRole-	AWS::IAM::Role	
ServerIdResource		AWS::ApiGateway::Resource	
ServersResource		AWS::ApiGateway::Resource	
TransferIdentityProviderRole	-TransferIdentityProviderRole-	AWS::IAM::Role	
TransferServer	arn:aws:transfer:us-east-2:::server/s-	AWS::Transfer::Server	
UserNameResource		AWS::ApiGateway::Resource	
UsersResource		AWS::ApiGateway::Resource	

Server ARN ditampilkan di kolom ID Fisik untuk TransferServerbaris. ID server terkandung dalam, misalnya s-11112222333344445. ARN

- Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>, dan pada halaman Server, pilih server baru.

ID server cocok dengan ID yang ditampilkan untuk TransferServersumber daya di AWS CloudFormation.

Langkah 4: Uji apakah pengguna Anda dapat terhubung ke server

Untuk menguji apakah pengguna Anda dapat terhubung ke server, menggunakan konsol Transfer Family

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Pada halaman Server, pilih server baru Anda, pilih Tindakan, lalu pilih Uji.
3. Masukkan teks untuk kredensi login Anda ke dalam bidang Nama Pengguna, dan ke bidang Kata Sandi. Ini adalah nilai yang Anda tetapkan saat Anda menerapkan AWS CloudFormation tumpukan.
4. Untuk Protokol Server, pilih SFTP, dan untuk IP Sumber, masukkan **127.0.0.1**.
5. Pilih Uji.

Jika otentikasi pengguna berhasil, pengujian mengembalikan `Status Code: 200 HTML` respons dan JSON objek yang berisi rincian peran dan izin pengguna. Sebagai contoh:

```
{
  "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\"
}
```

Jika pengujian gagal, tambahkan salah satu kebijakan AWS terkelola API Gateway ke peran yang Anda gunakan untuk AndaAPI.

Langkah 5: Uji SFTP koneksi dan transfer file

Untuk menguji SFTP koneksi

1. Pada perangkat Linux atau macOS, buka terminal perintah.
2. Masukkan salah satu perintah berikut, tergantung pada apakah Anda menggunakan kata sandi atau key pair untuk otentikasi.
 - Jika Anda menggunakan kata sandi, masukkan perintah ini:

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

Saat diminta, masukkan kata sandi Anda.

- Jika Anda menggunakan key pair, masukkan perintah ini:

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

Note

Untuk `sftp` perintah ini, masukkan kode Wilayah AWS tempat server Transfer Family Anda berada. Misalnya, jika server Anda berada di AS Timur (Ohio), masukkan **us-east-2**.

3. Pada `sftp>` prompt, pastikan bahwa Anda dapat meng-upload (put), download (get), dan melihat direktori dan file (`pwd` dan `ls`).

Langkah 6: Batasi akses ke ember

Anda dapat membatasi siapa yang dapat mengakses bucket Amazon S3 tertentu. Contoh berikut menunjukkan setelan yang akan digunakan dalam CloudFormation tumpukan Anda dan dalam kebijakan yang Anda pilih untuk pengguna Anda.

Dalam contoh ini, kami menetapkan parameter berikut untuk AWS CloudFormation tumpukan:

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

Important

Ini adalah contoh kata sandi. Saat Anda mengonfigurasi metode API Gateway Anda, pastikan Anda memasukkan kata sandi yang kuat.

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

UserPublicKey1 adalah kunci publik yang telah Anda hasilkan sebagai bagian dari public/private key pair.

role-id Ini unik untuk peran pengguna yang Anda buat. Kebijakan yang dilampirkan `myuser-api-gateway-role` adalah sebagai berikut:

```
{
  "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/*"
    }
  ]
}
```

Untuk terhubung ke server menggunakan SFTP, masukkan salah satu perintah berikut pada prompt.

- Jika Anda menggunakan kata sandi untuk mengautentikasi, jalankan perintah berikut:

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

Saat diminta, masukkan kata sandi Anda.

- Jika Anda menggunakan key pair untuk mengautentikasi, jalankan perintah berikut:

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

Note

Untuk sftp perintah ini, gunakan ID Wilayah AWS tempat server Transfer Family Anda berada. Misalnya, jika server Anda berada di AS Timur (Ohio), gunakan `us-east-2`.

Pada sftp prompt, Anda diarahkan ke direktori home Anda, yang dapat Anda lihat dengan menjalankan `pwd` perintah. Sebagai contoh:

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

Pengguna tidak dapat melihat direktori apa pun di atas direktori home. Sebagai contoh:

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

Perbarui Lambda jika menggunakan Amazon EFS

Jika Anda memilih Amazon EFS sebagai opsi penyimpanan untuk server Transfer Family, Anda perlu mengedit fungsi lambda untuk tumpukan Anda.

Untuk menambahkan profil posix ke fungsi Lambda Anda

1. Buka konsol Lambda di <https://console.aws.amazon.com/lambda/>
2. Pilih fungsi Lambda yang Anda buat sebelumnya. Fungsi Lambda memiliki format ***stack-name***-GetUserConfigLambda-***lambda-identifier***, tempat ***stack-name*** adalah nama CloudFormation tumpukan dan ***lambda-identifier*** adalah pengenal untuk fungsi tersebut.

3. Di tab Kode, pilih index.js untuk menampilkan kode untuk fungsi tersebut.
4. Dalam response, tambahkan baris berikut antara Policy dan HomeDirectory:

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

Dimana *uid-value* and *gid-value* adalah bilangan bulat, 0 atau lebih besar, yang mewakili ID Pengguna dan ID Grup masing-masing.

Misalnya, setelah Anda menambahkan profil Posix, bidang respons mungkin terlihat seperti berikut:

```
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 '/'  
};
```

Menyiapkan AS2 konfigurasi

Tutorial ini berjalan melalui cara mengatur konfigurasi Applicability Statement 2 (AS2) dengan AWS Transfer Family. Setelah Anda menyelesaikan langkah-langkah yang dijelaskan di sini, Anda akan memiliki server AS2 berkemampuan yang siap menerima AS2 pesan dari mitra dagang sampel. Anda juga akan memiliki konektor yang dapat digunakan untuk mengirim AS2 pesan ke mitra dagang sampel.

Note

Beberapa bagian dari pengaturan contoh menggunakan AWS Command Line Interface (AWS CLI). Jika Anda belum menginstal AWS CLI, lihat [Menginstal atau memperbarui versi terbaru dari AWS CLI](#) Panduan AWS Command Line Interface Pengguna.

1. Buat sertifikat untuk diri sendiri dan mitra dagang Anda. Jika Anda memiliki sertifikat yang dapat Anda gunakan, Anda dapat melewati bagian ini.

Proses ini dijelaskan di [Langkah 1: Buat sertifikat untuk AS2](#).

2. Buat AWS Transfer Family server yang menggunakan AS2 protokol. Secara opsional, Anda dapat menambahkan alamat IP Elastis ke server untuk membuatnya menghadap ke internet.

Proses ini dijelaskan di [Langkah 2: Buat server Transfer Family yang menggunakan AS2 protokol](#).

 Note

Anda harus membuat server Transfer Family hanya untuk transfer masuk. Jika Anda hanya melakukan transfer keluar, Anda tidak memerlukan server Transfer Family.

3. Impor sertifikat yang Anda buat di langkah 1.

Proses ini dijelaskan di [Langkah 3: Impor sertifikat sebagai sumber sertifikat Transfer Family](#).

4. Untuk mengatur mitra dagang Anda, buat profil lokal dan profil mitra.

Proses ini dijelaskan di [Langkah 4: Buat profil untuk Anda dan mitra dagang Anda](#).

5. Buat perjanjian antara Anda dan mitra dagang Anda.

Proses ini dijelaskan di [Langkah 5: Buat kesepakatan antara Anda dan pasangan](#).

 Note

Anda harus membuat perjanjian untuk transfer masuk saja. Jika Anda hanya melakukan transfer keluar, Anda tidak memerlukan perjanjian.

6. Buat konektor antara Anda dan mitra dagang Anda.

Proses ini dijelaskan di [Langkah 6: Buat konektor antara Anda dan pasangan](#).

 Note

Anda harus membuat konektor untuk transfer keluar saja. Jika Anda hanya melakukan transfer masuk, Anda tidak memerlukan konektor.

7. Uji pertukaran AS2 file.

Proses ini dijelaskan di [Langkah 7: Uji pertukaran file AS2 dengan menggunakan Transfer Family](#).

Setelah Anda menyelesaikan langkah-langkah ini, Anda dapat melakukan hal berikut:

- Kirim file ke server mitra AS2 berkemampuan jarak jauh dengan perintah Transfer Family `start-file-transfer` AWS Command Line Interface (AWS CLI).
- Terima file dari server mitra AS2 berkemampuan jarak jauh di port 5080 melalui titik akhir virtual private cloud (VPC) Anda.

Langkah 1: Buat sertifikat untuk AS2

Kedua belah pihak dalam AS2 pertukaran membutuhkan sertifikat X.509. Anda dapat membuat sertifikat ini dengan cara apa pun yang Anda suka. Topik ini menjelaskan cara menggunakan [Buka SSL](#) dari baris perintah untuk membuat sertifikat root, dan kemudian menandatangani sertifikat bawahan. Kedua belah pihak harus membuat sertifikat mereka sendiri.

Note

Panjang kunci untuk AS2 sertifikat harus setidaknya 2048 bit, dan paling banyak 4096.

Untuk mentransfer file dengan mitra, perhatikan hal-hal berikut:

- Anda dapat melampirkan sertifikat ke profil. Sertifikat berisi kunci publik atau pribadi.
- Mitra dagang Anda mengirimi Anda kunci publik mereka, dan Anda mengirimkannya milik Anda.
- Mitra dagang Anda mengenkripsi pesan dengan kunci publik Anda dan menandatangani dengan kunci pribadi mereka. Sebaliknya, Anda mengenkripsi pesan dengan kunci publik mitra Anda dan menandatangani dengan kunci pribadi Anda.

Note

Jika Anda lebih suka mengelola kunci dengan GUI, [Portecle](#) adalah salah satu opsi yang dapat Anda gunakan.

Untuk menghasilkan contoh sertifikat

Important

Jangan mengirim kunci pribadi Anda kepada pasangan Anda. Dalam contoh ini, Anda membuat satu set kunci publik dan pribadi yang ditandatangani sendiri untuk satu pihak. Jika Anda akan bertindak sebagai mitra dagang untuk tujuan pengujian, Anda dapat mengulangi instruksi ini untuk menghasilkan dua set kunci: satu untuk setiap mitra dagang. Dalam hal ini, Anda tidak perlu menghasilkan dua otoritas sertifikat root (CAs).

1. Jalankan perintah berikut untuk menghasilkan kunci RSA pribadi dengan modulus sepanjang 2048 bit.

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

2. Jalankan perintah berikut untuk membuat sertifikat yang ditandatangani sendiri dengan `root-ca-key.pem` file Anda.

```
/usr/bin/openssl req \  
-x509 -new -nodes -sha256 \  
-days 1825 \  
-subj "/C=US/ST=MA/L=Boston/O=TransferFamilyCustomer/OU=IT-dept/CN=R00TCA" \  
-key root-ca-key.pem \  
-out root-ca.pem
```

-subj Argumen terdiri dari nilai-nilai berikut.

	Nama	Penjelasan
C	Kode negara	Kode dua huruf untuk negara tempat organisasi Anda berada.
ST	Negara bagian, wilayah, atau provinsi	Negara bagian, wilayah, atau provinsi tempat organisasi Anda berada. (Dalam hal ini,

	Nama	Penjelasan
		wilayah tidak mengacu pada Anda Wilayah AWS.)
L	Nama lokal	Kota tempat organisasi Anda berada.
O	Nama Organisasi	Nama resmi lengkap organisasi Anda, termasuk sufiks, seperti, CorpLLC, dan sebagainya.
OU	Nama unit organisasi	Divisi dalam organisasi Anda yang berhubungan dengan sertifikat ini.
CN	Nama umum atau nama domain yang sepenuhnya memenuhi syarat (FQDN)	Dalam hal ini, kami membuat sertifikat root, jadi nilainya R00TCA. Dalam contoh-contoh ini, kami menggunakan CN untuk menggambarkan tujuan sertifikat.

3. Buat kunci penandatanganan dan kunci enkripsi untuk profil lokal Anda.

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

Note

Beberapa AS2 server yang diaktifkan, seperti OpenAS2, mengharuskan Anda menggunakan sertifikat yang sama untuk penandatanganan dan enkripsi. Dalam hal ini, Anda dapat mengimpor kunci pribadi dan sertifikat yang sama untuk kedua tujuan. Untuk melakukannya, jalankan perintah ini alih-alih dua perintah sebelumnya:

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

4. Jalankan perintah berikut untuk membuat Permintaan Penandatanganan Sertifikat (CSRs) agar kunci root ditandatangani.

```
/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. Selanjutnya, Anda harus membuat `signing-cert.conf` file dan `encryption-cert.conf` file.

- Gunakan editor teks untuk membuat `signing-cert.conf` file dengan konten berikut:

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

- Gunakan editor teks untuk membuat `encryption-cert.conf` file dengan konten berikut:

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

6. Terakhir, Anda membuat sertifikat yang ditandatangani dengan menjalankan perintah berikut.

```
/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
```

Langkah 2: Buat server Transfer Family yang menggunakan AS2 protokol

Prosedur ini menjelaskan cara membuat server AS2 yang diaktifkan dengan menggunakan Transfer Family AWS CLI.

Note

Banyak contoh langkah menggunakan perintah yang memuat parameter dari file. Untuk detail selengkapnya tentang menggunakan file untuk memuat parameter, lihat [Cara memuat parameter dari file](#).

Jika Anda ingin menggunakan konsol sebagai gantinya, lihat [Membuat AS2 server menggunakan konsol Transfer Family](#).

Mirip dengan cara Anda membuat FTPS AWS Transfer Family server SFTP atau server, Anda membuat AS2 server yang diaktifkan dengan menggunakan `--protocols AS2` parameter `create-server` AWS CLI perintah. Saat ini, Transfer Family hanya mendukung tipe VPC endpoint dan penyimpanan Amazon S3 dengan AS2 protokol.

Saat Anda membuat server yang AS2 diaktifkan untuk Transfer Family dengan menggunakan `create-server` perintah, VPC titik akhir secara otomatis dibuat untuk Anda. Titik akhir ini mengekspos TCP port 5080 sehingga dapat menerima pesan. AS2

Jika Anda ingin mengekspos VPC titik akhir Anda secara publik ke internet, Anda dapat mengaitkan alamat IP Elastic dengan titik akhir Anda. VPC

Untuk menggunakan petunjuk ini, Anda memerlukan yang berikut:

- ID Anda VPC (misalnya, `vpc-abcdef01`).
- VPCSubnet Anda (misalnya, `subnet-abcdef01`, `01`, `subnet-021345ab`). IDs subnet-subnet-abcdef
- Satu atau lebih IDs grup keamanan yang memungkinkan lalu lintas masuk pada TCP port 5080 dari mitra dagang Anda (misalnya, `sg-1234567890abcdef0` dan `sg-abcdef01234567890`).
- (Opsional) Alamat IP Elastis yang ingin Anda kaitkan dengan VPC titik akhir Anda.
- Jika mitra dagang Anda tidak terhubung dengan Anda VPC melalui aVPN, Anda memerlukan gateway internet. Untuk informasi selengkapnya, lihat [Connect to the internet menggunakan gateway internet](#) di Panduan VPC Pengguna Amazon.

Untuk membuat server AS2 yang diaktifkan

1. Jalankan perintah berikut. Ganti masing-masing *user input placeholder* dengan informasi Anda sendiri.

```
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. (Opsional) Anda dapat membuat VPC titik akhir publik. Anda dapat melampirkan alamat IP Elastis ke server Transfer Family hanya melalui update-server operasi. Perintah berikut menghentikan server, memperbaruinya dengan alamat IP Elastis, dan kemudian mulai lagi.

```
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-abcd012345ccccccc
```

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

start-serverPerintah ini secara otomatis membuat DNS catatan untuk Anda yang berisi alamat IP publik untuk server Anda. Untuk memberi mitra dagang Anda akses ke server, Anda memberi mereka informasi berikut. Dalam hal ini, *your-region* mengacu pada Anda Wilayah AWS.

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

Lengkap URL yang Anda berikan kepada mitra dagang Anda adalah sebagai berikut:

<http://s-your-server-id.server.transfer.your-region.amazonaws.com:5080>

3. Untuk menguji apakah server AS2 -enabled Anda dapat diakses, gunakan perintah berikut. Pastikan server Anda dapat diakses baik melalui DNS alamat pribadi VPC titik akhir Anda, atau melalui titik akhir publik Anda (jika Anda mengaitkan alamat IP Elastis dengan titik akhir Anda).

Jika server Anda dikonfigurasi dengan benar, koneksi akan berhasil. Namun, Anda akan menerima respons kode HTTP status 400 (Permintaan Buruk) karena Anda tidak mengirim AS2 pesan yang valid.

- Untuk titik akhir publik (jika Anda mengaitkan alamat IP Elastis pada langkah sebelumnya), jalankan perintah berikut, ganti ID server dan Region Anda.

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

- Jika Anda terhubung di dalam AndaVPC, cari DNS nama pribadi VPC titik akhir Anda dengan menjalankan perintah berikut.

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

`describe-server` Perintah ini mengembalikan ID VPC endpoint Anda dalam `VpcEndpointId` parameter. Gunakan nilai ini untuk menjalankan perintah berikut.

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

`describe-vpc-endpoints` Perintah ini mengembalikan `DNSEntries` array, dengan beberapa `DnsName` parameter. Gunakan DNS nama Regional (yang tidak menyertakan Availability Zone) dalam perintah berikut.

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

Misalnya, perintah berikut menunjukkan nilai sampel untuk placeholder di perintah sebelumnya.

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

4. (Opsional) Konfigurasi peran logging. Transfer Family mencatat status pesan yang dikirim dan diterima dalam JSON format terstruktur ke CloudWatch log Amazon. Untuk memberi Transfer Family akses ke CloudWatch log di akun Anda, Anda harus mengonfigurasi peran pencatatan di server Anda.

Buat peran AWS Identity and Access Management (IAM) yang dipercayai `transfer.amazonaws.com`, dan lampirkan kebijakan `AWSTransferLoggingAccess` terkelola. Untuk detailnya, lihat [Buat IAM peran dan kebijakan](#).

Perhatikan Amazon Resource Name (ARN) dari IAM peran yang baru saja Anda buat, dan kaitkan dengan server dengan menjalankan update-server perintah berikut:

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

 Note

Meskipun peran logging bersifat opsional, kami sangat menyarankan untuk mengaturnya sehingga Anda dapat melihat status pesan Anda dan memecahkan masalah konfigurasi.

Langkah 3: Impor sertifikat sebagai sumber sertifikat Transfer Family

Prosedur ini menjelaskan cara mengimpor sertifikat dengan menggunakan AWS CLI. Jika Anda ingin menggunakan konsol Transfer Family sebagai gantinya, lihat [the section called “Impor AS2 sertifikat”](#).

Untuk mengimpor sertifikat penandatanganan dan enkripsi yang Anda buat di langkah 1, jalankan import-certificate perintah berikut. Jika Anda menggunakan sertifikat yang sama untuk enkripsi dan penandatanganan, impor sertifikat yang sama dua kali (sekali dengan SIGNING penggunaan dan sekali lagi dengan ENCRYPTION penggunaan).

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

Perintah ini mengembalikan penandatanganan Anda `CertificateId`. Pada bagian selanjutnya, ID sertifikat ini disebut sebagai *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
```

Perintah ini mengembalikan enkripsi Anda `CertificateId`. Pada bagian selanjutnya, ID sertifikat ini disebut sebagai *my-encrypt-cert-id*.

Selanjutnya, impor enkripsi mitra Anda dan tandatangani sertifikat dengan menjalankan perintah berikut.

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

Perintah ini mengembalikan enkripsi pasangan AndaCertificateId. Pada bagian selanjutnya, ID sertifikat ini disebut sebagai *partner-encrypt-cert-id*.

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

Perintah ini mengembalikan penandatanganan pasangan AndaCertificateId. Pada bagian selanjutnya, ID sertifikat ini disebut sebagai *partner-signing-cert-id*.

Langkah 4: Buat profil untuk Anda dan mitra dagang Anda

Prosedur ini menjelaskan cara membuat AS2 profil dengan menggunakan AWS CLI. Jika Anda ingin menggunakan konsol Transfer Family sebagai gantinya, lihat [the section called “Buat AS2 profil”](#).

Buat AS2 profil lokal Anda dengan menjalankan perintah berikut. Perintah ini mereferensikan sertifikat yang berisi kunci publik dan pribadi Anda.

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

Perintah ini mengembalikan ID profil Anda. Pada bagian selanjutnya, ID ini disebut sebagai *my-profile-id*.

Sekarang buat profil mitra dengan menjalankan perintah berikut. Perintah ini hanya menggunakan sertifikat kunci publik mitra Anda. Untuk menggunakan perintah ini, ganti *user input placeholders* dengan informasi Anda sendiri; misalnya, AS2 nama dan sertifikat pasangan AndaIDs.

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

Perintah ini mengembalikan ID profil mitra Anda. Pada bagian selanjutnya, ID ini disebut sebagai *partner-profile-id*.

 Note

Pada perintah sebelumnya, ganti *MYCORP* dengan nama organisasi Anda, dan *PARTNER-COMPANY* dengan nama organisasi mitra dagang Anda.

Langkah 5: Buat kesepakatan antara Anda dan pasangan

Prosedur ini menjelaskan cara membuat AS2 perjanjian dengan menggunakan AWS CLI. Jika Anda ingin menggunakan konsol Transfer Family sebagai gantinya, lihat [the section called “Buat AS2 perjanjian”](#).

Perjanjian menyatukan dua profil (lokal dan mitra), sertifikat mereka, dan konfigurasi server yang memungkinkan AS2 transfer masuk antara dua pihak. Anda dapat membuat daftar item Anda dengan menjalankan perintah berikut.

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

Langkah ini memerlukan bucket dan IAM peran Amazon S3 dengan akses baca/tulis ke dan dari bucket. Instruksi untuk membuat peran ini sama dengan Transfer FamilySFTP,FTP, dan FTPS protokol dan tersedia di [Buat IAM peran dan kebijakan](#)

Untuk membuat perjanjian, Anda memerlukan item berikut:

- Nama bucket Amazon S3 (dan awalan objek, jika ditentukan)
- ARN IAM Peran dengan akses ke ember
- ID server Transfer Family Anda
- ID profil Anda dan ID profil mitra Anda

Buat perjanjian dengan menjalankan perintah berikut.

```
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
```


Jika berhasil, perintah ini mengembalikan ID untuk perjanjian. Anda kemudian dapat melihat detail perjanjian dengan perintah berikut.

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

Langkah 6: Buat konektor antara Anda dan pasangan

Prosedur ini menjelaskan cara membuat AS2 konektor dengan menggunakan AWS CLI. Jika Anda ingin menggunakan konsol Transfer Family sebagai gantinya, lihat [the section called “Konfigurasi AS2 konektor”](#).

Anda dapat menggunakan `StartFileTransfer` API operasi untuk mengirim file yang disimpan di Amazon S3 ke AS2 titik akhir mitra dagang Anda dengan menggunakan konektor. Anda dapat menemukan profil yang Anda buat sebelumnya dengan menjalankan perintah berikut.

```
aws transfer list-profiles
```

Saat Anda membuat konektor, Anda harus menyediakan AS2 server mitra Anda URL. Salin teks berikut ke file bernama `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

Untuk `EncryptionAlgorithm`, jangan tentukan `DES_EDE3_CBC` algoritme kecuali Anda harus mendukung klien lama yang membutuhkannya, karena ini adalah algoritma enkripsi yang lemah.

Kemudian jalankan perintah berikut untuk membuat konektor.

```
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
```

Langkah 7: Uji pertukaran file AS2 dengan menggunakan Transfer Family

Menerima file dari mitra dagang Anda

Jika Anda mengaitkan alamat IP Elastis publik dengan VPC titik akhir Anda, Transfer Family secara otomatis membuat DNS nama yang berisi alamat IP publik Anda. Subdomain adalah ID AWS Transfer Family server Anda (dari format `s-1234567890abcdef0`). Berikan server Anda URL kepada mitra dagang Anda dalam format berikut.

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

Jika Anda tidak mengaitkan alamat IP Elastis publik dengan VPC titik akhir Anda, cari nama host dari VPC titik akhir yang dapat menerima AS2 pesan HTTP POST dari mitra dagang Anda di port 5080. Untuk mengambil rincian VPC endpoint, gunakan perintah berikut.

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

Misalnya, asumsikan perintah sebelumnya mengembalikan ID VPC endpoint dari `vpce-1234abcd5678efghi`. Kemudian, Anda akan menggunakan perintah berikut untuk mengambil DNS nama.

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

Perintah ini mengembalikan semua rincian untuk VPC endpoint yang Anda butuhkan untuk menjalankan perintah berikut.

DNSNama tercantum dalam `DnsEntries` array. Mitra dagang Anda harus berada di dalam Anda VPC untuk mengakses VPC titik akhir Anda (misalnya melalui AWS PrivateLink atau aVPN). Berikan VPC titik akhir Anda URL kepada pasangan Anda dalam format berikut.

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

Misalnya, berikut ini URL menunjukkan nilai sampel untuk placeholder di perintah sebelumnya.

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

Dalam contoh ini, transfer yang berhasil disimpan di lokasi yang ditentukan dalam `base-directory` parameter yang Anda tentukan [Langkah 5: Buat kesepakatan antara Anda dan pasangan](#). Jika kami berhasil menerima file bernama `myfile1.txt` dan `myfile2.txt`, file disimpan sebagai `/path-defined-in-the-agreement/processed/original_filename.messageId.original_extension`. Di sini, file disimpan sebagai `/DOC-EXAMPLE-DESTINATION-BUCKET/AS2-inbox/processed/myfile1.messageId.txt` dan `/DOC-EXAMPLE-DESTINATION-BUCKET/AS2-inbox/processed/myfile2.messageId.txt`.

Jika Anda mengonfigurasi peran logging saat membuat server Transfer Family, Anda juga dapat memeriksa CloudWatch log untuk status AS2 pesan.

Kirim file ke mitra dagang Anda

Anda dapat menggunakan Transfer Family untuk mengirim AS2 pesan dengan mereferensikan ID konektor dan jalur ke file, seperti yang diilustrasikan dalam perintah `start-file-transfer` AWS Command Line Interface (AWS CLI) berikut:

```
aws transfer start-file-transfer --connector-id c-1234567890abcdef0 \
--send-file-paths "/DOC-EXAMPLE-SOURCE-BUCKET/myfile1.txt" "/DOC-EXAMPLE-SOURCE-BUCKET/myfile2.txt"
```

Untuk mendapatkan detail konektor Anda, jalankan perintah berikut:

```
aws transfer list-connectors
```

`list-connectors` Perintah mengembalikan konektorIDs, URLs, dan Amazon Resource Names (ARNs) untuk konektor Anda.

Untuk mengembalikan properti konektor tertentu, jalankan perintah berikut dengan ID yang ingin Anda gunakan:

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

`describe-connector` Perintah mengembalikan semua properti untuk konektor, termasuk, peran, profil URL, Pemberitahuan Disposisi Pesan (MDNs), tag, dan metrik pemantauan.

Anda dapat mengonfirmasi bahwa mitra berhasil menerima file dengan melihat JSON dan MDN file. File-file ini diberi nama sesuai dengan konvensi yang dijelaskan dalam [Nama dan lokasi file](#). Jika Anda mengonfigurasi peran logging saat membuat konektor, Anda juga dapat memeriksa CloudWatch log Anda untuk status AS2 pesan.

Mengkonfigurasi titik SFTP akhir,FTPS, atau FTP server

Topik ini memberikan rincian untuk membuat dan menggunakan endpoint AWS Transfer Family server yang menggunakan satu atau lebih dari SFTP,FTPS, dan FTP protokol.

Topik

- [Opsi penyedia identitas](#)
- [AWS Transfer Family matriks tipe titik akhir](#)
- [Mengkonfigurasi titik SFTP akhir,FTPS, atau FTP server](#)
- [Mentransfer file melalui titik akhir server menggunakan klien](#)
- [Mengelola pengguna untuk titik akhir server](#)
- [Menggunakan direktori logis untuk menyederhanakan struktur direktori Transfer Family](#)

Opsi penyedia identitas

AWS Transfer Family menyediakan beberapa metode untuk mengautentikasi dan mengelola pengguna. Tabel berikut membandingkan penyedia identitas yang tersedia yang dapat Anda gunakan dengan Transfer Family.

Tindakan	AWS Transfer Family layanan dikelola	AWS Managed Microsoft AD	API Gerbang Amazon	AWS Lambda
Protokol yang didukung	SFTP	SFTP, FTPS, FTP	SFTP, FTPS, FTP	SFTP, FTPS, FTP
Otentikasi berbasis kunci	Ya	Tidak	Ya	Ya
Autentikasi kata sandi	Tidak	Ya	Ya	Ya
AWS Identity and Access Managemen	Ya	Ya	Ya	Ya

Tindakan	AWS Transfer Family layanan dikelola	AWS Managed Microsoft AD	API Gerbang Amazon	AWS Lambda
t (IAM) dan POSIX				
Direktori home logis	Ya	Ya	Ya	Ya
Akses parameter (berbasis nama pengguna)	Ya	Ya	Ya	Ya
Struktur akses ad hoc	Ya	Tidak	Ya	Ya
AWS WAF	Tidak	Tidak	Ya	Tidak

Catatan:

- IAM digunakan untuk mengontrol akses untuk penyimpanan dukungan Amazon S3, dan POSIX digunakan untuk Amazon. EFS
- Ad hoc mengacu pada kemampuan untuk mengirim profil pengguna saat runtime. Misalnya, Anda dapat mendaratkan pengguna di direktori home mereka dengan meneruskan nama pengguna sebagai variabel.
- Untuk detailnya AWS WAF, lihat [Tambahkan firewall aplikasi web](#).
- Ada posting blog yang menjelaskan penggunaan fungsi Lambda yang terintegrasi dengan Microsoft Azure AD sebagai penyedia identitas Transfer Family Anda. Untuk detailnya, lihat [Mengautentikasi AWS Transfer Family dengan Azure Active Directory](#) dan. AWS Lambda
- Kami menyediakan beberapa AWS CloudFormation template untuk membantu Anda dengan cepat menyebarkan server Transfer Family yang menggunakan penyedia identitas khusus. Untuk detailnya, lihat [Template fungsi Lambda](#).

Dalam prosedur berikut, Anda dapat membuat server SFTP -enabled, server -enabled, server FTPS -enabled, atau AS2 server FTP -enabled.

Langkah selanjutnya

- [Buat server SFTP yang diaktifkan](#)
- [Buat server FTPS yang diaktifkan](#)
- [Buat server FTP yang diaktifkan](#)
- [Mengkonfigurasi AS2](#)

AWS Transfer Family matriks tipe titik akhir

Saat membuat server Transfer Family, Anda memilih jenis endpoint yang akan digunakan. Tabel berikut menjelaskan karakteristik untuk setiap jenis titik akhir.

Matriks tipe titik akhir

Karakteristik	Publik	VPC- Internet	VPC- Internal	VPC_Titik akhir (tidak digunakan lagi)
Protokol yang didukung	SFTP	SFTP, FTPS, AS2	SFTP, FTP, FTPS, AS2	SFTP
Akses	Dari internet. Jenis titik akhir ini tidak memerlukan konfigurasi khusus apa pun di Anda VPC.	Melalui internet dan dari dalam VPC dan lingkungan VPC yang terhubung, seperti pusat data lokal melalui AWS Direct Connect atau. VPN	Dari dalam VPC dan lingkungan VPC yang terhubung, seperti pusat data lokal di atas AWS Direct Connect atau. VPN	Dari dalam VPC dan lingkungan VPC yang terhubung, seperti pusat data lokal di atas AWS Direct Connect atau. VPN
Alamat IP statis	Anda tidak dapat melampirkan alamat IP statis. AWS menyediakan alamat IP	Anda dapat melampirkan alamat IP Elastis ke titik akhir. Ini bisa berupa alamat AWS IP	Alamat IP pribadi yang dilampirkan ke titik akhir tidak berubah.	Alamat IP pribadi yang dilampirkan ke titik akhir tidak berubah.

Karakteristik	Publik	VPC- Internet	VPC- Internal	VPC_Titik akhir (tidak digunakan lagi)
	yang dapat berubah.	milik atau alamat IP Anda sendiri (Bawa alamat IP Anda sendiri). Alamat IP elastis yang dilampirkan ke titik akhir tidak berubah. Alamat IP pribadi yang dilampirkan ke server juga tidak berubah.		

Karakteristik	Publik	VPC- Internet	VPC- Internal	VPC_Titik akhir (tidak digunakan lagi)
Daftar izin IP sumber	Jenis titik akhir ini tidak mendukung daftar izin berdasarkan alamat IP sumber. Titik akhir dapat diakses publik dan mendengarkan lalu lintas melalui port 22.  Note Untuk endpoint yang VPC di-host, server SFTP Transfer Family dapat beroperasi melalui port 22 (default) atau port 2222.	Untuk mengizinkan akses berdasarkan alamat IP sumber, Anda dapat menggunakan grup keamanan yang dilampirkan ke titik akhir server dan jaringan yang ACLs dilampirkan ke subnet tempat titik akhir berada.	Untuk mengizinkan akses berdasarkan alamat IP sumber, Anda dapat menggunakan grup keamanan yang dilampirkan ke titik akhir server dan daftar kontrol akses jaringan (jaringanACLs) yang dilampirkan ke subnet tempat titik akhir berada.	Untuk mengizinkan akses berdasarkan alamat IP sumber, Anda dapat menggunakan grup keamanan yang dilampirkan ke titik akhir server dan jaringan yang ACLs dilampirkan ke subnet tempat titik akhir berada.

Karakteristik	Publik	VPC- Internet	VPC- Internal	VPC_Titik akhir (tidak digunakan lagi)
Daftar izin firewall klien	Anda harus mengizinkan DNS nama server. Karena alamat IP dapat berubah, hindari menggunakan alamat IP untuk daftar izin firewall klien Anda.	Anda dapat mengizinkan DNS nama server atau alamat IP Elastis yang dilampirkan ke server.	Anda dapat mengizinkan alamat IP pribadi atau DNS nama titik akhir.	Anda dapat mengizinkan alamat IP pribadi atau DNS nama titik akhir.

Note

Jenis VPC_ENDPOINT endpoint sekarang sudah usang dan tidak dapat digunakan untuk membuat server baru. Alih-alih menggunakan `EndpointType=VPC_ENDPOINT`, gunakan tipe VPC endpoint baru (`EndpointType=VPC`), yang dapat Anda gunakan sebagai Internal atau Internet Facing, seperti yang dijelaskan dalam tabel sebelumnya. Untuk detailnya, lihat [Menghentikan penggunaan _ VPC ENDPOINT](#).

Pertimbangkan opsi berikut untuk meningkatkan postur keamanan AWS Transfer Family server Anda:

- Gunakan VPC titik akhir dengan akses internal, sehingga server hanya dapat diakses oleh klien dalam lingkungan Anda VPC atau VPC yang terhubung seperti pusat data lokal di atas atau. AWS Direct Connect VPN
- Untuk memungkinkan klien mengakses titik akhir melalui internet dan melindungi server Anda, gunakan VPC titik akhir dengan akses yang menghadap ke internet. Kemudian, ubah grup

keamanan untuk mengizinkan lalu lintas hanya dari alamat IP tertentu yang meng-host klien pengguna Anda. VPC

- Jika Anda memerlukan otentikasi berbasis kata sandi dan Anda menggunakan penyedia identitas khusus dengan server Anda, itu adalah praktik terbaik bahwa kebijakan kata sandi Anda mencegah pengguna membuat kata sandi yang lemah dan membatasi jumlah upaya login yang gagal.
- AWS Transfer Family adalah layanan terkelola, sehingga tidak menyediakan akses shell. Anda tidak dapat langsung mengakses SFTP server yang mendasarinya untuk menjalankan perintah asli OS di server Transfer Family.
- Gunakan Network Load Balancer di depan VPC titik akhir dengan akses internal. Ubah port listener pada penyeimbang beban dari port 22 ke port yang berbeda. Ini dapat mengurangi, tetapi tidak menghilangkan, risiko pemindai port dan bot yang menyelidiki server Anda, karena port 22 paling sering digunakan untuk pemindaian. Untuk detailnya, lihat posting blog [Network Load Balancers sekarang mendukung grup Keamanan](#).

Note

Jika Anda menggunakan Network Load Balancer, AWS Transfer Family CloudWatch log menampilkan alamat IP untuk NLB, bukan alamat IP klien yang sebenarnya.

Mengkonfigurasi titik SFTP akhir, FTPS, atau FTP server

Anda dapat membuat server transfer file dengan menggunakan AWS Transfer Family layanan ini. Protokol transfer file berikut tersedia:

- Secure Shell (SSH) Protokol Transfer File (SFTP) - Transfer file SSH. Untuk detailnya, lihat [the section called “Buat server SFTP yang diaktifkan”](#).

Note

Kami memberikan AWS CDK contoh untuk membuat server SFTP Transfer Family. Contoh menggunakan TypeScript, dan tersedia di GitHub [sini](#).

- File Transfer Protocol Secure (FTPS) — Transfer file dengan TLS enkripsi. Untuk detailnya, lihat [the section called “Buat server FTPS yang diaktifkan”](#).
- Protokol Transfer File (FTP) — Transfer file tidak terenkripsi. Untuk detailnya, lihat [the section called “Buat server FTP yang diaktifkan”](#).

- Pernyataan Penerapan 2 (AS2) — Transfer file untuk mengangkut data terstruktur business-to-business. Untuk detailnya, lihat [the section called “Konfigurasi AS2”](#). Untuk AS2, Anda dapat dengan cepat membuat AWS CloudFormation tumpukan untuk tujuan demonstrasi. Prosedur ini dijelaskan dalam [Gunakan template untuk membuat AS2 tumpukan Transfer Family demo](#).

Anda dapat membuat server dengan beberapa protokol.

 Note

Jika Anda memiliki beberapa protokol yang diaktifkan untuk titik akhir server yang sama dan Anda ingin memberikan akses dengan menggunakan nama pengguna yang sama melalui beberapa protokol, Anda dapat melakukannya selama kredensi khusus untuk protokol telah diatur di penyedia identitas Anda. Untuk FTP, kami sarankan untuk mempertahankan kredensial terpisah dari SFTP dan FTPS. Ini karena, tidak seperti SFTP dan FTPS, FTP mentransmisikan kredensial dalam teks yang jelas. Dengan mengisolasi FTP kredensial dari SFTP atau FTPS, jika FTP kredensial dibagikan atau diekspos, beban kerja Anda menggunakan atau tetap aman. SFTP FTPS

Saat Anda membuat server, Anda memilih spesifik Wilayah AWS untuk melakukan permintaan operasi file pengguna yang ditugaskan ke server tersebut. Seiring dengan menetapkan server satu atau beberapa protokol, Anda juga menetapkan salah satu jenis penyedia identitas berikut:

- Layanan dikelola dengan menggunakan SSH kunci. Untuk detailnya, lihat [Bekerja dengan pengguna yang dikelola layanan](#).
- AWS Directory Service for Microsoft Active Directory (AWS Managed Microsoft AD). Metode ini memungkinkan Anda mengintegrasikan grup Microsoft Active Directory untuk menyediakan akses ke server Transfer Family Anda. Untuk detailnya, lihat [Menggunakan penyedia identitas AWS Directory Service](#).
- Sebuah metode kustom. Metode penyedia identitas kustom menggunakan AWS Lambda atau Amazon API Gateway dan memungkinkan Anda mengintegrasikan layanan direktori untuk mengautentikasi dan mengotorisasi pengguna Anda. Layanan secara otomatis menetapkan pengenal yang secara unik mengidentifikasi server Anda. Untuk detailnya, lihat [Bekerja dengan penyedia identitas khusus](#). Transfer Family menyediakan AWS CloudFormation template yang dapat Anda gunakan untuk menyebarkan server dengan cepat yang menggunakan penyedia identitas khusus.

- [Fungsi Lambda untuk otentikasi](#) menjelaskan CloudFormation template yang menggunakan fungsi Lambda untuk otentikasi.
- [Mengautentikasi menggunakan metode API Gateway](#) menjelaskan CloudFormation template yang menggunakan metode Amazon API Gateway untuk otentikasi.

Anda juga menetapkan server jenis endpoint (dapat diakses publik atau VPC dihosting) dan nama host dengan menggunakan endpoint server default, atau nama host kustom dengan menggunakan layanan Amazon Route 53 atau dengan menggunakan layanan Domain Name System DNS () pilihan Anda. Nama host server harus unik di Wilayah AWS tempat pembuatannya.

Selain itu, Anda dapat menetapkan peran CloudWatch pencatatan Amazon untuk mendorong peristiwa ke CloudWatch log Anda, memilih kebijakan keamanan yang berisi algoritme kriptografi yang diaktifkan untuk digunakan oleh server Anda, dan menambahkan metadata ke server dalam bentuk tag yang merupakan pasangan nilai kunci.

Important

Anda dikenakan biaya untuk server instantiated dan untuk transfer data. Untuk informasi tentang harga dan penggunaan AWS Kalkulator Harga untuk mendapatkan perkiraan biaya penggunaan Transfer Family, lihat [AWS Transfer Family harga](#).

Buat server SFTP yang diaktifkan

Secure Shell (SSH) File Transfer Protocol (SFTP) adalah protokol jaringan yang digunakan untuk transfer data yang aman melalui internet. Protokol ini mendukung fungsionalitas keamanan dan otentikasi penuh. SSH ini banyak digunakan untuk bertukar data, termasuk informasi sensitif antara mitra bisnis di berbagai industri seperti layanan keuangan, perawatan kesehatan, ritel, dan periklanan.

Note

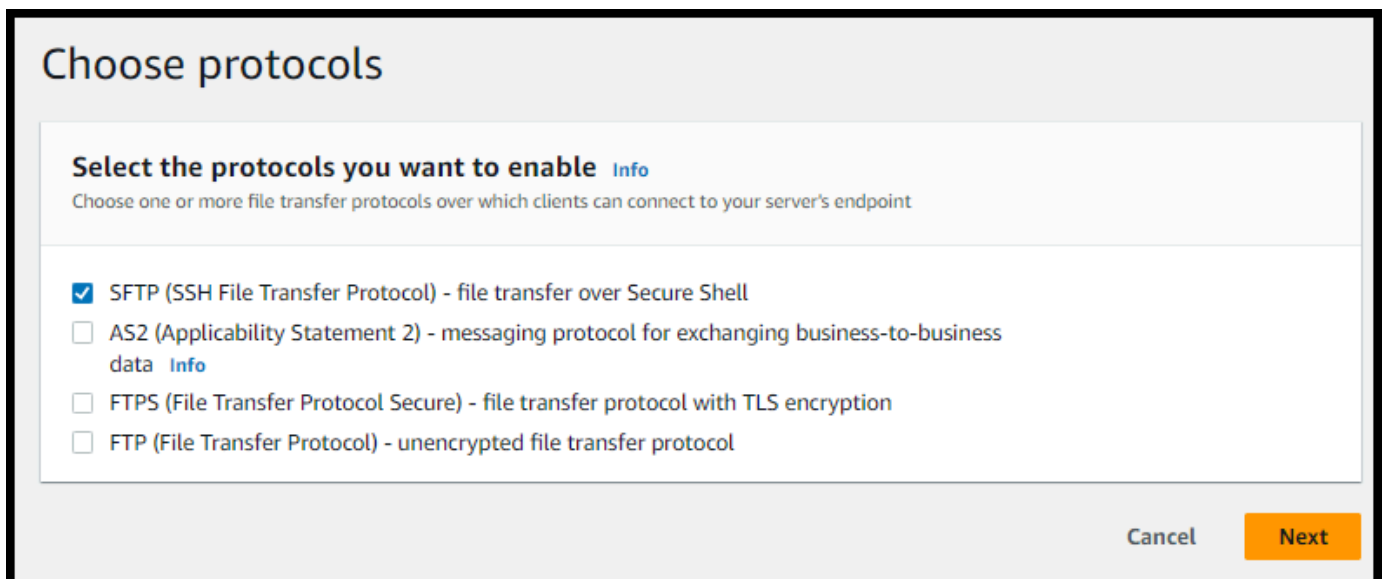
SFTPserver untuk Transfer Family beroperasi melalui port 22. Untuk endpoint yang VPC di-host, server SFTP Transfer Family juga dapat beroperasi melalui port 2222. Untuk rincian selengkapnya, lihat [Buat server di cloud pribadi virtual](#).

Lihat juga

- Kami memberikan AWS CDK contoh untuk membuat server SFTP Transfer Family. Contoh menggunakan TypeScript, dan tersedia di GitHub [sini](#).
- Untuk panduan tentang cara menerapkan server Transfer Family di dalam fileVPC, lihat [Gunakan daftar izin IP untuk mengamankan](#) server Anda. AWS Transfer Family

Untuk membuat server SFTP yang diaktifkan

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/> dan pilih Server dari panel navigasi, lalu pilih Buat server.
2. Di Pilih protokol, pilih SFTP, lalu pilih Berikutnya.



Choose protocols

Select the protocols you want to enable [Info](#)

Choose one or more file transfer protocols over which clients can connect to your server's endpoint

- ☒ SFTP (SSH File Transfer Protocol) - file transfer over Secure Shell
- ☐ AS2 (Applicability Statement 2) - messaging protocol for exchanging business-to-business data [Info](#)
- ☐ FTPS (File Transfer Protocol Secure) - file transfer protocol with TLS encryption
- ☐ FTP (File Transfer Protocol) - unencrypted file transfer protocol

Cancel Next

3. Di Pilih penyedia identitas, pilih penyedia identitas yang ingin Anda gunakan untuk mengelola akses pengguna. Anda memiliki opsi berikut:
 - Layanan dikelola - Anda menyimpan identitas dan kunci pengguna. AWS Transfer Family

Choose an identity provider

Identity provider

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

[Cancel](#) [Previous](#) [Next](#)

- AWS Directory Service for Microsoft Active Directory— Anda menyediakan AWS Directory Service direktori untuk mengakses titik akhir. Dengan demikian, Anda dapat menggunakan kredensial yang disimpan di Active Directory untuk mengautentikasi pengguna Anda. Untuk mempelajari lebih lanjut tentang bekerja dengan penyedia AWS Managed Microsoft AD identitas, lihat [Menggunakan penyedia identitas AWS Directory Service](#).

Note

- Direktori Cross-Account dan Shared tidak didukung untuk AWS Managed Microsoft AD
- Untuk menyiapkan server dengan Directory Service sebagai penyedia identitas Anda, Anda perlu menambahkan beberapa AWS Directory Service izin. Untuk detailnya, lihat [Sebelum Anda mulai menggunakan AWS Directory Service for Microsoft Active Directory](#).

Choose an identity provider

Identity provider

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 [Provider Info](#)
Manage users by integrating an identity provider of your choice

Directory
TATER3

Cancel Previous Next

- Penyedia identitas khusus - Pilih salah satu opsi berikut:
 - Gunakan AWS Lambda untuk menghubungkan penyedia identitas Anda — Anda dapat menggunakan penyedia identitas yang ada, didukung oleh fungsi Lambda. Anda memberikan nama fungsi Lambda. Untuk informasi selengkapnya, lihat [Menggunakan AWS Lambda untuk mengintegrasikan penyedia identitas Anda](#).
 - Gunakan Amazon API Gateway untuk menghubungkan penyedia identitas Anda — Anda dapat membuat metode API Gateway yang didukung oleh fungsi Lambda untuk digunakan sebagai penyedia identitas. Anda menyediakan Amazon API Gateway URL dan peran pemanggilan. Untuk informasi selengkapnya, lihat [Menggunakan Amazon API Gateway untuk mengintegrasikan penyedia identitas Anda](#).

Untuk salah satu opsi, Anda juga dapat menentukan cara mengautentikasi.

- Kata Sandi ATAU Kunci — pengguna dapat mengautentikasi dengan kata sandi atau kunci mereka. Ini adalah nilai default.
- Kata sandi ONLY — pengguna harus memberikan kata sandi mereka untuk terhubung.
- Kunci ONLY — pengguna harus menyediakan kunci pribadi mereka untuk terhubung.
- ANDKunci Kata Sandi — pengguna harus memberikan kunci pribadi dan kata sandi mereka untuk terhubung. Server memeriksa kunci terlebih dahulu, dan kemudian jika kuncinya valid,

sistem meminta kata sandi. Jika kunci pribadi yang diberikan tidak cocok dengan kunci publik yang disimpan, otentikasi gagal.

Choose an 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

AWS Lambda function

Choose a Lambda function

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

[i](#) Either a valid password or valid private key will be required during user authentication

Cancel Previous Next

4. Pilih Berikutnya.
5. Di Pilih titik akhir, lakukan hal berikut:
 - a. Untuk tipe Endpoint, pilih tipe titik akhir yang dapat diakses publik. Untuk titik akhir yang VPCdihosting, lihat[Buat server di cloud pribadi virtual](#).
 - b. (Opsional) Untuk nama host Kustom, pilih Tidak Ada.

Anda mendapatkan nama host server yang disediakan oleh AWS Transfer Family. Nama host server mengambil formulir `serverId.server.transfer.regionId.amazonaws.com`.

Untuk nama host kustom, Anda menentukan alias kustom untuk endpoint server Anda. Untuk mempelajari lebih lanjut tentang bekerja dengan nama host kustom, lihat [Bekerja dengan nama host khusus](#).

- c. (Opsional) Untuk FIPSDiaktifkan, pilih kotak centang FIPSEnabled endpoint untuk memastikan bahwa titik akhir sesuai dengan Standar Pemrosesan Informasi Federal (). FIPS

Note

FIPSEndpoint yang diaktifkan hanya tersedia di Wilayah Amerika AWS Utara. Untuk Wilayah yang tersedia, lihat [AWS Transfer Family titik akhir dan kuota](#) di Referensi Umum AWS Untuk informasi selengkapnya FIPS, lihat [Federal Information Processing Standard \(FIPS\) 140-2](#).

- d. Pilih Berikutnya.

Choose an endpoint

Endpoint configuration [Info](#)

Endpoint type
Select whether the endpoint will be publicly accessible or hosted inside your VPC

☒ **Publicly accessible**
Accessible over the internet

☐ **VPC hosted** [Info](#)
Access controlled using Security Groups

Custom hostname
Specify a custom alias for your server endpoint.

None ▼

FIPS Enabled
Select whether the endpoint should comply with Federal Information Processing Standards (FIPS)

☐ **FIPS Enabled endpoint**

Cancel Previous **Next**

6. Pada halaman Pilih domain, pilih layanan AWS penyimpanan yang ingin Anda gunakan untuk menyimpan dan mengakses data Anda melalui protokol yang dipilih:

- Pilih Amazon S3 untuk menyimpan dan mengakses file Anda sebagai objek di atas protokol yang dipilih.
- Pilih Amazon EFS untuk menyimpan dan mengakses file Anda di sistem EFS file Amazon Anda melalui protokol yang dipilih.

Pilih Berikutnya.

7. Di Konfigurasikan detail tambahan, lakukan hal berikut:

- a. Untuk logging, tentukan grup log yang ada atau buat yang baru (opsi default).

Jika Anda memilih grup log yang ada, Anda harus memilih salah satu yang terkait dengan Anda Akun AWS.

Transfer Family > Servers > Create server

Step 1
Choose protocols

Step 2
Choose an identity provider

Step 3
Choose an endpoint

Step 4
Choose a domain

Step 5
Configure additional details

Step 6
Review and create

Configure additional details

Logging Info

Log group Info
Choose the CloudWatch log group where your events will be delivered in a structured JSON format

☐ Create a new log group ☒ Choose an existing log group

Logging role Info
Choose the IAM role that will be used to deliver events to your CloudWatch logs

☐ Create a new role ☐ Choose an existing role

Logging role is only required when selecting a workflow in the Managed workflows section below.

Jika Anda memilih Buat grup log, CloudWatch konsol (<https://console.aws.amazon.com/cloudwatch/>) terbuka ke halaman Buat grup log. Untuk detailnya, lihat [Membuat grup log di CloudWatch Log](#).

- b. (Opsional) Untuk alur kerja Terkelola, pilih alur kerja IDs (dan peran terkait) yang harus diasumsikan oleh Transfer Family saat menjalankan alur kerja. Anda dapat memilih satu alur kerja untuk dieksekusi setelah unggahan lengkap, dan satu lagi untuk mengeksekusi pada unggahan sebagian. Untuk mempelajari lebih lanjut tentang memproses file Anda menggunakan alur kerja terkelola, lihat [AWS Transfer Family alur kerja terkelola](#).

Managed workflows Info

Workflow for complete file uploads
Select the workflow that AWS Transfer Family should run on all files that are uploaded in full via this server

Workflow for partial file uploads
Select the workflow that Transfer Family should run on all files that are only partially uploaded via this server

Managed workflows execution role Info
Select the role that AWS Transfer Family should assume when executing a workflow

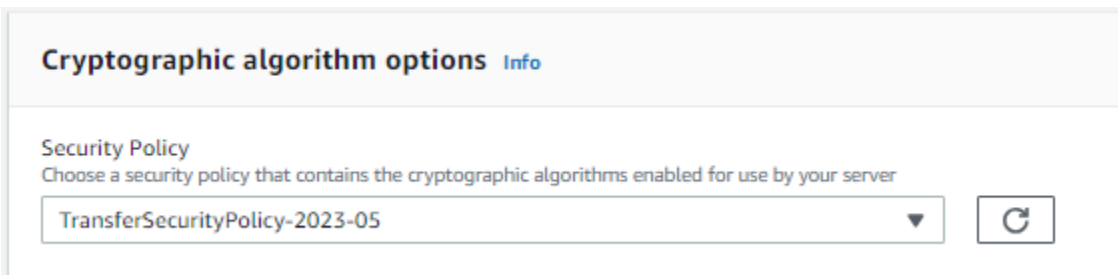
- c. Untuk opsi algoritma kriptografi, pilih kebijakan keamanan yang berisi algoritma kriptografi yang diaktifkan untuk digunakan oleh server Anda.

Note

Secara default:

- Jika Endpoint FIPS diaktifkan tidak dipilih, kebijakan `TransferSecurityPolicy-2020-06` keamanan dilampirkan ke server Anda.
- Jika Endpoint FIPS diaktifkan dipilih, kebijakan `TransferSecurityPolicy-FIPS-2020-06` keamanan dilampirkan ke server Anda.

Untuk informasi selengkapnya tentang kebijakan keamanan, lihat [Kebijakan keamanan untuk AWS Transfer Family](#).



Cryptographic algorithm options Info

Security Policy
Choose a security policy that contains the cryptographic algorithms enabled for use by your server

TransferSecurityPolicy-2023-05

- d. (Optional) Untuk Server Host Key RSAED25519, masukkan,, atau kunci ECDSA pribadi yang akan digunakan untuk mengidentifikasi server Anda ketika klien terhubung dengannya SFTP. Anda juga dapat menambahkan deskripsi untuk membedakan antara beberapa kunci host.

Setelah Anda membuat server Anda, Anda dapat menambahkan kunci host tambahan. Memiliki beberapa kunci host berguna jika Anda ingin memutar tombol atau jika Anda ingin memiliki berbagai jenis kunci, seperti RSA kunci dan juga ECDSA kunci.

Note

Bagian Server Host Key hanya digunakan untuk memigrasikan pengguna dari server SFTP berkemampuan yang ada.

Server Host Key [Info](#)

Private key - *optional*

Upload an RSA, ECDSA, or ED25519 private key that will be used to identify your SFTP server when clients connect to it. Additional keys can be added once the server is created.

Enter an optional RSA, ECDSA, or ED25519 key

Description - *optional*

Add a description to differentiate between multiple private keys

Enter optional description

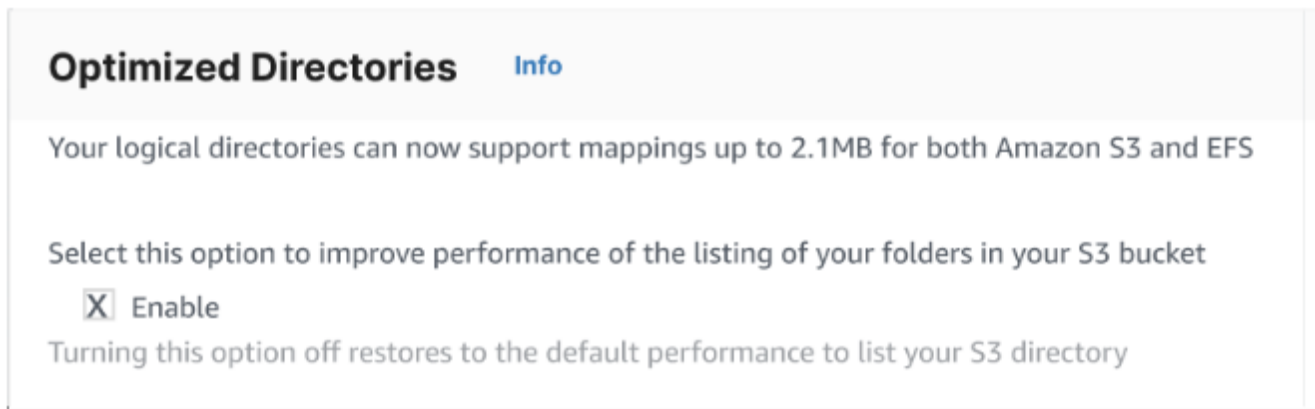
i You can ignore this section unless you are migrating users from an existing SFTP server.

- e. (Opsional) Untuk Tag, untuk Kunci dan Nilai, masukkan satu atau beberapa tag sebagai pasangan nilai kunci, lalu pilih Tambahkan tag.
- f. Pilih Berikutnya.

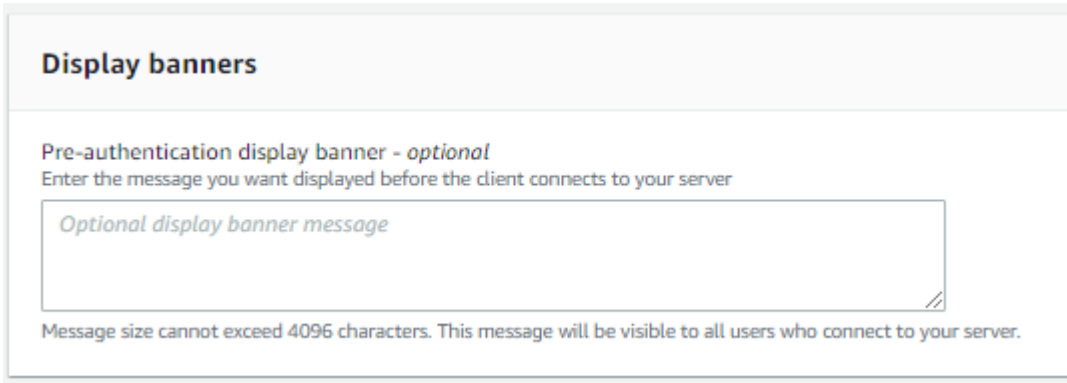
Tags

Key	Value	
Enter key	Enter value	Remove tag
Add tag		

- g. Anda dapat mengoptimalkan kinerja untuk direktori Amazon S3 Anda. Misalnya, Anda masuk ke direktori home Anda, dan Anda memiliki 10.000 subdirektori. Dengan kata lain, bucket S3 Anda memiliki 10.000 folder. Dalam skenario ini, jika Anda menjalankan perintah `ls` (daftar), operasi daftar memakan waktu antara enam dan delapan menit. Namun, jika Anda mengoptimalkan direktori Anda, operasi ini hanya membutuhkan beberapa detik.



- h. (Optional) Konfigurasi AWS Transfer Family server untuk menampilkan pesan yang disesuaikan seperti kebijakan organisasi atau syarat dan ketentuan kepada pengguna akhir Anda. Untuk spanduk Tampilan, di kotak teks spanduk tampilan Pra-otentikasi, masukkan pesan teks yang ingin ditampilkan kepada pengguna Anda sebelum mereka mengautentikasi.



- i. (Optional) Anda dapat mengonfigurasi opsi tambahan berikut.
- SetStat opsi: aktifkan opsi ini untuk mengabaikan kesalahan yang dihasilkan saat klien mencoba menggunakan SETSTAT pada file yang Anda unggah ke bucket Amazon S3. Untuk detail tambahan, lihat SetStatOption dokumentasi di [ProtocolDetails](#).
 - TLSdimulainya kembali sesi: opsi ini hanya tersedia jika Anda telah mengaktifkan FTPS sebagai salah satu protokol untuk server ini.
 - IP Pasif: opsi ini hanya tersedia jika Anda telah mengaktifkan FTPS atau FTP sebagai salah satu protokol untuk server ini.

Additional configuration

SetStat option - optional [Info](#)
Select whether you want this server to ignore SetStat command

☐ Enable

TLS session resumption - optional [Info](#)
Choose how you want your server to process TLS session resumption requests

☒ Enforce
☐ Enable
☐ Disable

 To enable TLS session resumption, enable FTPS as one of the protocols selected in Step 1

Passive IP - optional [Info](#)
Provide passive IP (PASV) that file transfer clients can use to connect this server

1.2.3.4

 To enable Passive IP, enable FTP or FTPS as one of the protocols selected in Step 1

8. Di Tinjau dan buat, tinjau pilihan Anda.

- Jika Anda ingin mengedit salah satu dari mereka, pilih Edit di sebelah langkah.

 Note

Anda harus meninjau setiap langkah setelah langkah yang Anda pilih untuk diedit.

- Jika Anda tidak memiliki perubahan, pilih Buat server untuk membuat server Anda. Anda dibawa ke halaman Server, ditampilkan berikut, di mana server baru Anda terdaftar.

Diperlukan beberapa menit sebelum status server baru Anda berubah menjadi Online. Pada saat itu, server Anda dapat melakukan operasi file untuk pengguna Anda.

Servers (1)							Actions	Add user	Create server
☐	Hostname	Server ID	State	users	Endpoint type	Domain			
☐	-	s-	Starting	No Users	Public	Amazon S3			

Buat server FTPS yang diaktifkan

File Transfer Protocol over SSL (FTPS) adalah ekstensi untuk FTP. Ini menggunakan protokol kriptografi Transport Layer Security (TLS) dan Secure Sockets Layer (SSL) untuk mengenkripsi lalu lintas. FTPS memungkinkan enkripsi koneksi kontrol dan saluran data baik secara bersamaan maupun independen.

Untuk membuat server FTPS yang diaktifkan

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/> dan pilih Server dari panel navigasi, lalu pilih Buat server.
2. Di Pilih protokol, pilih. FTPS

Untuk sertifikat Server, pilih sertifikat yang disimpan di AWS Certificate Manager (ACM) yang akan digunakan untuk mengidentifikasi server Anda ketika klien terhubung dengannya FTPS dan kemudian pilih Berikutnya.

Untuk meminta sertifikat publik baru, lihat [Meminta sertifikat publik](#) di Panduan AWS Certificate Manager Pengguna.

Untuk mengimpor sertifikat yang ada ACM, lihat [Mengimpor sertifikat ke ACM](#) dalam Panduan AWS Certificate Manager Pengguna.

Untuk meminta sertifikat pribadi untuk digunakan FTPS melalui alamat IP pribadi, lihat [Meminta Sertifikat Pribadi](#) di Panduan AWS Certificate Manager Pengguna.

Sertifikat dengan algoritme kriptografi dan ukuran kunci berikut didukung:

- 2048-bit RSA (_2048) RSA
- 4096-bit RSA (_4096) RSA
- Elliptic Prime Curve 256 bit (EC_prime256v1)
- Elliptic Prime Curve 384 bit (EC_secp384r1)
- Elliptic Prime Curve 521 bit (EC_secp521r1)

Note

Sertifikat harus berupa SSL TLS sertifikat/X.509 versi 3 yang valid dengan FQDN atau alamat IP yang ditentukan dan informasi tentang penerbit.

Choose protocols

Select the protocols you want to enable [Info](#)

Choose one or more file transfer protocols over which clients can connect to your server's endpoint

- ☐ SFTP (SSH File Transfer Protocol) - file transfer over Secure Shell
- ☐ AS2 (Applicability Statement 2) - messaging protocol for exchanging business-to-business data [Info](#)
- ☒ FTPS (File Transfer Protocol Secure) - file transfer protocol with TLS encryption
- ☐ FTP (File Transfer Protocol) - unencrypted file transfer protocol

AWS Certificate Manager (ACM) certificate [Info](#)

Server certificate
Choose a certificate stored in ACM which will be used to identify your server when clients connect to it over FTPS

[▼](#) [↻](#)

[Cancel](#) [Next](#)

3. Di Pilih penyedia identitas, pilih penyedia identitas yang ingin Anda gunakan untuk mengelola akses pengguna. Anda memiliki opsi berikut:
 - AWS Directory Service for Microsoft Active Directory— Anda menyediakan AWS Directory Service direktori untuk mengakses titik akhir. Dengan demikian, Anda dapat menggunakan kredensial yang disimpan di Active Directory untuk mengautentikasi pengguna Anda. Untuk mempelajari lebih lanjut tentang bekerja dengan penyedia AWS Managed Microsoft AD identitas, lihat [Menggunakan penyedia identitas AWS Directory Service](#).

Note

- Direktori Cross-Account dan Shared tidak didukung untuk AWS Managed Microsoft AD
- Untuk menyiapkan server dengan Directory Service sebagai penyedia identitas Anda, Anda perlu menambahkan beberapa AWS Directory Service izin. Untuk detailnya, lihat [Sebelum Anda mulai menggunakan AWS Directory Service for Microsoft Active Directory](#).

Choose an identity provider

Identity provider

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 **Provider Info**
Manage users by integrating an identity provider of your choice

Directory
TATER3

Cancel Previous Next

- Penyedia identitas khusus - Pilih salah satu opsi berikut:
 - Gunakan AWS Lambda untuk menghubungkan penyedia identitas Anda — Anda dapat menggunakan penyedia identitas yang ada, didukung oleh fungsi Lambda. Anda memberikan nama fungsi Lambda. Untuk informasi selengkapnya, lihat [Menggunakan AWS Lambda untuk mengintegrasikan penyedia identitas Anda](#).
 - Gunakan Amazon API Gateway untuk menghubungkan penyedia identitas Anda — Anda dapat membuat metode API Gateway yang didukung oleh fungsi Lambda untuk digunakan sebagai penyedia identitas. Anda menyediakan Amazon API Gateway URL dan peran

pemanggilan. Untuk informasi selengkapnya, lihat [Menggunakan Amazon API Gateway untuk mengintegrasikan penyedia identitas Anda](#).

Choose an 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

AWS Lambda function

Choose a Lambda function



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

i To choose an authentication method, enable SFTP as one of the protocols selected in Step 1

Cancel

Previous

Next

4. Pilih Berikutnya.
5. Di Pilih titik akhir, lakukan hal berikut:

i Note

FTPSServer untuk Transfer Family beroperasi melalui Port 21 (Control Channel) dan Port Range 8192—8200 (Saluran Data).

- a. Untuk tipe Endpoint, pilih tipe endpoint yang VPCdihosting untuk meng-host endpoint server Anda. Untuk informasi tentang menyiapkan titik akhir yang VPC dihosting, lihat [Buat server di cloud pribadi virtual](#).

 Note

Titik akhir yang dapat diakses publik tidak didukung.

- b. (Opsional) Untuk FIPSDiaktifkan, pilih kotak centang FIPSEnabled endpoint untuk memastikan bahwa titik akhir sesuai dengan Standar Pemrosesan Informasi Federal (). FIPS

 Note

FIPSEndpoint yang diaktifkan hanya tersedia di Wilayah Amerika AWS Utara. Untuk Wilayah yang tersedia, lihat [AWS Transfer Family titik akhir dan kuota](#) di Referensi Umum AWS Untuk informasi selengkapnyaFIPS, lihat [Federal Information Processing Standard \(FIPS\) 140-2](#).

- c. Pilih Berikutnya.

Choose an endpoint

Endpoint configuration [Info](#)

Endpoint type
Select whether the endpoint will be publicly accessible or hosted inside your VPC

☐ Publicly accessible
Accessible over the internet

☒ VPC hosted [Info](#)
Access controlled using Security Groups

Access [Info](#)

☒ Internal

☐ Internet Facing

VPC
Select a VPC ID



[Create a VPC](#) 

FIPS Enabled
Select whether the endpoint should comply with Federal Information Processing Standards (FIPS)

☐ FIPS Enabled endpoint

[Cancel](#) [Previous](#) [Next](#)

6. Pada halaman Pilih domain, pilih layanan AWS penyimpanan yang ingin Anda gunakan untuk menyimpan dan mengakses data Anda melalui protokol yang dipilih:

- Pilih Amazon S3 untuk menyimpan dan mengakses file Anda sebagai objek di atas protokol yang dipilih.
- Pilih Amazon EFS untuk menyimpan dan mengakses file Anda di sistem EFS file Amazon Anda melalui protokol yang dipilih.

Pilih Berikutnya.

7. Di Konfigurasi detail tambahan, lakukan hal berikut:

- a. Untuk logging, tentukan grup log yang ada atau buat yang baru (opsi default).

Transfer Family > Servers > Create server

Step 1
Choose protocols

Step 2
Choose an identity provider

Step 3
Choose an endpoint

Step 4
Choose a domain

Step 5
Configure additional details

Step 6
Review and create

Configure additional details

Logging Info

Log group Info
Choose the CloudWatch log group where your events will be delivered in a structured JSON format

☒ Create a new log group ☐ Choose an existing log group

Choose an existing log group

Logging role Info
Choose the IAM role that will be used to deliver events to your CloudWatch logs

☐ Create a new role ☐ Choose an existing role

Info Logging role is only required when selecting a workflow in the Managed workflows section below.

Jika Anda memilih grup log yang ada, Anda harus memilih salah satu yang terkait dengan Anda Akun AWS.

Transfer Family > Servers > Create server

Step 1
Choose protocols

Step 2
Choose an identity provider

Step 3
Choose an endpoint

Step 4
Choose a domain

Step 5
Configure additional details

Step 6
Review and create

Configure additional details

Logging Info

Log group Info
Choose the CloudWatch log group where your events will be delivered in a structured JSON format

☐ Create a new log group ☒ Choose an existing log group

/aws/transfer/

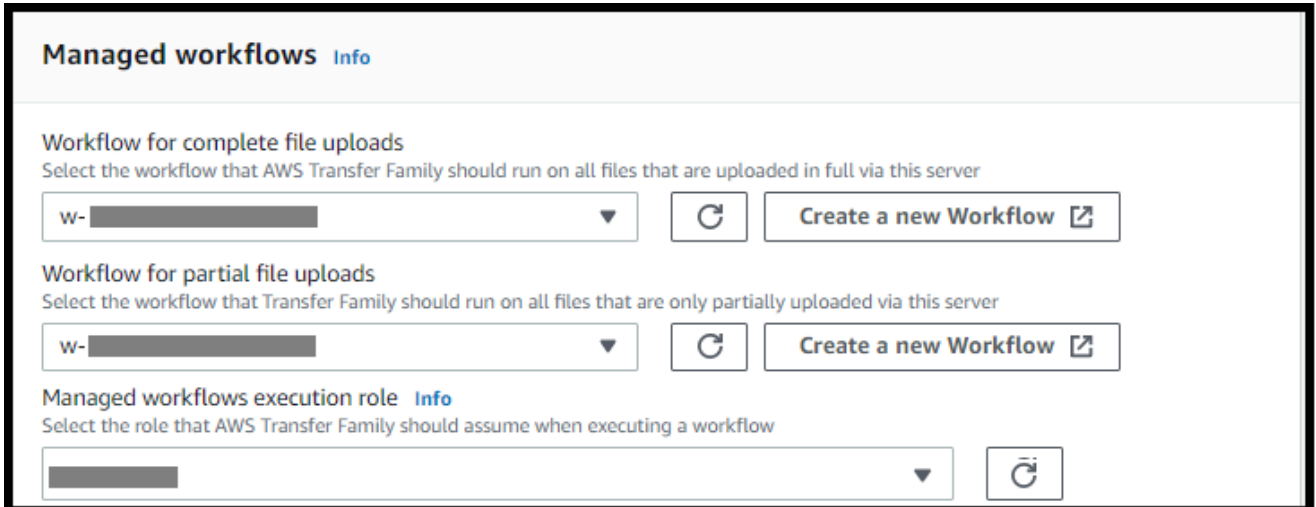
Logging role Info
Choose the IAM role that will be used to deliver events to your CloudWatch logs

☐ Create a new role ☐ Choose an existing role

Info Logging role is only required when selecting a workflow in the Managed workflows section below.

Jika Anda memilih Buat grup log, CloudWatch konsol (<https://console.aws.amazon.com/cloudwatch/>) terbuka ke halaman Buat grup log. Untuk detailnya, lihat [Membuat grup log di CloudWatch Log](#).

- b. (Opsional) Untuk alur kerja Terkelola, pilih alur kerja IDs (dan peran terkait) yang harus diasumsikan oleh Transfer Family saat menjalankan alur kerja. Anda dapat memilih satu alur kerja untuk dieksekusi setelah unggahan lengkap, dan satu lagi untuk mengeksekusi pada unggahan sebagian. Untuk mempelajari lebih lanjut tentang memproses file Anda menggunakan alur kerja terkelola, lihat [AWS Transfer Family alur kerja terkelola](#).



The screenshot shows the 'Managed workflows' section in the AWS Transfer Family console. It has a title 'Managed workflows' with an 'Info' link. Below it, there are two sections for selecting workflows:

- Workflow for complete file uploads:** 'Select the workflow that AWS Transfer Family should run on all files that are uploaded in full via this server'. It features a dropdown menu with 'w-' followed by a redacted ID, a refresh button, and a 'Create a new Workflow' button with an external link icon.
- Workflow for partial file uploads:** 'Select the workflow that Transfer Family should run on all files that are only partially uploaded via this server'. It also features a dropdown menu with 'w-' followed by a redacted ID, a refresh button, and a 'Create a new Workflow' button with an external link icon.

Below these is the **Managed workflows execution role** section with an 'Info' link. It says 'Select the role that AWS Transfer Family should assume when executing a workflow' and shows a dropdown menu with a redacted role name and a refresh button.

- c. Untuk opsi algoritma kriptografi, pilih kebijakan keamanan yang berisi algoritma kriptografi yang diaktifkan untuk digunakan oleh server Anda.

Note

Secara default:

- Jika Endpoint FIPS diaktifkan tidak dipilih, kebijakan `TransferSecurityPolicy-2020-06` keamanan dilampirkan ke server Anda.
- Jika Endpoint FIPS diaktifkan dipilih, kebijakan `TransferSecurityPolicy-FIPS-2020-06` keamanan dilampirkan ke server Anda.

Untuk informasi selengkapnya tentang kebijakan keamanan, lihat [Kebijakan keamanan untuk AWS Transfer Family](#).

Cryptographic algorithm options [Info](#)

Security Policy
Choose a security policy that contains the cryptographic algorithms enabled for use by your server

TransferSecurityPolicy-2023-05

▼



- d. Untuk Server Host Key, biarkan kosong.

 Note

Bagian Server Host Key hanya digunakan untuk memigrasikan pengguna dari server SFTP berkemampuan yang ada.

Server Host Key [Info](#)

Private key - optional

Upload an RSA, ECDSA, or ED25519 private key that will be used to identify your SFTP server when clients connect to it. Additional keys can be added once the server is created.

Enter an optional RSA, ECDSA, or ED25519 key

Description - optional

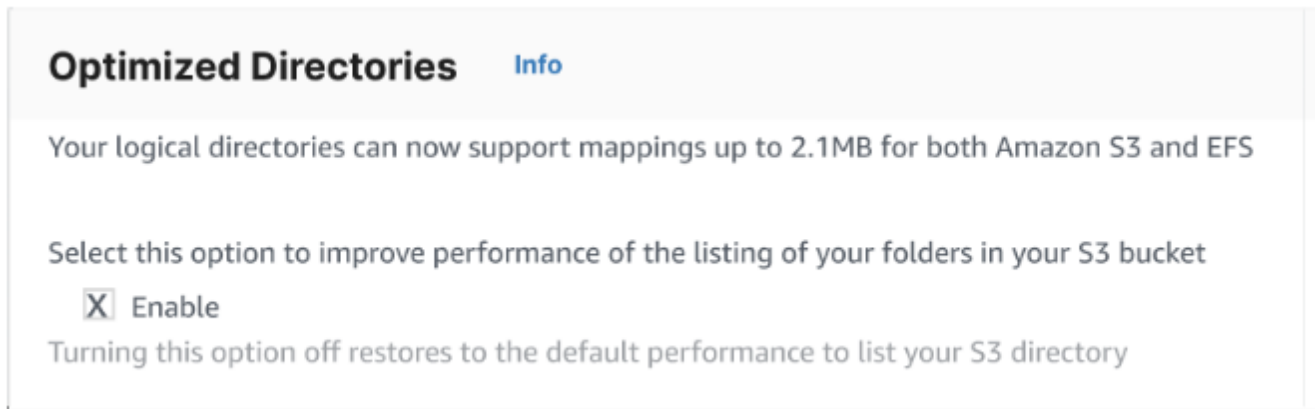
Add a description to differentiate between multiple private keys

Enter optional description

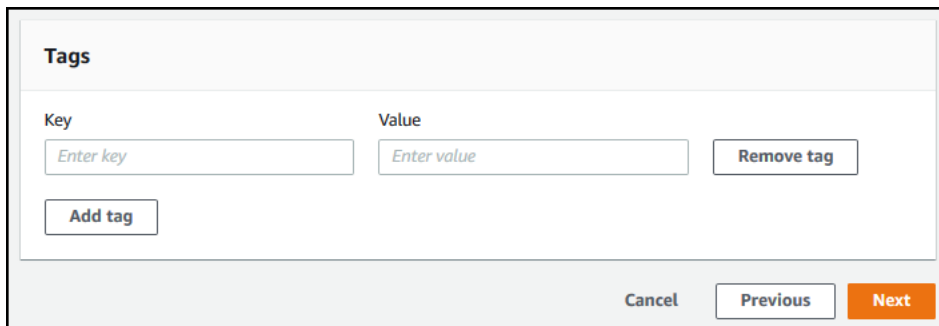
 You can ignore this section unless you are migrating users from an existing SFTP server.

- e. (Opsional) Untuk Tag, untuk Kunci dan Nilai, masukkan satu atau beberapa tag sebagai pasangan nilai kunci, lalu pilih Tambahkan tag.
- f. Anda dapat mengoptimalkan kinerja untuk direktori Amazon S3 Anda. Misalnya, Anda masuk ke direktori home Anda, dan Anda memiliki 10.000 subdirektori. Dengan kata lain, bucket S3 Anda memiliki 10.000 folder. Dalam skenario ini, jika Anda menjalankan perintah

1s (daftar), operasi daftar memakan waktu antara enam dan delapan menit. Namun, jika Anda mengoptimalkan direktori Anda, operasi ini hanya membutuhkan beberapa detik.



g. Pilih Berikutnya.



h. (Opsional) Anda dapat mengonfigurasi AWS Transfer Family server untuk menampilkan pesan yang disesuaikan seperti kebijakan organisasi atau syarat dan ketentuan kepada pengguna akhir Anda. Anda juga dapat menampilkan Message of The Day (MOTD) yang disesuaikan kepada pengguna yang telah berhasil diautentikasi.

Untuk spanduk Tampilan, di kotak teks spanduk tampilan Pra-otentikasi, masukkan pesan teks yang ingin ditampilkan kepada pengguna Anda sebelum mereka mengautentikasi, dan di kotak teks spanduk tampilan pasca-otentikasi, masukkan teks yang ingin ditampilkan kepada pengguna Anda setelah mereka berhasil mengautentikasi.

Display banners

Pre-authentication display banner - optional
Enter the message you want displayed before the client connects to your server

Optional display banner message

Message size cannot exceed 4096 characters. This message will be visible to all users who connect to your server.

Post-authentication display banner - optional
Enter the message you want displayed after the client has connected to your server

Optional display banner message

Message size cannot exceed 4096 characters. This message will be visible to all users who connect to your server.

i SFTP clients will only be able to see the pre-authentication message. FTPS and FTP clients will be able to see both pre-authentication and post-authentication messages.

- i. (Opsional) Anda dapat mengonfigurasi opsi tambahan berikut.
- SetStat opsi: aktifkan opsi ini untuk mengabaikan kesalahan yang dihasilkan saat klien mencoba menggunakan SETSTAT pada file yang Anda unggah ke bucket Amazon S3. Untuk detail tambahan, lihat SetStatOption dokumentasi dalam [ProtocolDetails](#) topik.
 - TLSDimulainya kembali sesi: menyediakan mekanisme untuk melanjutkan atau berbagi kunci rahasia yang dinegosiasikan antara kontrol dan koneksi data untuk suatu sesi. FTPS Untuk detail tambahan, lihat TlsSessionResumptionMode dokumentasi dalam [ProtocolDetails](#) topik.
 - IP pasif: menunjukkan mode pasif, untuk FTP dan FTPS protokol. Masukkan satu IPv4 alamat, seperti alamat IP publik firewall, router, atau penyeimbang beban. Untuk detail tambahan, lihat PassiveIp dokumentasi dalam [ProtocolDetails](#) topik.

Additional configuration

SetStat option - optional [Info](#)
Select whether you want this server to ignore SetStat command

☐ Enable

TLS session resumption - optional [Info](#)
Choose how you want your server to process TLS session resumption requests

☒ Enforce

☐ Enable

☐ Disable

Passive IP - optional [Info](#)
Provide passive IP (PASV) that file transfer clients can use to connect this server

1.2.3.4

8. Di Tinjau dan buat, tinjau pilihan Anda.

- Jika Anda ingin mengedit salah satu dari mereka, pilih Edit di sebelah langkah.

Note

Anda harus meninjau setiap langkah setelah langkah yang Anda pilih untuk diedit.

- Jika Anda tidak memiliki perubahan, pilih Buat server untuk membuat server Anda. Anda dibawa ke halaman Server, ditampilkan berikut, di mana server baru Anda terdaftar.

Diperlukan beberapa menit sebelum status server baru Anda berubah menjadi Online. Pada saat itu, server Anda dapat melakukan operasi file untuk pengguna Anda.

Servers (1)

Actions ▾

Add user

Create server

< 1 >

☐	Hostname	Server ID	State	users	Endpoint type	Domain
☐	-	s-	Starting	No Users	Public	Amazon S3

Langkah selanjutnya: Untuk langkah selanjutnya, lanjutkan [Bekerja dengan penyedia identitas khusus](#) untuk mengatur pengguna.

Buat server FTP yang diaktifkan

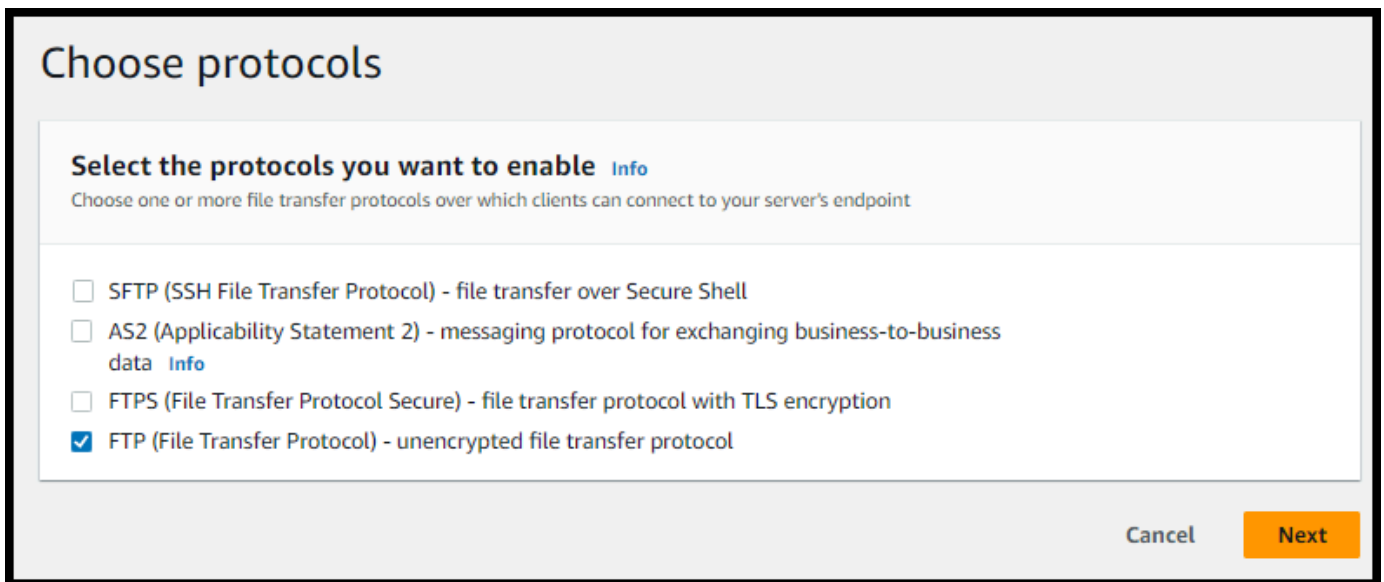
File Transfer Protocol (FTP) adalah protokol jaringan yang digunakan untuk transfer data. FTP menggunakan saluran terpisah untuk kontrol dan transfer data. Saluran kontrol terbuka hingga batas waktu dihentikan atau tidak aktif. Saluran data aktif selama transfer. FTP menggunakan teks yang jelas dan tidak mendukung enkripsi lalu lintas.

Note

Saat Anda mengaktifkan FTP, Anda harus memilih opsi akses internal untuk titik akhir yang VPC di-host. Jika Anda membutuhkan server Anda untuk memiliki data melintasi jaringan publik, Anda harus menggunakan protokol aman, seperti SFTP atau FTPS.

Untuk membuat server FTP yang diaktifkan

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/> dan pilih Server dari panel navigasi, lalu pilih Buat server.
2. Di Pilih protokol, pilih FTP, lalu pilih Berikutnya.



3. Di Pilih penyedia identitas, pilih penyedia identitas yang ingin Anda gunakan untuk mengelola akses pengguna. Anda memiliki opsi berikut:
 - AWS Directory Service for Microsoft Active Directory— Anda menyediakan AWS Directory Service direktori untuk mengakses titik akhir. Dengan demikian, Anda dapat menggunakan kredensial yang disimpan di Active Directory untuk mengautentikasi pengguna Anda.

Untuk mempelajari lebih lanjut tentang bekerja dengan penyedia AWS Managed Microsoft AD identitas, lihat [Menggunakan penyedia identitas AWS Directory Service](#).

Note

- Direktori Cross-Account dan Shared tidak didukung untuk AWS Managed Microsoft AD
- Untuk menyiapkan server dengan Directory Service sebagai penyedia identitas Anda, Anda perlu menambahkan beberapa AWS Directory Service izin. Untuk detailnya, lihat [Sebelum Anda mulai menggunakan AWS Directory Service for Microsoft Active Directory](#).

Choose an identity provider

Identity provider

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

Directory
TATER3

Cancel Previous Next

- Penyedia identitas khusus - Pilih salah satu opsi berikut:
 - Gunakan AWS Lambda untuk menghubungkan penyedia identitas Anda — Anda dapat menggunakan penyedia identitas yang ada, didukung oleh fungsi Lambda. Anda memberikan nama fungsi Lambda. Untuk informasi selengkapnya, lihat [Menggunakan AWS Lambda untuk mengintegrasikan penyedia identitas Anda](#).
 - Gunakan Amazon API Gateway untuk menghubungkan penyedia identitas Anda — Anda dapat membuat metode API Gateway yang didukung oleh fungsi Lambda untuk digunakan

sebagai penyedia identitas. Anda menyediakan Amazon API Gateway URL dan peran pemanggilan. Untuk informasi selengkapnya, lihat [Menggunakan Amazon API Gateway untuk mengintegrasikan penyedia identitas Anda](#).

Choose an 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

AWS Lambda function

Choose a Lambda function



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

To choose an authentication method, enable SFTP as one of the protocols selected in Step 1

Cancel

Previous

Next

4. Pilih Berikutnya.
5. Di Pilih titik akhir, lakukan hal berikut:

Note

FTPServer untuk Transfer Family beroperasi melalui Port 21 (Control Channel) dan Port Range 8192-8200 (Saluran Data).

- a. Untuk jenis Endpoint, pilih VPChost untuk meng-host endpoint server Anda. Untuk informasi tentang menyiapkan titik akhir yang VPC dihosting, lihat [Buat server di cloud pribadi virtual](#).

 Note

Titik akhir yang dapat diakses publik tidak didukung.

- b. Untuk FIPSDiaktifkan, biarkan kotak centang FIPSEnabled endpoint tetap bersih.

 Note

FIPSEndpoint yang diaktifkan tidak didukung untuk FTP server.

- c. Pilih Berikutnya.

Choose an endpoint

Endpoint configuration [Info](#)

Endpoint type
Select whether the endpoint will be publicly accessible or hosted inside your VPC

☐ Publicly accessible
Accessible over the internet

☒ VPC hosted [Info](#)
Access controlled using Security Groups

Access [Info](#)

☒ Internal

☐ Internet Facing

VPC
Select a VPC ID



[Create a VPC](#) 

FIPS Enabled
Select whether the endpoint should comply with Federal Information Processing Standards (FIPS)

☐ FIPS Enabled endpoint

[Cancel](#) [Previous](#) [Next](#)

6. Pada halaman Pilih domain, pilih layanan AWS penyimpanan yang ingin Anda gunakan untuk menyimpan dan mengakses data Anda melalui protokol yang dipilih.

- Pilih Amazon S3 untuk menyimpan dan mengakses file Anda sebagai objek di atas protokol yang dipilih.
- Pilih Amazon EFS untuk menyimpan dan mengakses file Anda di sistem EFS file Amazon Anda melalui protokol yang dipilih.

Pilih Berikutnya.

7. Di Konfigurasi detail tambahan, lakukan hal berikut:

- a. Untuk logging, tentukan grup log yang ada atau buat yang baru (opsi default).

Transfer Family > Servers > Create server

Step 1
Choose protocols

Step 2
Choose an identity provider

Step 3
Choose an endpoint

Step 4
Choose a domain

Step 5
Configure additional details

Step 6
Review and create

Configure additional details

Logging Info

Log group Info
Choose the CloudWatch log group where your events will be delivered in a structured JSON format

☒ Create a new log group ☐ Choose an existing log group

Choose an existing log group

Logging role Info
Choose the IAM role that will be used to deliver events to your CloudWatch logs

☐ Create a new role ☐ Choose an existing role

Info Logging role is only required when selecting a workflow in the Managed workflows section below.

Jika Anda memilih grup log yang ada, Anda harus memilih salah satu yang terkait dengan Anda Akun AWS.

Transfer Family > Servers > Create server

Step 1
Choose protocols

Step 2
Choose an identity provider

Step 3
Choose an endpoint

Step 4
Choose a domain

Step 5
Configure additional details

Step 6
Review and create

Configure additional details

Logging Info

Log group Info
Choose the CloudWatch log group where your events will be delivered in a structured JSON format

☐ Create a new log group ☒ Choose an existing log group

/aws/transfer/

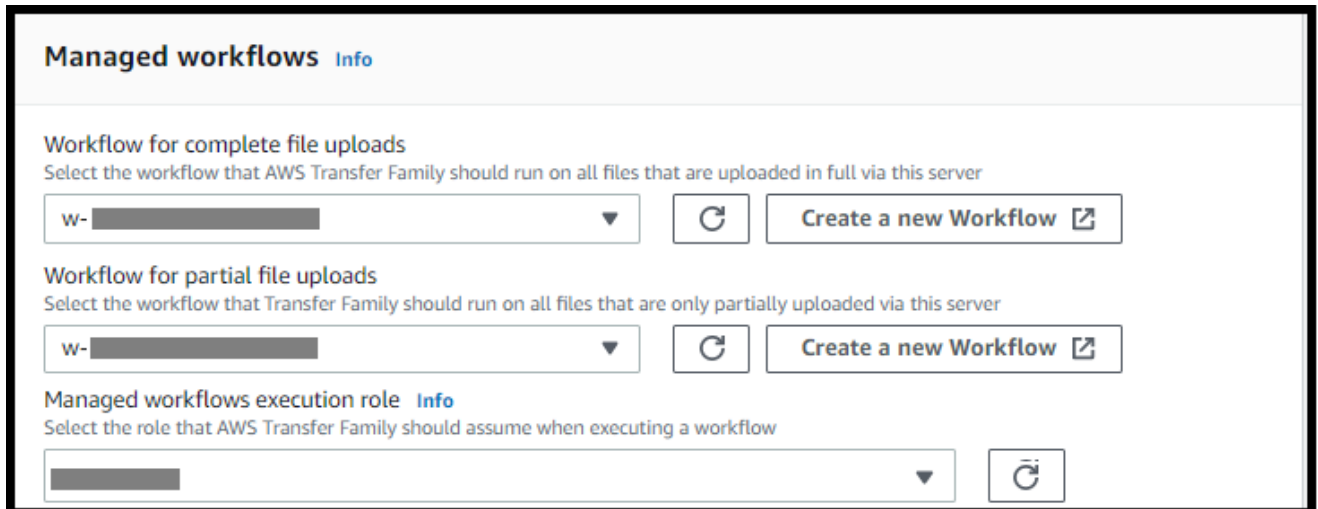
Logging role Info
Choose the IAM role that will be used to deliver events to your CloudWatch logs

☐ Create a new role ☐ Choose an existing role

Info Logging role is only required when selecting a workflow in the Managed workflows section below.

Jika Anda memilih Buat grup log, CloudWatch konsol (<https://console.aws.amazon.com/cloudwatch/>) terbuka ke halaman Buat grup log. Untuk detailnya, lihat [Membuat grup log di CloudWatch Log](#).

- b. (Opsional) Untuk alur kerja Terkelola, pilih alur kerja IDs (dan peran terkait) yang harus diasumsikan oleh Transfer Family saat menjalankan alur kerja. Anda dapat memilih satu alur kerja untuk dieksekusi setelah unggahan lengkap, dan satu lagi untuk mengeksekusi pada unggahan sebagian. Untuk mempelajari lebih lanjut tentang memproses file Anda menggunakan alur kerja terkelola, lihat [AWS Transfer Family alur kerja terkelola](#).



Managed workflows [Info](#)

Workflow for complete file uploads
Select the workflow that AWS Transfer Family should run on all files that are uploaded in full via this server

W- [dropdown] [refresh] [Create a new Workflow](#)

Workflow for partial file uploads
Select the workflow that Transfer Family should run on all files that are only partially uploaded via this server

W- [dropdown] [refresh] [Create a new Workflow](#)

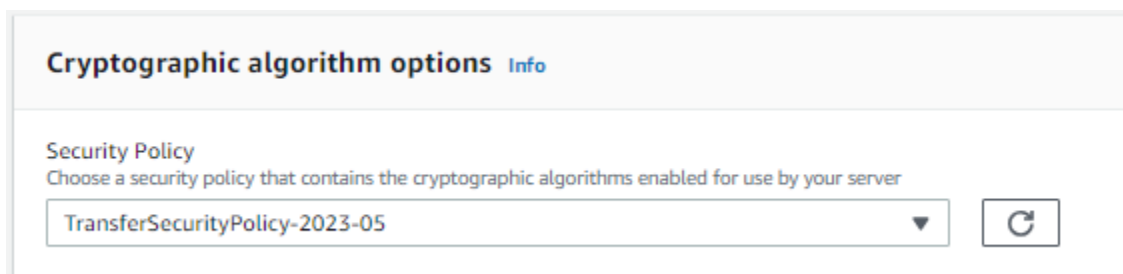
Managed workflows execution role [Info](#)
Select the role that AWS Transfer Family should assume when executing a workflow

[dropdown] [refresh]

- c. Untuk opsi algoritma kriptografi, pilih kebijakan keamanan yang berisi algoritma kriptografi yang diaktifkan untuk digunakan oleh server Anda.

Note

Transfer Family menetapkan kebijakan keamanan terbaru ke FTP server Anda. Namun, karena FTP protokol tidak menggunakan enkripsi apa pun, FTP server tidak menggunakan algoritma kebijakan keamanan apa pun. Kecuali server Anda juga menggunakan SFTP protokol FTPS atau, kebijakan keamanan tetap tidak digunakan.



Cryptographic algorithm options [Info](#)

Security Policy
Choose a security policy that contains the cryptographic algorithms enabled for use by your server

TransferSecurityPolicy-2023-05 [dropdown] [refresh]

- d. Untuk Server Host Key, biarkan kosong.

Note

Bagian Server Host Key hanya digunakan untuk memigrasikan pengguna dari server SFTP berkemampuan yang ada.

Server Host Key [Info](#)**Private key - optional**

Upload an RSA, ECDSA, or ED25519 private key that will be used to identify your SFTP server when clients connect to it. Additional keys can be added once the server is created.

Enter an optional RSA, ECDSA, or ED25519 key

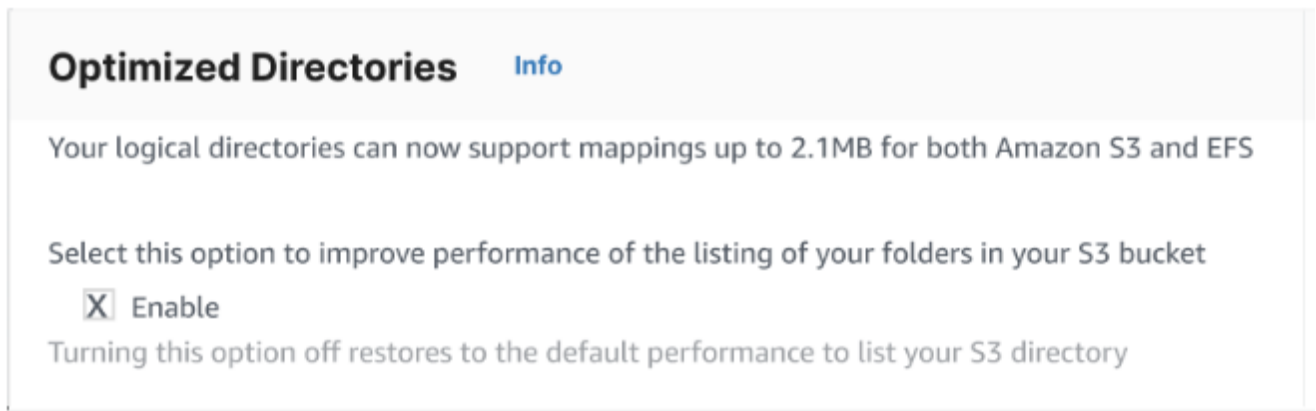
Description - optional

Add a description to differentiate between multiple private keys

Enter optional description

i You can ignore this section unless you are migrating users from an existing SFTP server.

- e. (Opsional) Untuk Tag, untuk Kunci dan Nilai, masukkan satu atau beberapa tag sebagai pasangan nilai kunci, lalu pilih Tambahkan tag.
- f. Anda dapat mengoptimalkan kinerja untuk direktori Amazon S3 Anda. Misalnya, Anda masuk ke direktori home Anda, dan Anda memiliki 10.000 subdirektori. Dengan kata lain, bucket S3 Anda memiliki 10.000 folder. Dalam skenario ini, jika Anda menjalankan perintah `ls` (daftar), operasi daftar memakan waktu antara enam dan delapan menit. Namun, jika Anda mengoptimalkan direktori Anda, operasi ini hanya membutuhkan beberapa detik.



Optimized Directories [Info](#)

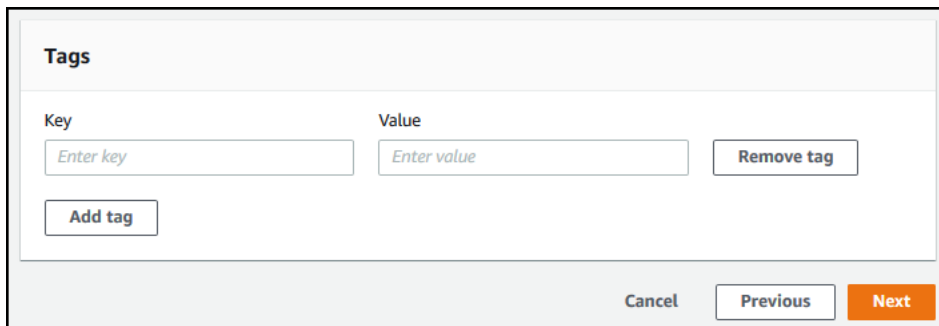
Your logical directories can now support mappings up to 2.1MB for both Amazon S3 and EFS

Select this option to improve performance of the listing of your folders in your S3 bucket

☒ Enable

Turning this option off restores to the default performance to list your S3 directory

g. Pilih Berikutnya.



Tags

Key	Value	
Enter key	Enter value	Remove tag

h. (Opsional) Anda dapat mengonfigurasi AWS Transfer Family server untuk menampilkan pesan yang disesuaikan seperti kebijakan organisasi atau syarat dan ketentuan kepada pengguna akhir Anda. Anda juga dapat menampilkan Message of The Day (MOTD) yang disesuaikan kepada pengguna yang telah berhasil diautentikasi.

Untuk spanduk Tampilan, di kotak teks spanduk tampilan Pra-otentikasi, masukkan pesan teks yang ingin ditampilkan kepada pengguna Anda sebelum mereka mengautentikasi, dan di kotak teks spanduk tampilan pasca-otentikasi, masukkan teks yang ingin ditampilkan kepada pengguna Anda setelah mereka berhasil mengautentikasi.

Display banners

Pre-authentication display banner - optional
Enter the message you want displayed before the client connects to your server

Optional display banner message

Message size cannot exceed 4096 characters. This message will be visible to all users who connect to your server.

Post-authentication display banner - optional
Enter the message you want displayed after the client has connected to your server

Optional display banner message

Message size cannot exceed 4096 characters. This message will be visible to all users who connect to your server.

i SFTP clients will only be able to see the pre-authentication message. FTPS and FTP clients will be able to see both pre-authentication and post-authentication messages.

- i. (Opsional) Anda dapat mengonfigurasi opsi tambahan berikut.
- SetStat opsi: aktifkan opsi ini untuk mengabaikan kesalahan yang dihasilkan saat klien mencoba menggunakan SETSTAT pada file yang Anda unggah ke bucket Amazon S3. Untuk detail tambahan, lihat SetStatOption dokumentasi dalam [ProtocolDetails](#) topik.
 - TLSDimulainya kembali sesi: menyediakan mekanisme untuk melanjutkan atau berbagi kunci rahasia yang dinegosiasikan antara kontrol dan koneksi data untuk suatu sesi. FTPS Untuk detail tambahan, lihat TlsSessionResumptionMode dokumentasi dalam [ProtocolDetails](#) topik.
 - IP pasif: menunjukkan mode pasif, untuk FTP dan FTPS protokol. Masukkan satu IPv4 alamat, seperti alamat IP publik firewall, router, atau penyeimbang beban. Untuk detail tambahan, lihat PassiveIp dokumentasi dalam [ProtocolDetails](#) topik.

Additional configuration

SetStat option - optional [Info](#)
Select whether you want this server to ignore SetStat command

☐ Enable

TLS session resumption - optional [Info](#)
Choose how you want your server to process TLS session resumption requests

☒ Enforce
☐ Enable
☐ Disable

Passive IP - optional [Info](#)
Provide passive IP (PASV) that file transfer clients can use to connect this server

1.2.3.4

8. Di Tinjau dan buat, tinjau pilihan Anda.

- Jika Anda ingin mengedit salah satu dari mereka, pilih Edit di sebelah langkah.

Note

Anda harus meninjau setiap langkah setelah langkah yang Anda pilih untuk diedit.

- Jika Anda tidak memiliki perubahan, pilih Buat server untuk membuat server Anda. Anda dibawa ke halaman Server, ditampilkan berikut, di mana server baru Anda terdaftar.

Diperlukan beberapa menit sebelum status server baru Anda berubah menjadi Online. Pada saat itu, server Anda dapat melakukan operasi file untuk pengguna Anda.

Servers (1)									Actions ▾	Add user	Create server
☐	Hostname	Server ID	State	users	Endpoint type	Domain					
☐	-	s-	 Starting	 No Users	Public	Amazon S3					

Langkah selanjutnya — Untuk langkah selanjutnya, lanjutkan [Bekerja dengan penyedia identitas khusus](#) untuk mengatur pengguna.

Buat server di cloud pribadi virtual

Anda dapat meng-host endpoint server Anda di dalam virtual private cloud (VPC) untuk digunakan untuk mentransfer data ke dan dari bucket Amazon S3 atau sistem file EFS Amazon tanpa melalui internet publik.

Note

Setelah 19 Mei 2021, Anda tidak akan dapat membuat server menggunakan `EndpointType=VPC_ENDPOINT` di AWS akun Anda jika akun Anda belum melakukannya sebelum 19 Mei 2021. Jika Anda telah membuat server dengan `EndpointType=VPC_ENDPOINT` di AWS akun Anda pada atau sebelum 21 Februari 2021, Anda tidak akan terpengaruh. Setelah tanggal ini, gunakan `EndpointType=VPC`. Untuk informasi selengkapnya, lihat [the section called “Menghentikan penggunaan _ VPC_ENDPOINT”](#).

Jika Anda menggunakan Amazon Virtual Private Cloud (AmazonVPC) untuk meng-host AWS sumber daya Anda, Anda dapat membuat koneksi pribadi antara Anda VPC dan server. Anda kemudian dapat menggunakan server ini untuk mentransfer data melalui klien Anda ke dan dari bucket Amazon S3 Anda tanpa menggunakan alamat IP publik atau memerlukan gateway internet.

Menggunakan AmazonVPC, Anda dapat meluncurkan AWS sumber daya di jaringan virtual khusus. Anda dapat menggunakan a VPC untuk mengontrol pengaturan jaringan Anda, seperti rentang alamat IP, subnet, tabel rute, dan gateway jaringan. Untuk informasi selengkapnyaVPCs, lihat [Apa itu AmazonVPC?](#) di Panduan VPC Pengguna Amazon.

Di bagian berikutnya, temukan petunjuk tentang cara membuat dan menghubungkan Anda VPC ke server. Sebagai ikhtisar, Anda melakukan ini sebagai berikut:

1. Siapkan server menggunakan VPC titik akhir.
2. Connect ke server Anda menggunakan klien yang ada di dalam Anda VPC melalui VPC endpoint. Melakukan hal ini memungkinkan Anda untuk mentransfer data yang disimpan di bucket Amazon S3 Anda melalui klien Anda menggunakan. AWS Transfer Family Anda dapat melakukan transfer ini meskipun jaringan terputus dari internet publik.
3. Selain itu, jika Anda memilih untuk membuat titik akhir server Anda menghadap ke internet, Anda dapat mengaitkan alamat IP Elastic dengan titik akhir Anda. Melakukan hal ini memungkinkan klien

di luar VPC koneksi Anda ke server Anda. Anda dapat menggunakan grup VPC keamanan untuk mengontrol akses ke pengguna yang diautentikasi yang permintaannya hanya berasal dari alamat yang diizinkan.

Topik

- [Buat endpoint server yang hanya dapat diakses dalam VPC](#)
- [Buat titik akhir yang menghadap ke internet untuk server Anda](#)
- [Ubah tipe endpoint untuk server Anda](#)
- [Menghentikan penggunaan _ VPC ENDPOINT](#)
- [Memperbarui tipe endpoint AWS Transfer Family server dari VPC _ ENDPOINT ke VPC](#)

Buat endpoint server yang hanya dapat diakses dalam VPC

Dalam prosedur berikut, Anda membuat endpoint server yang hanya dapat diakses oleh sumber daya di dalam AndaVPC.

Untuk membuat endpoint server di dalam VPC

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Dari panel navigasi, pilih Server, lalu pilih Buat server.
3. Di Pilih protokol, pilih satu atau beberapa protokol, lalu pilih Berikutnya. Untuk informasi lebih lanjut tentang protokol, lihat [Langkah 2: Buat server SFTP yang diaktifkan](#)
4. Di Pilih penyedia identitas, pilih Layanan yang dikelola untuk menyimpan identitas dan kunci pengguna AWS Transfer Family, lalu pilih Berikutnya.

Note

Prosedur ini menggunakan opsi yang dikelola layanan. Jika Anda memilih Kustom, Anda memberikan titik akhir Amazon API Gateway dan peran AWS Identity and Access Management (IAM) untuk mengakses titik akhir. Dengan demikian, Anda dapat mengintegrasikan layanan direktori Anda untuk mengautentikasi dan mengotorisasi pengguna Anda. Untuk mempelajari selengkapnya tentang bekerja dengan penyedia identitas kustom, lihat [Bekerja dengan penyedia identitas khusus](#).

5. Di Pilih titik akhir, lakukan hal berikut:

Note

FTP dan FTPS server untuk Transfer Family beroperasi melalui Port 21 (Control Channel) dan Port Range 8192-8200 (Saluran Data).

- a. Untuk tipe Endpoint, pilih tipe endpoint yang VPC dihosting untuk meng-host endpoint server Anda.
- b. Untuk Access, pilih Internal untuk membuat endpoint Anda hanya dapat diakses oleh klien menggunakan alamat IP pribadi endpoint.

Note

Untuk detail tentang opsi Menghadapi Internet, lihat [Buat titik akhir yang menghadap ke internet untuk server Anda](#). Server yang dibuat hanya VPC untuk akses internal tidak mendukung nama host khusus.

- c. Untuk VPC, pilih VPC ID yang ada atau pilih Buat VPC untuk membuat yang baru VPC.
- d. Di bagian Availability Zones, pilih hingga tiga Availability Zones dan subnet terkait.
- e. Di bagian Grup Keamanan, pilih ID grup keamanan yang ada IDs atau pilih Buat grup keamanan untuk membuat grup keamanan baru. Untuk informasi selengkapnya tentang grup keamanan, lihat [Grup keamanan untuk Anda VPC](#) di Panduan Pengguna Amazon Virtual Private Cloud. Untuk membuat grup keamanan, lihat [Membuat grup keamanan](#) di Panduan Pengguna Amazon Virtual Private Cloud.

Note

Anda VPC secara otomatis dilengkapi dengan grup keamanan default. Jika Anda tidak menentukan grup atau grup keamanan yang berbeda saat meluncurkan server, kami mengaitkan grup keamanan default dengan server Anda.

Untuk aturan masuk untuk grup keamanan, Anda dapat mengonfigurasi SSH lalu lintas untuk menggunakan port 22, 2222, atau keduanya. Port 22 dikonfigurasi secara default. Untuk menggunakan port 2222, Anda menambahkan aturan masuk ke grup keamanan Anda. Untuk jenisnya, pilih Kustom TCP, lalu masukkan **2222** untuk rentang Port, dan untuk

sumbernya, masukkan CIDR rentang yang sama dengan yang Anda miliki untuk aturan SSH port 22 Anda.

The screenshot shows the 'Edit inbound rules' interface in the AWS Management Console. It displays a table of inbound rules for a specific security group. The fourth rule is highlighted with a red rectangle. The rules are as follows:

Security group rule ID	Type	Protocol	Port range	Source
sgr-...	HTTP	TCP	80	Custom
sgr-...	RDP	TCP	3389	Custom
sgr-...	HTTPS	TCP	443	Custom
sgr-...	Custom TCP	TCP	2222	Custom
sgr-...	SSH	TCP	22	Custom

- f. (Optional) Untuk FIPSDiaktifkan, pilih kotak centang FIPSEnabled endpoint untuk memastikan titik akhir sesuai dengan Standar Pemrosesan Informasi Federal (). FIPS

Note

FIPSEndpoint yang diaktifkan hanya tersedia di Wilayah Amerika AWS Utara. Untuk Wilayah yang tersedia, lihat [AWS Transfer Family titik akhir dan kuota](#) di Referensi Umum AWS Untuk informasi selengkapnya FIPS, lihat [Federal Information Processing Standard \(FIPS\) 140-2](#).

- g. Pilih Berikutnya.
6. Di Konfigurasi detail tambahan, lakukan hal berikut:
- Untuk CloudWatch logging, pilih salah satu dari berikut ini untuk mengaktifkan CloudWatch pencatatan Amazon dari aktivitas pengguna Anda:
 - Buat peran baru untuk memungkinkan Transfer Family membuat IAM peran secara otomatis, selama Anda memiliki izin yang tepat untuk membuat peran baru. IAMPeran yang diciptakan disebut `AWSTransferLoggingAccess`.
 - Pilih peran yang ada untuk memilih IAM peran yang ada dari akun Anda. Di bawah Peran logging, pilih peran. IAMPeran ini harus menyertakan kebijakan kepercayaan dengan Layanan yang disetel ke `transfer.amazonaws.com`.

Untuk informasi selengkapnya tentang CloudWatch pencatatan, lihat [Konfigurasi peran CloudWatch logging](#).

 Note

- Anda tidak dapat melihat aktivitas pengguna akhir CloudWatch jika Anda tidak menentukan peran logging.
- Jika Anda tidak ingin menyiapkan peran CloudWatch logging, pilih Pilih peran yang ada, tetapi jangan pilih peran logging.

- b. Untuk opsi algoritma kriptografi, pilih kebijakan keamanan yang berisi algoritma kriptografi yang diaktifkan untuk digunakan oleh server Anda.

 Note

Secara default, kebijakan `TransferSecurityPolicy-2020-06` keamanan dilampirkan ke server Anda kecuali Anda memilih yang berbeda.

Untuk informasi selengkapnya tentang kebijakan keamanan, lihat [Kebijakan keamanan untuk AWS Transfer Family](#).

- c. (Opsional) Untuk Server Host Key RSAED25519, masukkan,, atau kunci ECDSA pribadi yang akan digunakan untuk mengidentifikasi server Anda ketika klien terhubung dengannyaSFTP.

 Note

Bagian ini hanya untuk memigrasi pengguna dari server SFTP berkemampuan - yang ada.

- d. (Opsional) Untuk Tag, untuk Kunci dan Nilai, masukkan satu atau beberapa tag sebagai pasangan nilai kunci, lalu pilih Tambahkan tag.
- e. Pilih Berikutnya.

7. Di Tinjau dan buat, tinjau pilihan Anda. Jika Anda:

- Ingin mengedit salah satu dari mereka, pilih Edit di sebelah langkah.

 Note

Anda perlu meninjau setiap langkah setelah langkah yang Anda pilih untuk diedit.

- Tidak ada perubahan, pilih Buat server untuk membuat server Anda. Anda dibawa ke halaman Server, ditampilkan berikut, di mana server baru Anda terdaftar.

Diperlukan beberapa menit sebelum status server baru Anda berubah menjadi Online. Pada saat itu, server Anda dapat melakukan operasi file untuk pengguna Anda.

Buat titik akhir yang menghadap ke internet untuk server Anda

Dalam prosedur berikut, Anda membuat endpoint server. Titik akhir ini dapat diakses melalui internet hanya untuk klien yang alamat IP sumbernya diizinkan di grup keamanan default AndaVPC. Selain itu, dengan menggunakan alamat IP Elastic untuk membuat titik akhir Anda menghadap ke internet, klien Anda dapat menggunakan alamat IP Elastic untuk memungkinkan akses ke titik akhir Anda di firewall mereka.

 Note

Hanya SFTP dan FTPS dapat digunakan pada titik akhir VPC host yang menghadap ke internet.

Untuk membuat titik akhir yang menghadap ke internet

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Dari panel navigasi, pilih Server, lalu pilih Buat server.
3. Di Pilih protokol, pilih satu atau beberapa protokol, lalu pilih Berikutnya. Untuk informasi lebih lanjut tentang protokol, lihat. [Langkah 2: Buat server SFTP yang diaktifkan](#)
4. Di Pilih penyedia identitas, pilih Layanan yang dikelola untuk menyimpan identitas dan kunci pengguna AWS Transfer Family, lalu pilih Berikutnya.

 Note

Prosedur ini menggunakan opsi yang dikelola layanan. Jika Anda memilih Kustom, Anda memberikan titik akhir Amazon API Gateway dan peran AWS Identity and Access Management (IAM) untuk mengakses titik akhir. Dengan demikian, Anda dapat mengintegrasikan layanan direktori Anda untuk mengautentikasi dan mengotorisasi pengguna Anda. Untuk mempelajari selengkapnya tentang bekerja dengan penyedia identitas kustom, lihat [Bekerja dengan penyedia identitas khusus](#).

5. Di Pilih titik akhir, lakukan hal berikut:

- a. Untuk tipe Endpoint, pilih tipe endpoint yang VPCdihosting untuk meng-host endpoint server Anda.
- b. Untuk Akses, pilih Internet Facing untuk membuat titik akhir Anda dapat diakses oleh klien melalui internet.

 Note

Ketika Anda memilih Internet Facing, Anda dapat memilih alamat IP Elastis yang ada di setiap subnet atau subnet. Atau Anda dapat pergi ke VPC konsol (<https://console.aws.amazon.com/vpc/>) untuk mengalokasikan satu atau lebih alamat IP elastis baru. Alamat ini dapat dimiliki oleh AWS atau oleh Anda. Anda tidak dapat mengaitkan alamat IP Elastis yang sudah digunakan dengan titik akhir Anda.

- c. (Opsional) Untuk nama host Kustom, pilih salah satu dari berikut ini:

 Note

Pelanggan AWS GovCloud (US) perlu terhubung melalui alamat IP Elastic secara langsung, atau membuat catatan nama host dalam Commercial Route 53 yang mengarah ke merekaEIP. Untuk informasi selengkapnya tentang menggunakan Route 53 untuk GovCloud titik akhir, lihat [Menyiapkan Amazon Route 53 dengan AWS GovCloud \(US\) sumber daya Anda](#) di Panduan AWS GovCloud (US) Pengguna.

- Amazon Route 53 DNS alias — jika nama host yang ingin Anda gunakan terdaftar di Route 53. Anda kemudian dapat memasukkan nama host.
- Lainnya DNS — jika nama host yang ingin Anda gunakan terdaftar dengan DNS penyedia lain. Anda kemudian dapat memasukkan nama host.
- Tidak ada - untuk menggunakan titik akhir server dan tidak menggunakan nama host khusus. Nama host server mengambil formulir `server-id.server.transfer.region.amazonaws.com`.

 Note

Untuk pelanggan di AWS GovCloud (US), memilih None tidak membuat nama host dalam format ini.

Untuk mempelajari lebih lanjut tentang bekerja dengan nama host kustom, lihat [Bekerja dengan nama host khusus](#).

- d. Untuk VPC, pilih VPC ID yang ada atau pilih Buat VPC untuk membuat yang baru VPC.
- e. Di bagian Availability Zones, pilih hingga tiga Availability Zones dan subnet terkait. Untuk IPv4 Alamat, pilih alamat IP Elastis untuk setiap subnet. Ini adalah alamat IP yang dapat digunakan klien Anda untuk memungkinkan akses ke titik akhir Anda di firewall mereka.
- f. Di bagian Grup Keamanan, pilih ID grup keamanan yang ada IDs atau atau pilih Buat grup keamanan untuk membuat grup keamanan baru. Untuk informasi selengkapnya tentang grup keamanan, lihat [Grup keamanan untuk Anda VPC](#) di Panduan Pengguna Amazon Virtual Private Cloud. Untuk membuat grup keamanan, lihat [Membuat grup keamanan](#) di Panduan Pengguna Amazon Virtual Private Cloud.

 Note

Anda VPC secara otomatis dilengkapi dengan grup keamanan default. Jika Anda tidak menentukan grup atau grup keamanan yang berbeda saat meluncurkan server, kami mengaitkan grup keamanan default dengan server Anda.

Untuk aturan masuk untuk grup keamanan, Anda dapat mengonfigurasi SSH lalu lintas untuk menggunakan port 22, 2222, atau keduanya. Port 22 dikonfigurasi secara default.

Untuk menggunakan port 2222, Anda menambahkan aturan masuk ke grup keamanan Anda. Untuk jenisnya, pilih Kustom TCP, lalu masukkan **2222** untuk rentang Port, dan untuk sumbernya, masukkan CIDR rentang yang sama dengan yang Anda miliki untuk aturan SSH port 22 Anda.

The screenshot shows the 'Edit inbound rules' interface in the AWS IAM console. It displays a table of inbound rules for a specific security group. The rules are as follows:

Security group rule ID	Type	Protocol	Port range	Source
sg-...	HTTP	TCP	80	Custom (0.0.0.0/0)
sg-...	RDP	TCP	3389	Custom (0.0.0.0/0)
sg-...	HTTPS	TCP	443	Custom (0.0.0.0/0)
sg-...	Custom TCP	TCP	2222	Custom (72.21.196.64/32)
sg-...	SSH	TCP	22	Custom (72.21.196.64/32)

An 'Add rule' button is located at the bottom left of the table.

- g. (Opsional) Untuk FIPSDiaktifkan, pilih kotak centang FIPSEnabled endpoint untuk memastikan titik akhir sesuai dengan Standar Pemrosesan Informasi Federal (). FIPS

Note

FIPSEndpoint yang diaktifkan hanya tersedia di Wilayah Amerika AWS Utara. Untuk Wilayah yang tersedia, lihat [AWS Transfer Family titik akhir dan kuota](#) di Referensi Umum AWS Untuk informasi selengkapnya FIPS, lihat [Federal Information Processing Standard \(FIPS\) 140-2](#).

- h. Pilih Berikutnya.
6. Di Konfigurasi detail tambahan, lakukan hal berikut:
- Untuk CloudWatch logging, pilih salah satu dari berikut ini untuk mengaktifkan CloudWatch pencatatan Amazon dari aktivitas pengguna Anda:
 - Buat peran baru untuk memungkinkan Transfer Family membuat IAM peran secara otomatis, selama Anda memiliki izin yang tepat untuk membuat peran baru. IAM Peran yang diciptakan disebut `AWSTransferLoggingAccess`.

- Pilih peran yang ada untuk memilih IAM peran yang ada dari akun Anda. Di bawah Peran logging, pilih peran. IAMPeran ini harus menyertakan kebijakan kepercayaan dengan Layanan yang disetel ke `transfer.amazonaws.com`.

Untuk informasi selengkapnya tentang CloudWatch pencatatan, lihat [Konfigurasi peran CloudWatch logging](#).

 Note

- Anda tidak dapat melihat aktivitas pengguna akhir CloudWatch jika Anda tidak menentukan peran logging.
- Jika Anda tidak ingin menyiapkan peran CloudWatch logging, pilih Pilih peran yang ada, tetapi jangan pilih peran logging.

- b. Untuk opsi algoritma kriptografi, pilih kebijakan keamanan yang berisi algoritma kriptografi yang diaktifkan untuk digunakan oleh server Anda.

 Note

Secara default, kebijakan `TransferSecurityPolicy-2020-06` keamanan dilampirkan ke server Anda kecuali Anda memilih yang berbeda.

Untuk informasi selengkapnya tentang kebijakan keamanan, lihat [Kebijakan keamanan untuk AWS Transfer Family](#).

- c. (Opsional) Untuk Server Host Key RSAED25519, masukkan,, atau kunci ECDSA pribadi yang akan digunakan untuk mengidentifikasi server Anda ketika klien terhubung dengannyaSFTP.

 Note

Bagian ini hanya untuk memigrasi pengguna dari server SFTP berkemampuan - yang ada.

- d. (Opsional) Untuk Tag, untuk Kunci dan Nilai, masukkan satu atau beberapa tag sebagai pasangan nilai kunci, lalu pilih Tambahkan tag.

- e. Pilih Berikutnya.
- f. (Opsional) Untuk alur kerja Terkelola, pilih alur kerja IDs (dan peran terkait) yang harus diasumsikan oleh Transfer Family saat menjalankan alur kerja. Anda dapat memilih satu alur kerja untuk dieksekusi setelah unggahan lengkap, dan satu lagi untuk mengeksekusi pada unggahan sebagian. Untuk mempelajari lebih lanjut tentang memproses file Anda menggunakan alur kerja terkelola, lihat [AWS Transfer Family alur kerja terkelola](#).

Managed workflows [Info](#)

Workflow for complete file uploads
Select the workflow that AWS Transfer Family should run on all files that are uploaded in full via this server

w- [dropdown] [refresh] [Create a new Workflow](#)

Workflow for partial file uploads
Select the workflow that Transfer Family should run on all files that are only partially uploaded via this server

w- [dropdown] [refresh] [Create a new Workflow](#)

Managed workflows execution role [Info](#)
Select the role that AWS Transfer Family should assume when executing a workflow

[dropdown] [refresh]

7. Di Tinjau dan buat, tinjau pilihan Anda. Jika Anda:

- Ingin mengedit salah satu dari mereka, pilih Edit di sebelah langkah.

Note

Anda perlu meninjau setiap langkah setelah langkah yang Anda pilih untuk diedit.

- Tidak ada perubahan, pilih Buat server untuk membuat server Anda. Anda dibawa ke halaman Server, ditampilkan berikut, di mana server baru Anda terdaftar.

Anda dapat memilih ID server untuk melihat pengaturan rinci dari server yang baru saja Anda buat. Setelah kolom IPv4Alamat publik diisi, alamat IP Elastis yang Anda berikan berhasil dikaitkan dengan titik akhir server Anda.

 Note

Ketika server Anda di online, hanya subnet yang dapat dimodifikasi dan hanya melalui VPC [UpdateServer](#) API Anda harus [menghentikan server](#) untuk menambah atau mengubah alamat IP Elastis titik akhir server.

Ubah tipe endpoint untuk server Anda

Jika Anda memiliki server yang sudah ada yang dapat diakses melalui internet (yaitu, memiliki tipe titik akhir publik), Anda dapat mengubah titik akhir ke titik akhir VPC

 Note

Jika Anda memiliki server yang ada di VPC display asVPC_ENDPOINT, kami sarankan Anda memodifikasinya ke tipe VPC endpoint baru. Dengan tipe endpoint baru ini, Anda tidak perlu lagi menggunakan Network Load Balancer NLB () untuk mengaitkan alamat IP Elastis dengan endpoint server Anda. Selain itu, Anda dapat menggunakan grup VPC keamanan untuk membatasi akses ke titik akhir server Anda. Namun, Anda dapat terus menggunakan tipe VPC_ENDPOINT endpoint sesuai kebutuhan.

Prosedur berikut mengasumsikan bahwa Anda memiliki server yang menggunakan tipe endpoint publik saat ini atau tipe yang lebih lamaVPC_ENDPOINT.

Untuk mengubah tipe endpoint untuk server Anda

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Di panel navigasi, pilih Server.
3. Pilih kotak centang server yang ingin Anda ubah tipe endpoint.

 Important

Anda harus menghentikan server sebelum Anda dapat mengubah titik akhirnya.

4. Untuk Tindakan, pilih Berhenti.
5. Di kotak dialog konfirmasi yang muncul, pilih Berhenti untuk mengonfirmasi bahwa Anda ingin menghentikan server.

 Note

Sebelum melanjutkan ke langkah berikutnya, di detail Endpoint, tunggu Status server berubah menjadi Offline; ini bisa memakan waktu beberapa menit. Anda mungkin harus memilih Refresh di halaman Server untuk melihat perubahan status. Anda tidak akan dapat melakukan pengeditan apa pun sampai server Offline.

6. Di detail Endpoint, pilih Edit.
7. Dalam konfigurasi Edit titik akhir, lakukan hal berikut:
 - a. Untuk jenis Edit titik akhir, pilih yang VPCdihosting.
 - b. Untuk Access, pilih salah satu dari berikut ini:
 - Internal untuk membuat titik akhir Anda hanya dapat diakses oleh klien menggunakan alamat IP pribadi titik akhir.
 - Menghadapi Internet untuk membuat titik akhir Anda dapat diakses oleh klien melalui internet publik.

 Note

Ketika Anda memilih Internet Facing, Anda dapat memilih alamat IP Elastis yang ada di setiap subnet atau subnet. Atau, Anda dapat pergi ke VPC konsol (<https://console.aws.amazon.com/vpc/>) untuk mengalokasikan satu atau lebih alamat IP Elastis baru. Alamat ini dapat dimiliki oleh AWS atau oleh Anda. Anda tidak dapat mengaitkan alamat IP Elastis yang sudah digunakan dengan titik akhir Anda.

- c. (Opsional untuk akses yang menghadap internet saja) Untuk nama host Kustom, pilih salah satu dari berikut ini:
 - Amazon Route 53 DNS alias — jika nama host yang ingin Anda gunakan terdaftar di Route 53. Anda kemudian dapat memasukkan nama host.
 - Lainnya DNS — jika nama host yang ingin Anda gunakan terdaftar dengan DNS penyedia lain. Anda kemudian dapat memasukkan nama host.
 - Tidak ada - untuk menggunakan titik akhir server dan tidak menggunakan nama host khusus. Nama host server mengambil formulir `serverId.server.transfer.regionId.amazonaws.com`.

Untuk mempelajari lebih lanjut tentang bekerja dengan nama host kustom, lihat [Bekerja dengan nama host khusus](#).

- d. Untuk VPC, pilih VPC ID yang ada, atau pilih Buat VPC untuk membuat yang baru VPC.
- e. Di bagian Availability Zones, pilih hingga tiga Availability Zones dan subnet terkait. Jika Internet Facing dipilih, pilih juga alamat IP Elastis untuk setiap subnet.

 Note

Jika Anda menginginkan maksimum tiga Availability Zone, tetapi tidak cukup tersedia, buat di VPC console (<https://console.aws.amazon.com/vpc/>).

Jika Anda memodifikasi subnet atau alamat IP Elastis, server membutuhkan beberapa menit untuk memperbarui. Anda tidak dapat menyimpan perubahan Anda sampai pembaruan server selesai.

- f. Pilih Simpan.
8. Untuk Tindakan, pilih Mulai dan tunggu status server berubah menjadi Online; ini bisa memakan waktu beberapa menit.

 Note

Jika Anda mengubah tipe titik akhir publik menjadi tipe VPC endpoint, perhatikan bahwa tipe Endpoint untuk server Anda telah berubah menjadi VPC

Grup keamanan default dilampirkan ke titik akhir. Untuk mengubah atau menambahkan grup keamanan tambahan, lihat [Membuat Grup Keamanan](#).

Menghentikan penggunaan _ VPC ENDPOINT

AWS Transfer Family menghentikan kemampuan untuk membuat server dengan EndpointType=VPC_ENDPOINT untuk AWS akun baru. Per 19 Mei 2021, AWS akun yang tidak memiliki AWS Transfer Family server dengan tipe titik akhir tidak VPC_ENDPOINT akan dapat membuat server baru. EndpointType=VPC_ENDPOINT Jika Anda sudah memiliki server yang menggunakan tipe VPC_ENDPOINT endpoint, kami sarankan Anda mulai menggunakan EndpointType=VPC sesegera mungkin. Untuk detailnya, lihat [Memperbarui jenis titik akhir AWS Transfer Family server Anda dari VPC _ ENDPOINT ke VPC](#).

Kami meluncurkan tipe VPC endpoint baru di awal tahun 2020. Untuk informasi selengkapnya, lihat [AWS Transfer Family untuk SFTP mendukung Grup VPC Keamanan dan alamat IP Elastis](#). Titik akhir baru ini lebih kaya fitur dan hemat biaya dan tidak ada PrivateLink biaya. Untuk informasi lebih lanjut, lihat [AWS PrivateLink harga](#).

Tipe endpoint ini secara fungsional setara dengan tipe endpoint sebelumnya (). VPC_ENDPOINT Anda dapat melampirkan alamat IP Elastis langsung ke titik akhir untuk membuatnya menghadap internet dan menggunakan grup keamanan untuk penyaringan IP sumber. Untuk informasi selengkapnya, lihat [daftar Use IP allow untuk mengamankan posting blog AWS Transfer Family untuk SFTP server Anda](#).

Anda juga dapat meng-host titik akhir ini di VPC lingkungan bersama. Untuk informasi selengkapnya, lihat [AWS Transfer Family sekarang mendukung VPC lingkungan layanan bersama](#).

Selain itu SFTP, Anda dapat menggunakan VPC EndpointType untuk mengaktifkan FTPS dan FTP. Kami tidak berencana untuk menambahkan fitur ini FTPS dan/atau FTP dukungan ke EndpointType=VPC_ENDPOINT. Kami juga telah menghapus jenis titik akhir ini sebagai opsi dari AWS Transfer Family konsol.

Anda dapat mengubah jenis endpoint untuk server Anda menggunakan konsol Transfer Family,, AWS CLI, APISDKs, atau AWS CloudFormation. Untuk mengubah jenis endpoint server Anda, lihat [Memperbarui tipe endpoint AWS Transfer Family server dari VPC _ ENDPOINT ke VPC](#).

Jika Anda memiliki pertanyaan, hubungi AWS Dukungan atau tim AWS akun Anda.

Note

Kami tidak berencana untuk menambahkan fitur ini dan FTPS atau FTP dukungan ke EndpointType = VPC _ENDPOINT. Kami tidak lagi menawarkannya sebagai opsi di AWS Transfer Family Konsol.

Jika Anda memiliki pertanyaan tambahan, Anda dapat menghubungi kami melalui AWS Dukungan atau tim akun Anda.

Memperbarui tipe endpoint AWS Transfer Family server dari VPC _ ENDPOINT ke VPC

Anda dapat menggunakan AWS Management Console, AWS CloudFormation, atau Transfer Family API untuk memperbarui server EndpointType dari VPC_ENDPOINT ke VPC. Prosedur dan contoh

terperinci untuk menggunakan masing-masing metode ini untuk memperbarui jenis endpoint server disediakan di bagian berikut. Jika Anda memiliki server di beberapa AWS wilayah dan di beberapa AWS akun, Anda dapat menggunakan contoh skrip yang disediakan di bagian berikut, dengan modifikasi, untuk mengidentifikasi server menggunakan VPC_ENDPOINT jenis yang perlu Anda perbarui.

Topik

- [Mengidentifikasi server menggunakan tipe VPC_ENDPOINT endpoint](#)
- [Memperbarui tipe endpoint server menggunakan AWS Management Console](#)
- [Memperbarui tipe endpoint server menggunakan AWS CloudFormation](#)
- [Memperbarui server EndpointType menggunakan API](#)

Mengidentifikasi server menggunakan tipe **VPC_ENDPOINT** endpoint

Anda dapat mengidentifikasi server mana yang VPC_ENDPOINT menggunakan AWS Management Console.

Untuk mengidentifikasi server menggunakan tipe **VPC_ENDPOINT** endpoint menggunakan konsol

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Pilih Server di panel navigasi untuk menampilkan daftar server di akun Anda di wilayah tersebut.
3. Urutkan daftar server berdasarkan jenis Endpoint untuk melihat semua server yang menggunakan VPC_ENDPOINT.

Untuk mengidentifikasi server yang menggunakan **VPC_ENDPOINT** berbagai AWS Wilayah dan akun

Jika Anda memiliki server di beberapa AWS wilayah dan di beberapa AWS akun, Anda dapat menggunakan contoh skrip berikut, dengan modifikasi, untuk mengidentifikasi server menggunakan tipe VPC_ENDPOINT endpoint. Contoh skrip menggunakan Amazon EC2 [DescribeRegions](#) dan Transfer Family [ListServers](#) API panggilan untuk mendapatkan daftar server IDs dan wilayah dari semua server Anda menggunakan VPC_ENDPOINT. Jika Anda memiliki banyak AWS akun, Anda dapat melakukan loop melalui akun menggunakan IAM Peran dengan akses auditor hanya baca jika Anda mengautentikasi menggunakan profil sesi ke penyedia identitas Anda.

1. Berikut ini adalah contoh sederhana.

```
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. Setelah Anda memiliki daftar server untuk diperbarui, Anda dapat menggunakan salah satu metode yang dijelaskan di bagian berikut untuk memperbarui EndpointType keVPC.

Memperbarui tipe endpoint server menggunakan AWS Management Console

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Di panel navigasi, pilih Server.
3. Pilih kotak centang server yang ingin Anda ubah tipe endpoint.

Important

Anda harus menghentikan server sebelum Anda dapat mengubah titik akhirnya.

4. Untuk Tindakan, pilih Berhenti.
5. Di kotak dialog konfirmasi yang muncul, pilih Berhenti untuk mengonfirmasi bahwa Anda ingin menghentikan server.

 Note

Sebelum melanjutkan ke langkah berikutnya, tunggu Status server berubah menjadi Offline; ini bisa memakan waktu beberapa menit. Anda mungkin harus memilih Refresh di halaman Server untuk melihat perubahan status.

6. Setelah status berubah menjadi Offline, pilih server untuk menampilkan halaman detail server.
7. Di bagian Detail titik akhir, pilih Edit.
8. Pilih VPCHost untuk tipe Endpoint.
9. Pilih Simpan.
10. Untuk Tindakan, pilih Mulai dan tunggu status server berubah menjadi Online; ini bisa memakan waktu beberapa menit.

Memperbarui tipe endpoint server menggunakan AWS CloudFormation

Bagian ini menjelaskan cara menggunakan AWS CloudFormation untuk memperbarui server EndpointType keVPC. Gunakan prosedur ini untuk server Transfer Family yang telah Anda gunakan AWS CloudFormation. Dalam contoh ini, AWS CloudFormation template asli yang digunakan untuk menyebarkan server Transfer Family ditampilkan sebagai berikut:

```
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

```

Template diperbarui dengan perubahan berikut:

- EndpointType diubah menjadi VPC.
- AWS::EC2::VPCendpoint Sumber daya dihapus.
- Itu SecurityGroupIdSubnetIds,, dan VpcId dipindahkan ke EndpointDetails bagian AWS::Transfer::Server sumber daya,
- VpcEndpointId Properti EndpointDetails telah dihapus.

Template yang diperbarui terlihat sebagai berikut:

```

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
```

Untuk memperbarui jenis endpoint dari server Transfer Family yang digunakan AWS CloudFormation

1. Hentikan server yang ingin Anda perbarui menggunakan langkah-langkah berikut.

- a. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
- b. Di panel navigasi, pilih Server.
- c. Pilih kotak centang server yang ingin Anda ubah tipe endpoint.

 Important

Anda harus menghentikan server sebelum Anda dapat mengubah titik akhirnya.

- d. Untuk Tindakan, pilih Berhenti.
- e. Di kotak dialog konfirmasi yang muncul, pilih Berhenti untuk mengonfirmasi bahwa Anda ingin menghentikan server.

 Note

Sebelum melanjutkan ke langkah berikutnya, tunggu Status server berubah menjadi Offline; ini bisa memakan waktu beberapa menit. Anda mungkin harus memilih Refresh di halaman Server untuk melihat perubahan status.

2. Perbarui CloudFormation tumpukan

- a. Buka AWS CloudFormation konsol di <https://console.aws.amazon.com/cloudformation>.
- b. Pilih tumpukan yang digunakan untuk membuat server Transfer Family.
- c. Pilih Perbarui.
- d. Pilih Ganti template saat ini

- e. Unggah template baru. CloudFormation Change Sets membantu Anda memahami bagaimana perubahan template akan memengaruhi sumber daya yang sedang berjalan sebelum Anda menerapkannya. Dalam contoh ini, sumber daya server Transfer akan dimodifikasi, dan VPC Endpoint sumber daya akan dihapus. Server tipe VPC endpoint membuat VPC Endpoint atas nama Anda, menggantikan sumber daya asli VPC Endpoint.

Setelah mengunggah template baru, set perubahan akan terlihat mirip dengan yang berikut:

Change set preview				
Changes (2)				
Search changes				
Action	Logical ID	Physical ID	Resource type	Replacement
Modify	TransferServer	arn:aws:transfer:us-east-1:364810874344:server/s-6a7d04e12d494ec98	AWS::Transfer::Server	Conditional
Remove	VPCEndpoint	vpce-04e685f8702849573 🔗	AWS::EC2::VPCEndpoint	-

- f. Perbarui tumpukan.
- Setelah pembaruan tumpukan selesai, navigasikan ke konsol manajemen Transfer Family di <https://console.aws.amazon.com/transfer/>.
 - Mulai ulang server. Pilih server yang Anda perbarui AWS CloudFormation, lalu pilih Mulai dari menu Tindakan.

Memperbarui server EndpointType menggunakan API

Anda dapat menggunakan [perintah deskripsi-server](#), atau AWS CLI perintah. [UpdateServer](#) API. Contoh skrip berikut menghentikan server Transfer Family, memperbarui EndpointType, menghapus VPC _ENDPOINT, dan memulai 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={'SecurityGroupIds':
        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])

```

Bekerja dengan nama host khusus

Nama host server Anda adalah nama host yang pengguna Anda masukkan di klien mereka ketika mereka terhubung ke server Anda. Anda dapat menggunakan domain khusus yang telah Anda daftarkan untuk nama host server Anda saat Anda bekerja dengannya AWS Transfer Family. Misalnya, Anda mungkin menggunakan nama host khusus seperti `mysftpserver.mysubdomain.domain.com`.

Untuk mengarahkan lalu lintas dari domain kustom terdaftar ke titik akhir server, Anda dapat menggunakan Amazon Route 53 atau penyedia Sistem Nama Domain (DNS) mana pun. Route 53 adalah DNS layanan yang mendukung AWS Transfer Family secara asli.

Topik

- [Gunakan Amazon Route 53 sebagai DNS penyedia Anda](#)
- [Gunakan DNS penyedia lain](#)
- [Nama host khusus untuk server yang dibuat non-konsol](#)

Di konsol, Anda dapat memilih salah satu opsi ini untuk menyiapkan nama host khusus:

- Amazon Route 53 DNS alias — jika nama host yang ingin Anda gunakan terdaftar di Route 53. Anda kemudian dapat memasukkan nama host.
- Lainnya DNS — jika nama host yang ingin Anda gunakan terdaftar dengan DNS penyedia lain. Anda kemudian dapat memasukkan nama host.
- Tidak ada - untuk menggunakan titik akhir server dan tidak menggunakan nama host khusus.

Anda mengatur opsi ini ketika Anda membuat server baru atau mengedit konfigurasi server yang ada. Untuk informasi selengkapnya tentang membuat server baru, lihat [Langkah 2: Buat server SFTP yang diaktifkan](#). Untuk informasi selengkapnya tentang mengedit konfigurasi server yang ada, lihat [Edit detail server](#).

Untuk detail selengkapnya tentang penggunaan domain Anda sendiri untuk nama host server dan cara AWS Transfer Family menggunakan Route 53, lihat bagian berikut.

Gunakan Amazon Route 53 sebagai DNS penyedia Anda

Saat membuat server, Anda dapat menggunakan Amazon Route 53 sebagai DNS penyedia Anda. Sebelum Anda menggunakan domain dengan Route 53, Anda mendaftarkan domain. Untuk informasi selengkapnya, lihat [Cara kerja pendaftaran Domain](#) di Panduan Pengembang Amazon Route 53.

Saat Anda menggunakan Route 53 untuk menyediakan DNS perutean ke server Anda, AWS Transfer Family gunakan nama host khusus yang Anda masukkan untuk mengekstrak zona hostingnya. Saat AWS Transfer Family mengekstrak zona yang dihosting, tiga hal dapat terjadi:

1. Jika Anda baru mengenal Route 53 dan tidak memiliki zona yang dihosting, AWS Transfer Family tambahkan zona host baru dan CNAME catatan. Nilai CNAME catatan ini adalah nama host endpoint untuk server Anda. A CNAME adalah nama domain alternatif.
2. Jika Anda memiliki zona yang dihosting di Route 53 tanpa CNAME catatan apa pun, AWS Transfer Family tambahkan CNAME catatan ke zona yang dihosting.
3. Jika layanan mendeteksi bahwa CNAME rekaman sudah ada di zona yang dihosting, Anda akan melihat kesalahan yang menunjukkan bahwa CNAME rekaman sudah ada. Dalam hal ini, ubah nilai CNAME catatan ke nama host server Anda.

 Note

Jika langkah ini merupakan bagian dari alur kerja pembuatan server, server Anda berhasil dibuat dan nama host khusus Anda disetel ke Tidak Ada.

Untuk informasi selengkapnya tentang zona yang dihosting di Route 53, lihat [Zona yang dihosting](#) di Panduan Pengembang Amazon Route 53.

Gunakan DNS penyedia lain

Saat membuat server, Anda juga dapat menggunakan DNS penyedia selain Amazon Route 53. Jika Anda menggunakan DNS penyedia alternatif, pastikan lalu lintas dari domain Anda diarahkan ke titik akhir server Anda.

Untuk melakukannya, atur domain Anda ke nama host endpoint untuk server. Nama host endpoint terlihat seperti ini di konsol:

`serverid.server.transfer.region.amazonaws.com`

 Note

Jika server Anda memiliki VPC titik akhir, maka format untuk nama host berbeda dari yang dijelaskan di atas. Untuk menemukan VPC titik akhir Anda, pilih VPC pada halaman detail server, lalu pilih ID VPC titik akhir di dasbor. VPC Endpoint adalah DNS nama depan dari yang terdaftar.

Nama host khusus untuk server yang dibuat non-konsol

Saat Anda membuat server menggunakan AWS Cloud Development Kit (AWS CDK), AWS CloudFormation, atau melalui CLI, Anda harus menambahkan tag jika Anda ingin server tersebut memiliki nama host khusus. Saat Anda membuat server Transfer Family menggunakan konsol, penandaan dilakukan secara otomatis.

Note

Anda juga perlu membuat DNS catatan untuk mengarahkan lalu lintas dari domain Anda ke titik akhir server Anda. Untuk detailnya, lihat [Bekerja dengan catatan](#) di Panduan Pengembang Amazon Route 53.

Gunakan kunci berikut untuk nama host kustom Anda:

- Tambahkan `transfer:customHostname` untuk menampilkan nama host khusus di konsol.
- Jika Anda menggunakan Route 53 sebagai DNS penyedia Anda, tambahkan `transfer:route53HostedZoneId`. Tag ini menautkan nama host kustom ke Route 53 Hosted Zone ID Anda.

Untuk menambahkan nama host khusus, keluarkan CLI perintah berikut.

```
aws transfer tag-resource --arn arn:aws:transfer:region:Akun AWS:server/server-ID --tags Key=transfer:customHostname,Value="custom-host-name"
```

Sebagai contoh:

```
aws transfer tag-resource --arn arn:aws:transfer:us-east-1:111122223333:server/s-1234567890abcdef0 --tags Key=transfer:customHostname,Value="abc.example.com"
```

Jika Anda menggunakan Route 53, keluarkan perintah berikut untuk menautkan nama host kustom Anda ke Route 53 Hosted Zone ID.

```
aws transfer tag-resource --arn server-ARN:server/server-ID --tags Key=transfer:route53HostedZoneId,Value=HOSTED-ZONE-ID
```

Sebagai contoh:


```
aws transfer tag-resource --arn arn:aws:transfer:us-east-1:111122223333:server/s-1234567890abcdef0 --tags Key=transfer:route53HostedZoneId,Value=ABCDE1111222233334444
```

Dengan asumsi nilai sampel dari perintah sebelumnya, jalankan perintah berikut untuk melihat tag Anda:

```
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

Zona publik Anda yang dihosting dan mereka IDs tersedia di Amazon Route 53. Masuk ke AWS Management Console dan buka konsol Route 53 di <https://console.aws.amazon.com/route53/>.

Mentransfer file melalui titik akhir server menggunakan klien

Anda mentransfer file melalui AWS Transfer Family layanan dengan menentukan operasi transfer di klien. AWS Transfer Family mendukung klien berikut:

- Kami mendukung SFTP protokol versi 3.
- Buka SSH (macOS dan Linux)

Note

Klien ini hanya berfungsi dengan server yang diaktifkan untuk Secure Shell (SSH) File Transfer Protocol (SFTP).

- Menang SCP (hanya Microsoft Windows)
- Cyberduck (Windows, macOS, dan Linux)
- FileZilla (Windows, macOS, dan Linux)

Batasan berikut berlaku untuk setiap klien:

- Jumlah maksimum SFTP sesi bersamaan, multipleks, per koneksi adalah 10.
- Amazon S3 dan Amazon EFS (karena NFSv4 protokol) memerlukan nama file dalam pengkodean -8. UTF Menggunakan pengkodean yang berbeda dapat menyebabkan hasil yang tidak terduga. Untuk Amazon S3, lihat Pedoman [penamaan kunci objek](#).
- Untuk File Transfer Protocol over SSL (FTPS), hanya mode Eksplisit yang didukung. Mode implisit tidak didukung.
- Untuk File Transfer Protocol (FTP) danFTPS, hanya mode Pasif yang didukung.
- Untuk FTP danFTPS, hanya STREAM mode yang didukung.
- Untuk FTP danFTPS, hanya mode Gambar/Biner yang didukung.
- Untuk FTP danFTPS, TLS - PROT C (tidak dilindungi) TLS untuk koneksi data adalah default tetapi PROT C tidak didukung dalam AWS Transfer Family FTPS protokol. Jadi untukFTPS, Anda perlu mengeluarkan PROT P agar operasi data Anda diterima.
- Jika Anda menggunakan Amazon S3 untuk penyimpanan server Anda, dan jika klien Anda berisi opsi untuk menggunakan beberapa koneksi untuk satu transfer, pastikan untuk menonaktifkan opsi tersebut. Jika tidak, unggahan file besar dapat gagal dengan cara yang tidak terduga. Perhatikan bahwa jika Anda menggunakan Amazon EFS sebagai backend penyimpanan Anda, EFS mendukung beberapa koneksi untuk satu transfer.

Berikut ini adalah daftar perintah yang tersedia untuk FTP danFTPS:

Perintah yang tersedia					
ABOR	FEAT	MLST	PASS	RETR	STOR

Perintah yang tersedia					
AUTH	LANG	MKD	PASV	RMD	STOU
CDUP	LIST	MODE	PBSZ	RNFR	STRU
CWD	MDTM	NLST	PROT	RNTO	SYST
DELE	MFMT	NOOP	PWD	SIZE	TYPE
EPSV	MLSD	OPTS	QUIT	STAT	USER

 Note

APPE tidak didukung.

Untuk SFTP, operasi berikut saat ini tidak didukung untuk pengguna yang menggunakan direktori home logis di server yang menggunakan Amazon Elastic File System (AmazonEFS).

Perintah yang tidak didukung SFTP			
SSH_FXP_R EADLINK	SSH_FXP_SYMLINK	SSH_FXP_STAT ketika file yang diminta adalah symlink	SSH_FXP_ REALPATH ketika jalur yang diminta berisi komponen symlink apa pun

Hasilkan key pair publik-pribadi

Sebelum Anda dapat mentransfer file, Anda harus memiliki key pair publik-pribadi yang tersedia. Jika sebelumnya Anda belum membuat key pair, lihat [Buat SSH kunci untuk pengguna yang dikelola layanan](#).

Topik

- [SFTP/FTPS/FTP Perintah yang Tersedia](#)
- [Temukan titik VPC akhir Amazon Anda](#)

- [Hindari setstat kesalahan](#)
- [Gunakan Buka SSH](#)
- [Gunakan Win SCP](#)
- [Gunakan Cyberduck](#)
- [Gunakan FileZilla](#)
- [Gunakan klien Perl](#)
- [Pemrosesan unggahan pasca](#)

SFTP/FTPS/FTPPerintah yang Tersedia

Tabel berikut menjelaskan perintah yang tersedia untuk AWS Transfer Family, untuk SFTP, FTPS, dan FTP protokol.

Note

Tabel menyebutkan file dan direktori untuk Amazon S3, yang hanya mendukung bucket dan objek: tidak ada hierarki. Namun, Anda dapat menggunakan awalan dalam nama kunci objek untuk menyiratkan hierarki dan mengatur data Anda dengan cara yang mirip dengan folder. Perilaku ini dijelaskan dalam [Bekerja dengan metadata objek](#) di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

SFTP/FTPS/FTPPerintah

Perintah	Amazon S3	Amazon EFS
cd	Didukung	Didukung
chgrp	Tidak Support	Didukung (root atau owner hanya)
chmod	Tidak didukung	Didukung (root hanya)
chmtime	Tidak didukung	Didukung
chown	Tidak Support	Didukung (root hanya)

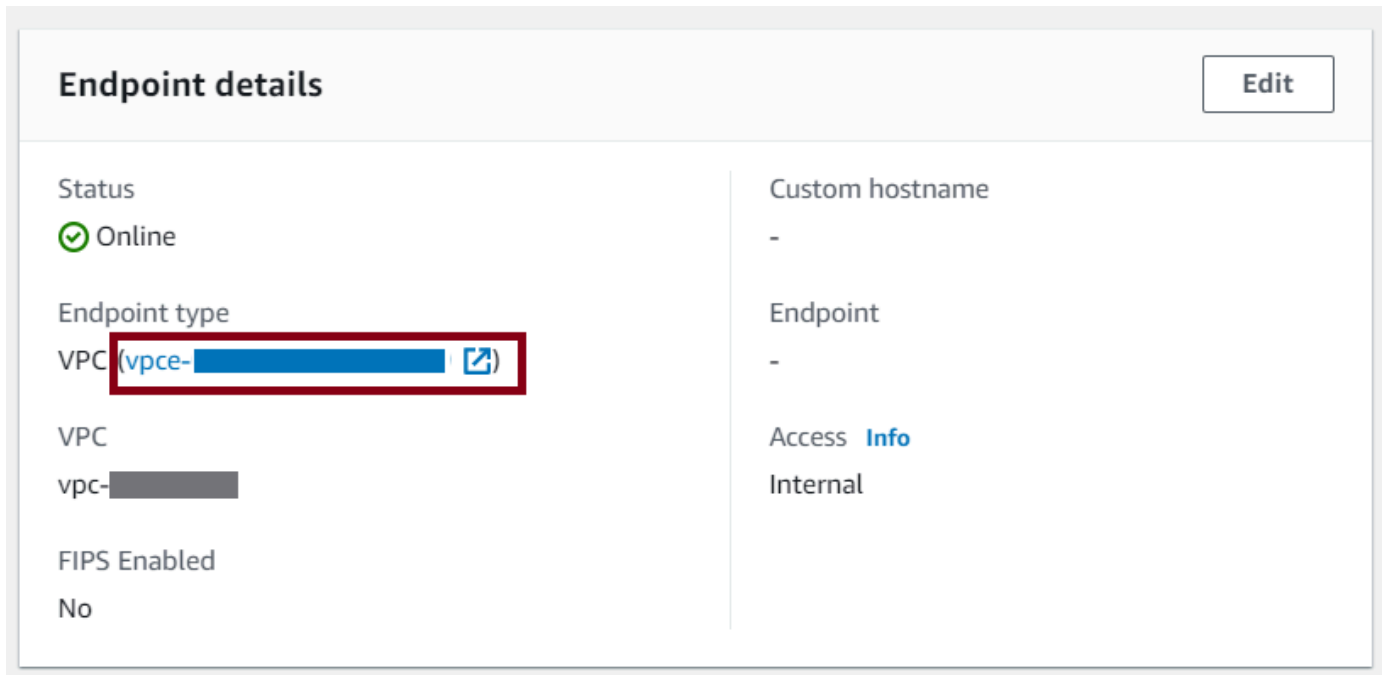
Perintah	Amazon S3	Amazon EFS
<code>get</code>	Didukung	Didukung (termasuk menyelesaikan tautan simbolik)
<code>ln -s</code>	Tidak didukung	Didukung
<code>ls/dir</code>	Didukung	Didukung
<code>mkdir</code>	Didukung	Didukung
<code>put</code>	Didukung	Didukung
<code>pwd</code>	Didukung	Didukung
<code>rename</code>	Didukung hanya untuk file	Didukung  Note Mengganti nama yang akan menimpa file atau direktori yang ada tidak didukung.
<code>rm</code>	Didukung	Didukung
<code>rmdir</code>	Didukung (hanya direktori kosong)	Didukung
<code>version</code>	Didukung	Didukung

Temukan titik VPC akhir Amazon Anda

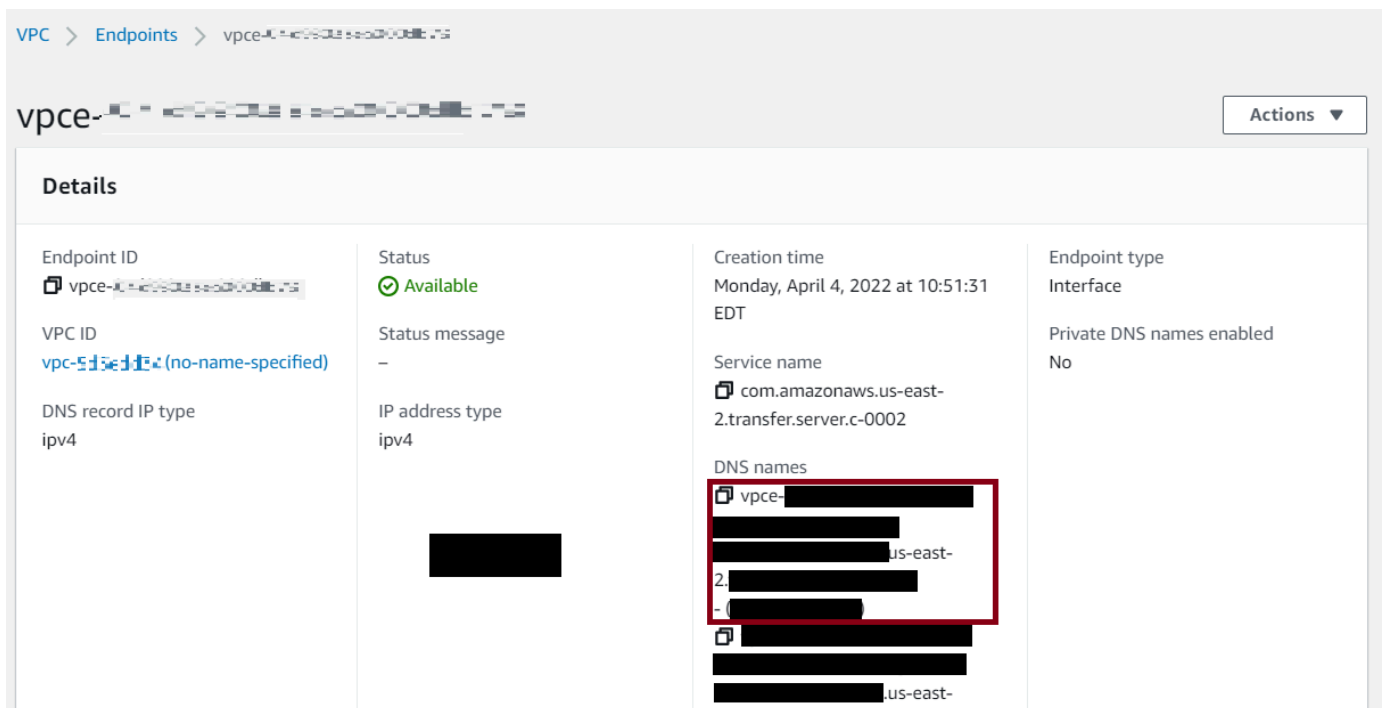
Jika tipe endpoint untuk server Transfer Family Anda VPC, mengidentifikasi titik akhir yang akan digunakan untuk mentransfer file tidaklah mudah. Dalam hal ini, gunakan prosedur berikut untuk menemukan VPC titik akhir Amazon Anda.

Temukan titik VPC akhir Amazon Anda

1. Arahkan ke halaman detail server Anda.
2. Di panel Detail titik akhir, pilih VPC



3. Di VPC dasbor Amazon, pilih ID VPC titik akhir.
4. Dalam daftar DNSNama, titik akhir server Anda adalah yang pertama terdaftar.



Hindari **setstat** kesalahan

Beberapa klien transfer SFTP file dapat mencoba mengubah atribut file jarak jauh, termasuk stempel waktu dan izin, menggunakan perintah, seperti SETSTAT saat mengunggah file. Namun, perintah ini tidak kompatibel dengan sistem penyimpanan objek, seperti Amazon S3. Karena ketidakcocokan ini, unggahan file dari klien ini dapat mengakibatkan kesalahan bahkan ketika file tersebut berhasil diunggah.

- Saat Anda memanggil `CreateServer` or `UpdateServerAPI`, gunakan `ProtocolDetails` opsi `SetStatOption` untuk mengabaikan kesalahan yang dihasilkan saat klien mencoba menggunakan SETSTAT pada file yang Anda unggah ke bucket S3.
- Tetapkan nilai `ENABLE_NO_OP` agar server Transfer Family mengabaikan SETSTAT perintah, dan unggah file tanpa perlu membuat perubahan apa pun pada SFTP klien Anda.
- Perhatikan bahwa meskipun `SetStatOption ENABLE_NO_OP` pengaturan mengabaikan kesalahan, itu menghasilkan entri CloudWatch log di Log, sehingga Anda dapat menentukan kapan klien melakukan SETSTAT panggilan.

Untuk API detail opsi ini, lihat [ProtocolDetails](#).

Gunakan Buka SSH

Gunakan instruksi yang mengikuti untuk mentransfer file dari baris perintah menggunakan BukaSSH.

Note

Klien ini hanya berfungsi dengan SFTP server yang diaktifkan.

Untuk mentransfer file AWS Transfer Family menggunakan utilitas SSH baris perintah Open

1. Di Linux, macOS, atau Windows, buka terminal perintah.
2. Pada prompt, masukkan perintah berikut:

```
sftp -i transfer-key sftp_user@service_endpoint
```

Dalam perintah sebelumnya, *sftp_user* adalah nama pengguna dan *transfer-key* merupakan kunci SSH pribadi. Di sini, *service_endpoint* adalah titik akhir server seperti yang ditunjukkan di AWS Transfer Family konsol untuk server yang dipilih.

Note

Perintah ini menggunakan pengaturan yang ada di `ssh_config` file default. Kecuali Anda sebelumnya telah mengedit file ini, SFTP menggunakan port 22. Anda dapat menentukan port yang berbeda (misalnya 2222) dengan menambahkan **-P** bendera ke perintah, sebagai berikut.

```
sftp -P 2222 -i transfer-key sftp_user@service_endpoint
```

Atau, jika Anda selalu ingin menggunakan port 2222, Anda dapat memperbarui port default Anda di `ssh_config` file Anda.

`sftp` Prompt akan muncul.

3. (Opsional) Untuk melihat direktori home pengguna, masukkan perintah berikut pada `sftp` prompt:

```
pwd
```

4. Untuk mengunggah file dari sistem file Anda ke server Transfer Family, gunakan `put` perintah. Misalnya, untuk mengunggah `hello.txt` (dengan asumsi bahwa file ada di direktori Anda saat ini di sistem file Anda), jalankan perintah berikut pada `sftp` prompt:

```
put hello.txt
```

Pesan yang mirip dengan berikut ini muncul, menunjukkan bahwa transfer file sedang berlangsung, atau selesai.

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

```
hello.txt 100% 127 0.1KB/s 00:00
```

Note

Setelah server Anda dibuat, diperlukan beberapa menit agar nama host endpoint server dapat diselesaikan oleh DNS layanan di lingkungan Anda.

Gunakan Win SCP

Gunakan instruksi yang mengikuti untuk mentransfer file dari baris perintah menggunakan WinSCP.

Note

Jika Anda menggunakan Win SCP 5.19, Anda dapat langsung terhubung ke Amazon S3 menggunakan kredensial AWS Anda dan mengunggah/mengunduh file. Untuk detail selengkapnya, lihat [Menghubungkan ke layanan Amazon S3](#).

Untuk mentransfer file AWS Transfer Family menggunakan Win SCP

1. Buka SCP klien Win.
2. Di kotak dialog Login, untuk protokol File, pilih protokol: SFTP atau FTP.

Jika Anda memilih FTP, untuk Enkripsi, pilih salah satu dari berikut ini:

- Tidak ada enkripsi untuk FTP
 - TLS/SSL Enkripsi eksplisit untuk FTPS
3. Untuk nama Host, masukkan endpoint server Anda. Titik akhir server terletak di halaman detail Server. Untuk informasi selengkapnya, lihat [Lihat SFTP, FTPS, dan detail FTP server](#).

Note

Jika server Anda menggunakan VPC titik akhir, lihat [Temukan titik VPC akhir Amazon Anda](#).

4. Untuk nomor Port, masukkan yang berikut ini:
 - **22** untuk SFTP
 - **21** untuk FTP/FTPS
5. Untuk nama Pengguna, masukkan nama untuk pengguna yang Anda buat untuk penyedia identitas spesifik Anda.

 Note

Nama pengguna harus menjadi salah satu pengguna yang Anda buat atau konfigurasi untuk penyedia identitas Anda. AWS Transfer Family menyediakan penyedia identitas berikut:

- [Bekerja dengan pengguna yang dikelola layanan](#)
- [Menggunakan penyedia identitas AWS Directory Service](#)
- [Bekerja dengan penyedia identitas khusus](#)

6. Pilih Advanced untuk membuka kotak dialog Advanced Site Settings. Di SSHbagian ini, pilih Otentikasi.
7. Untuk file kunci pribadi, telusuri dan pilih file kunci SSH pribadi dari sistem file Anda.

 Note

Jika Win SCP menawarkan untuk mengonversi kunci SSH pribadi Anda ke PPK format, pilih OK.

8. Pilih OK untuk kembali ke kotak dialog Login, lalu pilih Simpan.
9. Dalam kotak dialog Simpan sesi sebagai situs, pilih OK untuk menyelesaikan pengaturan koneksi Anda.
10. Di kotak dialog Login, pilih Tools, lalu pilih Preferences.
11. Di kotak dialog Preferensi, untuk Transfer, pilih Endurance.

Untuk opsi Aktifkan transfer lanjut/transfer ke nama file sementara untuk opsi, pilih Nonaktifkan.

 Note

Jika Anda membiarkan opsi ini diaktifkan, ini meningkatkan biaya unggahan, secara substansional mengurangi kinerja unggahan. Hal ini juga dapat menyebabkan kegagalan upload file besar.

12. Untuk Transfer, pilih Latar Belakang, dan kosongkan kotak centang Gunakan beberapa koneksi untuk transfer tunggal.

 Note

Jika Anda membiarkan opsi ini dipilih, unggahan file besar dapat gagal dengan cara yang tidak terduga. Misalnya, unggahan multipart yang tidak terduga yang dikenakan biaya Amazon S3 dapat dibuat. Korupsi data senyap juga dapat terjadi.

13. Lakukan transfer file Anda.

Anda dapat menggunakan drag-and-drop metode untuk menyalin file antara target dan jendela sumber. Anda dapat menggunakan ikon toolbar untuk mengunggah, mengunduh, menghapus, mengedit, atau memodifikasi properti file di WinSCP.

 Note

Catatan ini tidak berlaku jika Anda menggunakan Amazon EFS untuk penyimpanan. Perintah yang mencoba mengubah atribut file jarak jauh, termasuk stempel waktu, tidak kompatibel dengan sistem penyimpanan objek seperti Amazon S3. Oleh karena itu, jika Anda menggunakan Amazon S3 untuk penyimpanan, pastikan untuk menonaktifkan pengaturan SCP stempel waktu Win (atau gunakan `SetStatOption` seperti yang dijelaskan dalam [Hindari setstat kesalahan](#)) sebelum Anda melakukan transfer file. Untuk melakukannya, di kotak dialog Win SCP Transfer settings, nonaktifkan opsi Setel izin upload dan opsi Preserve timestamp common.

Gunakan Cyberduck

Gunakan instruksi yang mengikuti untuk mentransfer file dari baris perintah menggunakan Cyberduck.

Untuk mentransfer file AWS Transfer Family menggunakan Cyberduck

1. Buka klien [Cyberduck](#).
2. Pilih Buka Koneksi.
3. Di kotak dialog Open Connection, pilih protokol: SFTP(SSH File Transfer Protocol), FTP- SSL (Explicit AUTH TLS), atau FTP(File Transfer Protocol).

4. Untuk Server, masukkan endpoint server Anda. Titik akhir server terletak di halaman detail Server. Untuk informasi selengkapnya, lihat [LihatSFTP,FTPS, dan detail FTP server](#).

 Note

Jika server Anda menggunakan VPC titik akhir, lihat [Temukan titik VPC akhir Amazon Anda](#).

5. Untuk nomor Port, masukkan yang berikut ini:
 - **22** untuk SFTP
 - **21** untuk FTP/FTPS
6. Untuk Nama Pengguna, masukkan nama untuk pengguna yang Anda buat [Mengelola pengguna untuk titik akhir server](#).
7. Jika SFTP dipilih, untuk Kunci SSH Pribadi, pilih atau masukkan kunci SSH pribadi.
8. Pilih Hubungan.
9. Lakukan transfer file Anda.

Tergantung di mana file Anda berada, lakukan salah satu hal berikut:

- Di direktori lokal Anda (sumber), pilih file yang ingin Anda transfer, dan seret dan jatuhkan ke direktori Amazon S3 (target).
- Di direktori Amazon S3 (sumber), pilih file yang ingin Anda transfer, dan seret dan jatuhkan ke direktori lokal Anda (target).

Gunakan FileZilla

Gunakan instruksi yang mengikuti untuk mentransfer file menggunakan FileZilla.

FileZilla Untuk mengatur transfer file

1. Buka FileZilla klien.
2. Pilih File, lalu pilih Site Manager.
3. Di kotak dialog Pengelola Situs, pilih Situs baru.
4. Pada tab Umum, untuk Protokol, pilih protokol: SFTP atau FTP.

Jika Anda memilih FTP, untuk Enkripsi, pilih salah satu dari berikut ini:

- Hanya gunakan biasa FTP (tidak aman) — untuk FTP
 - Gunakan eksplisit FTP TLS jika tersedia — untuk FTPS
5. Untuk nama Host, masukkan protokol yang Anda gunakan, diikuti oleh endpoint server Anda. Titik akhir server terletak di halaman detail Server. Untuk informasi selengkapnya, lihat [LihatSFTP,FTPS, dan detail FTP server](#).

 Note

Jika server Anda menggunakan VPC titik akhir, lihat [Temukan titik VPC akhir Amazon Anda](#).

- Jika Anda menggunakanSFTP, masukkan: `sftp://hostname`
- Jika Anda menggunakanFTPS, masukkan: `ftps://hostname`

Pastikan untuk mengganti *hostname* dengan titik akhir server Anda yang sebenarnya.

6. Untuk nomor Port, masukkan yang berikut ini:
- **22**untuk SFTP
 - **21**untukFTP/FTPS
7. Jika SFTP dipilih, untuk Logon Type, pilih Key file.

Untuk file Kunci, pilih atau masukkan kunci SSH pribadi.

8. Untuk Pengguna, masukkan nama untuk pengguna yang Anda buat[Mengelola pengguna untuk titik akhir server](#).
9. Pilih Hubungkan.
10. Lakukan transfer file Anda.

 Note

Jika Anda mengganggu transfer file yang sedang berlangsung, AWS Transfer Family mungkin menulis sebagian objek di bucket Amazon S3 Anda. Jika Anda mengganggu unggahan, periksa apakah ukuran file di bucket Amazon S3 cocok dengan ukuran file objek sumber sebelum melanjutkan.

Gunakan klien Perl

Jika Anda menggunakan klien `Net::SFTP::Foreign` perl, Anda harus mengatur `queue_size` ke 1. Sebagai contoh:

```
my $sftp = Net::SFTP::Foreign->new('user@s-12345.server.transfer.us-east-2.amazonaws.com', queue_size => 1);
```

Note

[Solusi ini diperlukan untuk revisi sebelum 1.92.02. Net::SFTP::Foreign](#)

Pemrosesan unggahan pasca

Anda dapat melihat informasi pemrosesan unggahan posting termasuk metadata objek Amazon S3 dan pemberitahuan acara.

Topik

- [Metadata objek Amazon S3](#)
- [Pemberitahuan acara Amazon S3](#)

Metadata objek Amazon S3

Sebagai bagian dari metadata objek Anda, Anda melihat kunci yang disebut `x-amz-meta-user-agent` yang nilainya `AWSTransfer` dan `x-amz-meta-user-agent-id` nilainya `username@server-id` `username` ini adalah pengguna Transfer Family yang mengunggah file dan `server-id` merupakan server yang digunakan untuk mengunggah. Informasi ini dapat diakses menggunakan [HeadObject](#) operasi pada objek S3 di dalam fungsi Lambda Anda.

Key	Value
Content-Type	text/plain
x-amz-meta-user-agent	AWSTransfer
x-amz-meta-user-agent-id	sftp_user@s-

Pemberitahuan acara Amazon S3

Saat objek diunggah ke bucket S3 Anda menggunakan Transfer Family, `RoleSessionName` terdapat dalam bidang Peminta dalam struktur notifikasi [peristiwa S3](#) sebagai `[AWS:Role Unique Identifier]/username.sessionid@server-id`. Misalnya, berikut ini adalah konten untuk bidang Peminta sampel dari log akses S3 untuk file yang disalin ke bucket S3.

```
arn:aws:sts::AWS-Account-ID:assumed-role/IamRoleName/
username.sessionid@server-id
```

Di bidang Pemohon di atas, ini menunjukkan IAM Peran yang disebut `IamRoleName`. Untuk informasi selengkapnya tentang mengonfigurasi notifikasi peristiwa S3, lihat [Mengonfigurasi notifikasi peristiwa Amazon S3 di Panduan Pengembang Layanan](#) Penyimpanan Sederhana Amazon. Untuk informasi selengkapnya tentang AWS Identity and Access Management (IAM) pengidentifikasi unik peran, lihat [Pengidentifikasi unik](#) di AWS Identity and Access Management Panduan Pengguna.

Mengelola pengguna untuk titik akhir server

Di bagian berikut, Anda dapat menemukan informasi tentang cara menambahkan pengguna yang menggunakan AWS Transfer Family, AWS Directory Service for Microsoft Active Directory atau penyedia identitas khusus.

Jika Anda menggunakan tipe identitas yang dikelola layanan, Anda menambahkan pengguna ke server yang diaktifkan protokol transfer file Anda. Ketika Anda melakukannya, setiap nama pengguna harus unik di server Anda.

Sebagai bagian dari properti setiap pengguna, Anda juga menyimpan kunci publik Secure Shell (SSH) pengguna tersebut. Melakukan hal itu diperlukan untuk otentikasi berbasis kunci, yang digunakan prosedur ini. Kunci pribadi disimpan secara lokal di komputer pengguna Anda. Ketika pengguna Anda mengirim permintaan otentikasi ke server Anda dengan menggunakan klien, server Anda terlebih dahulu mengonfirmasi bahwa pengguna memiliki akses ke kunci SSH pribadi terkait. Server kemudian berhasil mengotentikasi pengguna.

Selain itu, Anda menentukan direktori home pengguna, atau direktori landing, dan menetapkan peran AWS Identity and Access Management (IAM) ke pengguna. Secara opsional, Anda dapat memberikan kebijakan sesi untuk membatasi akses pengguna hanya ke direktori home bucket Amazon S3 Anda.

Important

AWS Transfer Family memblokir nama pengguna yang panjangnya 1 atau 2 karakter dari otentikasi ke SFTP server. Selain itu, kami juga memblokir nama `root` pengguna. Alasan di balik ini adalah karena volume besar upaya login berbahaya oleh pemindai kata sandi.

Amazon EFS vs Amazon S3

Karakteristik masing-masing opsi penyimpanan:

- Untuk membatasi akses: Amazon S3 mendukung kebijakan sesi; Amazon EFS mendukung POSIX pengguna, grup, dan grup sekunder IDs
- Keduanya mendukung kunci publik/pribadi
- Keduanya mendukung direktori rumah
- Keduanya mendukung direktori logis

Note

Untuk Amazon S3, sebagian besar dukungan untuk direktori logis adalah melalui/. API CLI Anda dapat menggunakan kotak centang Dibatasi di konsol untuk mengunci pengguna ke direktori home mereka, tetapi Anda tidak dapat menentukan struktur direktori virtual.

Direktori logis

Jika Anda menentukan nilai direktori logis untuk pengguna Anda, parameter yang Anda gunakan tergantung pada jenis pengguna.

- Untuk pengguna yang dikelola layanan, berikan nilai direktori logis di `HomeDirectoryMappings`
- Untuk pengguna penyedia identitas kustom, berikan nilai direktori logis di `HomeDirectoryDetails`.

Topik

- [Bekerja dengan pengguna yang dikelola layanan](#)
- [Menggunakan penyedia identitas AWS Directory Service](#)
- [Bekerja dengan penyedia identitas khusus](#)

Bekerja dengan pengguna yang dikelola layanan

Anda dapat menambahkan pengguna Amazon S3 atau Amazon yang EFS dikelola layanan ke server Anda, tergantung pada pengaturan Domain server. Untuk informasi selengkapnya, lihat [Mengkonfigurasi titik SFTP akhir,FTPS, atau FTP server](#).

[Untuk menambahkan pengguna yang dikelola layanan secara terprogram, lihat contoh untuk CreateUserAPI](#)

Note

Untuk pengguna yang dikelola layanan ada batas 2.000 entri direktori logis. Untuk informasi tentang menggunakan direktori logis lihat [Menggunakan direktori logis untuk menyederhanakan struktur direktori Transfer Family](#).

Topik

- [Menambahkan pengguna yang dikelola layanan Amazon S3](#)
- [Menambahkan pengguna yang EFS dikelola layanan Amazon](#)
- [Mengelola pengguna yang dikelola layanan](#)

Menambahkan pengguna yang dikelola layanan Amazon S3

Note

Jika Anda ingin mengonfigurasi bucket Amazon S3 lintas akun, ikuti langkah-langkah yang disebutkan dalam artikel Pusat Pengetahuan ini: [Bagaimana cara mengonfigurasi AWS Transfer Family server saya untuk menggunakan bucket Amazon Simple Storage Service yang ada di akun lain? AWS](#).

Untuk menambahkan pengguna yang dikelola layanan Amazon S3 ke server Anda

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>, lalu pilih Server dari panel navigasi.
2. Pada halaman Server, pilih kotak centang server yang ingin Anda tambahkan pengguna.
3. Pilih Tambahkan pengguna.
4. Di bagian Konfigurasi pengguna, untuk Nama Pengguna, masukkan nama pengguna. Nama pengguna ini harus minimal 3 dan maksimal 100 karakter. Anda dapat menggunakan karakter berikut dalam nama pengguna: a—z, A-Z, 0-9, garis bawah '_', tanda hubung '-', periode '.', dan pada tanda "@". Nama pengguna tidak dapat dimulai dengan tanda hubung '-', titik '.', atau pada tanda "@".
5. Untuk Access, pilih IAM peran yang sebelumnya Anda buat yang menyediakan akses ke bucket Amazon S3 Anda.

Anda membuat IAM peran ini menggunakan prosedur di [Buat IAM peran dan kebijakan](#).

IAM Peran tersebut mencakup IAM kebijakan yang menyediakan akses ke bucket Amazon S3 Anda. Ini juga mencakup hubungan kepercayaan dengan AWS Transfer Family layanan, yang didefinisikan dalam IAM kebijakan lain. Jika Anda memerlukan kontrol akses berbutir halus untuk pengguna Anda, lihat [Tingkatkan kontrol akses data dengan dan posting blog Amazon AWS Transfer Family S3](#).

6. (Opsional) Untuk Kebijakan, pilih salah satu dari berikut ini:
 - Tidak ada
 - Kebijakan yang ada
 - Pilih kebijakan dari IAM: memungkinkan Anda memilih kebijakan sesi yang ada. Pilih Lihat untuk melihat JSON objek yang berisi detail kebijakan.

- Kebijakan buat otomatis berdasarkan folder beranda: menghasilkan kebijakan sesi untuk Anda. Pilih Lihat untuk melihat JSON objek yang berisi detail kebijakan.

 Note

Jika Anda memilih Kebijakan buat otomatis berdasarkan folder beranda, jangan pilih Dibatasi untuk pengguna ini.

Untuk mempelajari lebih lanjut tentang kebijakan sesi, lihat [Buat IAM peran dan kebijakan](#). Untuk mempelajari lebih lanjut tentang membuat kebijakan sesi, lihat [Membuat kebijakan sesi untuk bucket Amazon S3](#).

7. Untuk direktori Home, pilih bucket Amazon S3 untuk menyimpan data yang akan ditransfer. AWS Transfer Family Masukkan jalur ke home direktori tempat pengguna Anda mendarat saat mereka masuk menggunakan klien mereka.

Jika parameter ini kosong, `root` direktori bucket Amazon S3 Anda akan digunakan. Dalam hal ini, pastikan IAM peran Anda menyediakan akses ke `root` direktori ini.

 Note

Sebaiknya pilih jalur direktori yang berisi nama pengguna pengguna, yang memungkinkan Anda menggunakan kebijakan sesi secara efektif. Kebijakan sesi membatasi akses pengguna di bucket Amazon S3 ke direktori pengguna tersebut. `home`

8. (Opsional) Untuk Dibatasi, pilih kotak centang sehingga pengguna Anda tidak dapat mengakses apa pun di luar folder itu dan tidak dapat melihat keranjang Amazon S3 atau nama folder.

 Note

Menugaskan pengguna direktori home dan membatasi pengguna ke direktori home itu harus cukup untuk mengunci akses pengguna ke folder yang ditunjuk. Jika Anda perlu menerapkan kontrol lebih lanjut, gunakan kebijakan sesi.

Jika Anda memilih Dibatasi untuk pengguna ini, Anda tidak dapat memilih Kebijakan buat otomatis berdasarkan folder beranda, karena folder beranda bukan nilai yang ditentukan untuk pengguna Terbatas.

9. Untuk SSHpublic key, masukkan bagian SSH kunci publik dari SSH key pair.

Kunci Anda divalidasi oleh layanan sebelum Anda dapat menambahkan pengguna baru Anda.

 Note

Untuk petunjuk tentang cara membuat SSH key pair, lihat [Buat SSH kunci untuk pengguna yang dikelola layanan](#).

10. (Opsional) Untuk Kunci dan Nilai, masukkan satu atau beberapa tag sebagai pasangan nilai kunci, dan pilih Tambahkan tag.
11. Pilih Tambah untuk menambahkan pengguna baru Anda ke server yang Anda pilih.

Pengguna baru muncul di bagian Pengguna pada halaman detail Server.

Langkah selanjutnya — Untuk langkah selanjutnya, lanjutkan ke [Mentransfer file melalui titik akhir server menggunakan klien](#).

Menambahkan pengguna yang EFS dikelola layanan Amazon

Amazon EFS menggunakan model izin file Antarmuka Sistem Operasi Portabel (POSIX) untuk mewakili kepemilikan file.

- Untuk detail selengkapnya tentang kepemilikan EFS file Amazon, lihat [Kepemilikan EFS file Amazon](#).
- Untuk detail selengkapnya tentang menyiapkan direktori untuk EFS pengguna Anda, lihat [Mengatur EFS pengguna Amazon untuk Transfer Family](#).

Untuk menambahkan pengguna yang EFS dikelola layanan Amazon ke server Anda

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>, lalu pilih Server dari panel navigasi.
2. Pada halaman Server, pilih EFS server Amazon yang ingin Anda tambahkan pengguna.
3. Pilih Tambah pengguna untuk menampilkan halaman Tambah pengguna.
4. Di bagian Konfigurasi pengguna, gunakan pengaturan berikut.
 - a. Nama pengguna, harus minimal 3 dan maksimal 100 karakter. Anda dapat menggunakan karakter berikut dalam nama pengguna: a—z, A-Z, 0-9, garis bawah '_', tanda hubung '-',

periode ' . ', dan pada tanda "@". Nama pengguna tidak dapat dimulai dengan tanda hubung '-', titik ' . ', atau pada tanda "@".

- b. Untuk User ID dan Group ID, perhatikan hal berikut:
 - Untuk pengguna pertama yang Anda buat, kami sarankan Anda memasukkan nilai **0** untuk ID Grup dan ID Pengguna. Ini memberikan hak administrator pengguna untuk Amazon. EFS
 - Untuk pengguna tambahan, masukkan ID POSIX pengguna dan ID grup pengguna. Ini IDs digunakan untuk semua operasi Amazon Elastic File System yang dilakukan oleh pengguna.
 - Untuk ID Pengguna dan ID Grup, jangan gunakan angka nol di depan. Misalnya, dapat **12345** diterima, **012345** tidak.
- c. (Opsional) Untuk Grup Sekunder IDs, masukkan satu atau lebih POSIX grup tambahan IDs untuk setiap pengguna, dipisahkan dengan koma.
- d. Untuk Access, pilih IAM peran yang:
 - Memberikan pengguna akses hanya ke EFS sumber daya Amazon (sistem file) yang Anda ingin mereka akses.
 - Mendefinisikan operasi sistem file mana yang dapat dan tidak dapat dilakukan oleh pengguna.

Kami menyarankan Anda menggunakan IAM peran untuk pemilihan sistem EFS file Amazon dengan akses mount dan izin baca/tulis. Misalnya, kombinasi dari dua kebijakan AWS terkelola berikut, meskipun cukup permisif, memberikan izin yang diperlukan untuk pengguna Anda:

- AmazonElasticFileSystemClientFullAccess
- AWSTransferConsoleFullAccess

Untuk informasi selengkapnya, lihat [AWS Transfer Family dukungan posting blog untuk Amazon Elastic File System](#).

- e. Untuk direktori Home, lakukan hal berikut:
 - Pilih sistem EFS file Amazon yang ingin Anda gunakan untuk menyimpan data yang akan ditransfer AWS Transfer Family.

- Putuskan apakah akan mengatur direktori home ke Restricted. Mengatur direktori home ke Restricted memiliki efek berikut:
- EFS Pengguna Amazon tidak dapat mengakses file atau direktori apa pun di luar folder itu.
- EFS Pengguna Amazon tidak dapat melihat nama sistem EFS file Amazon (fs-xxxxxxx).

 Note

Saat Anda memilih opsi Dibatasi, symlink tidak dapat diselesaikan untuk EFS pengguna Amazon.

- (Opsional) Masukkan jalur ke direktori home yang Anda inginkan agar pengguna masuk saat mereka masuk menggunakan klien mereka.

Jika Anda tidak menentukan direktori home, direktori root sistem EFS file Amazon Anda akan digunakan. Dalam hal ini, pastikan bahwa IAM peran Anda menyediakan akses ke direktori root ini.

5. Untuk SSH public key, masukkan bagian SSH kunci publik dari SSH key pair.

Kunci Anda divalidasi oleh layanan sebelum Anda dapat menambahkan pengguna baru Anda.

 Note

Untuk petunjuk tentang cara membuat SSH key pair, lihat [Buat SSH kunci untuk pengguna yang dikelola layanan](#).

6. (Opsional) Masukkan tag apa pun untuk pengguna. Untuk Kunci dan Nilai, masukkan satu atau beberapa tag sebagai pasangan nilai kunci, dan pilih Tambah tag.
7. Pilih Tambah untuk menambahkan pengguna baru Anda ke server yang Anda pilih.

Pengguna baru muncul di bagian Pengguna pada halaman detail Server.

Masalah yang mungkin Anda temui saat pertama kali SFTP ke server Transfer Family:

- Jika Anda menjalankan `sftp` perintah dan prompt tidak muncul, Anda mungkin menemukan pesan berikut:

```
Couldn't canonicalize: Permission denied
```

Need `cwd`

Dalam hal ini, Anda harus meningkatkan izin kebijakan untuk peran pengguna Anda. Anda dapat menambahkan kebijakan AWS terkelola, seperti `AmazonElasticFileSystemClientFullAccess`.

- Jika Anda memasukkan `pwd` pada `sftp` prompt untuk melihat direktori home pengguna, Anda mungkin melihat pesan berikut, di mana ***USER-HOME-DIRECTORY*** adalah direktori home untuk SFTP pengguna:

```
remote readdir("/USER-HOME-DIRECTORY"): No such file or directory
```

Dalam hal ini, Anda harus dapat menavigasi ke direktori induk (`cd ..`), dan membuat direktori home pengguna (`mkdir username`).

Langkah selanjutnya — Untuk langkah selanjutnya, lanjutkan ke [Mentransfer file melalui titik akhir server menggunakan klien](#).

Mengelola pengguna yang dikelola layanan

Di bagian ini, Anda dapat menemukan informasi tentang cara melihat daftar pengguna, cara mengedit detail pengguna, dan cara menambahkan kunci SSH publik.

- [Melihat daftar pengguna](#)
- [Melihat atau mengedit detail pengguna](#)
- [Hapus pengguna](#)
- [Tambahkan kunci SSH publik](#)
- [Hapus kunci SSH publik](#)

Untuk menemukan daftar pengguna Anda

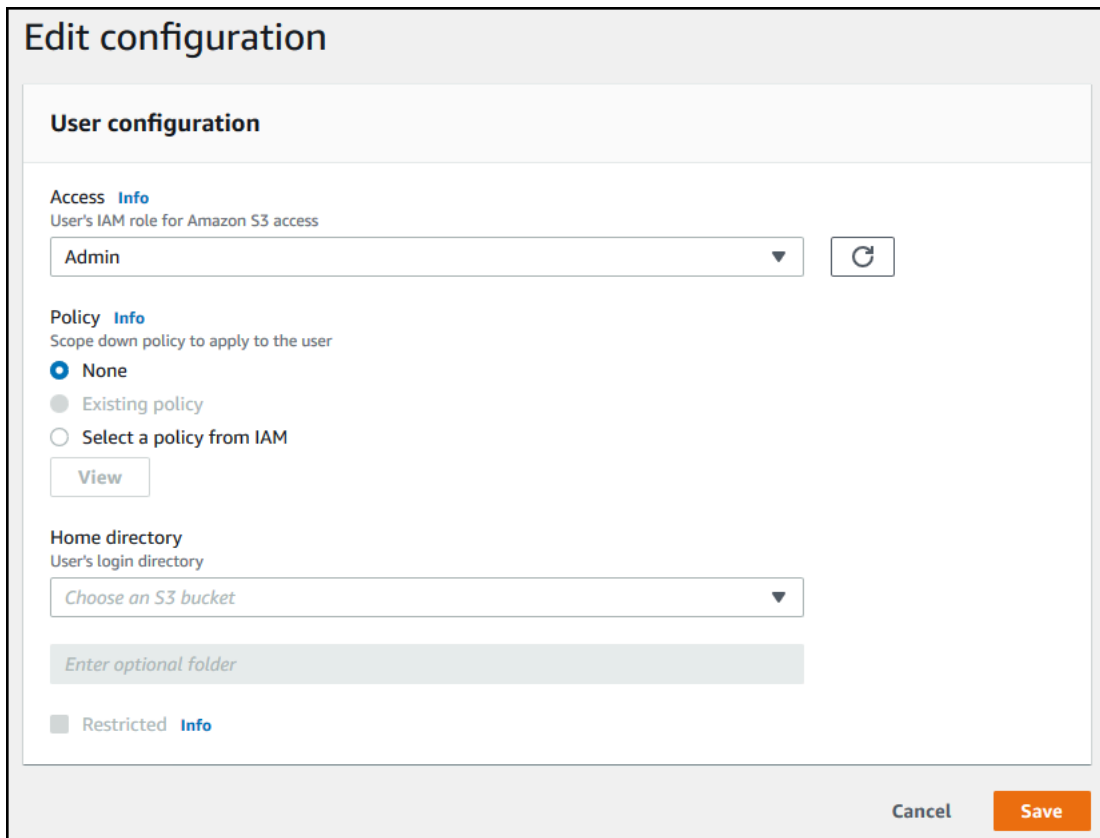
1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Pilih Server dari panel navigasi untuk menampilkan halaman Server.
3. Pilih pengenalan di kolom ID Server untuk melihat halaman Detail Server.
4. Di bawah Pengguna, lihat daftar pengguna.

Untuk melihat atau mengedit detail pengguna

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Pilih Server dari panel navigasi untuk menampilkan halaman Server.
3. Pilih pengenal di kolom ID Server untuk melihat halaman Detail Server.
4. Di bawah Pengguna, pilih nama pengguna untuk melihat halaman Detail pengguna.

Anda dapat mengubah properti pengguna di halaman ini dengan memilih Edit.

5. Pada halaman Detail pengguna, pilih Edit di samping Konfigurasi pengguna.



6. Pada halaman Edit konfigurasi, untuk Access, pilih IAM peran yang sebelumnya Anda buat yang menyediakan akses ke bucket Amazon S3 Anda.

Anda membuat IAM peran ini menggunakan prosedur di [Buat IAM peran dan kebijakan](#).

IAMPeran tersebut mencakup IAM kebijakan yang menyediakan akses ke bucket Amazon S3 Anda. Ini juga mencakup hubungan kepercayaan dengan AWS Transfer Family layanan, yang didefinisikan dalam IAM kebijakan lain.

7. (Opsional) Untuk Kebijakan, pilih salah satu dari berikut ini:
 - Tidak ada

- Kebijakan yang ada
- Pilih kebijakan IAM untuk memilih kebijakan yang ada. Pilih Lihat untuk melihat JSON objek yang berisi detail kebijakan.

Untuk mempelajari lebih lanjut tentang kebijakan sesi, lihat [Buat IAM peran dan kebijakan](#). Untuk mempelajari lebih lanjut tentang membuat kebijakan sesi, lihat [Membuat kebijakan sesi untuk bucket Amazon S3](#).

8. Untuk direktori Home, pilih bucket Amazon S3 untuk menyimpan data yang akan ditransfer. AWS Transfer Family Masukkan jalur ke home direktori tempat pengguna Anda mendarat saat mereka masuk menggunakan klien mereka.

Jika Anda membiarkan parameter ini kosong, root direktori bucket Amazon S3 Anda digunakan. Dalam hal ini, pastikan bahwa IAM peran Anda menyediakan akses ke root direktori ini.

 Note

Sebaiknya pilih jalur direktori yang berisi nama pengguna pengguna, yang memungkinkan Anda menggunakan kebijakan sesi secara efektif. Kebijakan sesi membatasi akses pengguna di bucket Amazon S3 ke direktori pengguna tersebut. home

9. (Opsional) Untuk Dibatasi, pilih kotak centang sehingga pengguna Anda tidak dapat mengakses apa pun di luar folder itu dan tidak dapat melihat keranjang Amazon S3 atau nama folder.

 Note

Saat menugaskan pengguna direktori home dan membatasi pengguna ke direktori home itu, ini harus cukup untuk mengunci akses pengguna ke folder yang ditunjuk. Gunakan kebijakan sesi saat Anda perlu menerapkan kontrol lebih lanjut.

10. Pilih Simpan untuk menyimpan perubahan Anda.

Untuk menghapus klaster

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Pilih Server dari panel navigasi untuk menampilkan halaman Server.

3. Pilih pengenalan di kolom ID Server untuk melihat halaman Detail Server.
4. Di bawah Pengguna, pilih nama pengguna untuk melihat halaman Detail pengguna.
5. Pada halaman Detail pengguna, pilih Hapus di sebelah kanan nama pengguna.
6. Di kotak dialog konfirmasi yang muncul, masukkan kata **delete**, lalu pilih Hapus untuk mengonfirmasi bahwa Anda ingin menghapus pengguna.

Pengguna dihapus dari daftar pengguna.

Untuk menambahkan kunci SSH publik untuk pengguna

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Di panel navigasi, pilih Server.
3. Pilih pengenalan di kolom ID Server untuk melihat halaman Detail Server.
4. Di bawah Pengguna, pilih nama pengguna untuk melihat halaman Detail pengguna.
5. Pilih Tambahkan kunci SSH publik untuk menambahkan kunci SSH publik baru ke pengguna.

 Note

SSH kunci hanya digunakan oleh server yang diaktifkan untuk Secure Shell (SSH) File Transfer Protocol (SFTP). Untuk informasi tentang cara membuat SSH key pair, lihat [Buat SSH kunci untuk pengguna yang dikelola layanan](#).

6. Untuk SSH public key, masukkan bagian kunci SSH publik dari SSH key pair.

Kunci Anda divalidasi oleh layanan sebelum Anda dapat menambahkan pengguna baru Anda. Format SSH kuncinya adalah `ssh-rsa string`. Untuk menghasilkan SSH key pair, lihat [Buat SSH kunci untuk pengguna yang dikelola layanan](#).

7. Pilih Tambah kunci.

Untuk menghapus kunci SSH publik untuk pengguna

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Di panel navigasi, pilih Server.
3. Pilih pengenalan di kolom ID Server untuk melihat halaman Detail Server.
4. Di bawah Pengguna, pilih nama pengguna untuk melihat halaman Detail pengguna.

5. Untuk menghapus kunci publik, pilih kotak centang SSH kuncinya dan pilih Hapus.

Menggunakan penyedia identitas AWS Directory Service

Topik ini menjelaskan cara menggunakan penyedia identitas AWS Directory Service untuk AWS Transfer Family.

Topik

- [Menggunakan AWS Directory Service for Microsoft Active Directory](#)
- [Menggunakan AWS Directory Service untuk Azure Active Directory Domain Services](#)

Menggunakan AWS Directory Service for Microsoft Active Directory

Anda dapat menggunakan AWS Transfer Family untuk mengautentikasi pengguna akhir transfer file Anda menggunakan AWS Directory Service for Microsoft Active Directory. Ini memungkinkan migrasi mulus dari alur kerja transfer file yang mengandalkan otentikasi Active Directory tanpa mengubah kredensi pengguna akhir atau memerlukan otorisasi khusus.

Dengan AWS Managed Microsoft AD, Anda dapat memberikan akses kepada AWS Directory Service pengguna dan grup dengan aman SFTPFTPS, dan FTP untuk data yang disimpan di Amazon Simple Storage Service (Amazon S3) atau Amazon Elastic File System (Amazon EFS). Jika Anda menggunakan Active Directory untuk menyimpan kredensi pengguna Anda, Anda sekarang memiliki cara yang lebih mudah untuk mengaktifkan transfer file untuk pengguna ini.

Anda dapat memberikan akses ke grup Active Directory AWS Managed Microsoft AD di lingkungan lokal atau di AWS Cloud menggunakan konektor Active Directory. Anda dapat memberi pengguna yang sudah dikonfigurasi di lingkungan Microsoft Windows Anda, baik di AWS Cloud atau di jaringan lokal mereka, akses ke AWS Transfer Family server yang menggunakan AWS Managed Microsoft AD identitas.

Note

- AWS Transfer Family tidak mendukung Simple AD.
- Transfer Family tidak mendukung konfigurasi Direktori Aktif lintas wilayah: kami hanya mendukung integrasi Direktori Aktif yang berada di wilayah yang sama dengan server Transfer Family.

- Transfer Family tidak mendukung penggunaan salah satu AWS Managed Microsoft AD atau AD Connector untuk mengaktifkan autentikasi multi-faktor (MFA) untuk infrastruktur RADIUS berbasis MFA Anda yang ada.
- AWS Transfer Family tidak mendukung wilayah yang direplikasi dari Direktori Aktif Terkelola.

Untuk menggunakannya AWS Managed Microsoft AD, Anda harus melakukan langkah-langkah berikut:

1. Buat satu atau lebih AWS Managed Microsoft AD direktori menggunakan AWS Directory Service konsol.
2. Gunakan konsol Transfer Family untuk membuat server yang digunakan AWS Managed Microsoft AD sebagai penyedia identitasnya.
3. Tambahkan akses dari satu atau beberapa AWS Directory Service grup Anda.
4. Meskipun tidak diperlukan, kami menyarankan Anda menguji dan memverifikasi akses pengguna.

Topik

- [Sebelum Anda mulai menggunakan AWS Directory Service for Microsoft Active Directory](#)
- [Bekerja dengan ranah Active Directory](#)
- [Memilih AWS Managed Microsoft AD sebagai penyedia identitas Anda](#)
- [Memberikan akses ke grup](#)
- [Menguji pengguna](#)
- [Menghapus akses server untuk grup](#)
- [Menghubungkan ke server menggunakan SSH \(Secure Shell\)](#)
- [Menghubungkan AWS Transfer Family ke Active Directory yang dikelola sendiri menggunakan hutan dan trust](#)

Sebelum Anda mulai menggunakan AWS Directory Service for Microsoft Active Directory

Menyediakan pengenalan unik untuk grup iklan Anda

Sebelum dapat menggunakan AWS Managed Microsoft AD, Anda harus memberikan pengenalan unik untuk setiap grup di direktori Microsoft AD Anda. Anda dapat menggunakan pengenalan keamanan

(SID) untuk setiap grup untuk melakukan ini. Pengguna grup yang Anda kaitkan memiliki akses ke EFS sumber daya Amazon S3 atau Amazon Anda melalui protokol yang diaktifkan menggunakan Transfer Family. AWS

Gunakan PowerShell perintah Windows berikut untuk mengambil SID untuk grup, menggantikan *YourGroupName* dengan nama grup.

```
Get-ADGroup -Filter {samAccountName -like "YourGroupName*"} -Properties * | Select SamAccountName, ObjectSid
```

Note

Jika Anda menggunakan AWS Directory Service sebagai penyedia identitas Anda, dan jika `userPrincipalName` dan `SamAccountName` memiliki nilai yang berbeda, AWS Transfer Family terima nilainya. `SamAccountName` Transfer Family tidak menerima nilai yang ditentukan dalam `userPrincipalName`.

Tambahkan AWS Directory Service izin ke peran Anda

Anda juga memerlukan AWS Directory Service API izin untuk digunakan AWS Directory Service sebagai penyedia identitas Anda. Izin berikut diperlukan atau disarankan:

- `ds:DescribeDirectories` diperlukan Transfer Family untuk mencari direktori
- `ds:AuthorizeApplication` diperlukan untuk menambahkan otorisasi untuk Transfer Family
- `ds:UnauthorizeApplication` disarankan untuk menghapus sumber daya apa pun yang dibuat untuk sementara, jika terjadi kesalahan selama proses pembuatan server

Tambahkan izin ini ke peran yang Anda gunakan untuk membuat server Transfer Family Anda. Untuk detail selengkapnya tentang izin ini, lihat [AWS Directory Service API izin: Referensi tindakan, sumber daya, dan kondisi](#).

Bekerja dengan ranah Active Directory

Saat Anda mempertimbangkan cara agar pengguna Active Directory mengakses AWS Transfer Family server, ingatlah ranah pengguna, dan ranah grup mereka. Idealnya, ranah pengguna dan ranah grup mereka harus cocok. Artinya, baik pengguna maupun grup berada di ranah default, atau

keduanya berada di ranah tepercaya. Jika tidak demikian, pengguna tidak dapat diautentikasi oleh Transfer Family.

Anda dapat menguji pengguna untuk memastikan konfigurasi sudah benar. Untuk detailnya, lihat [Menguji pengguna](#). Jika ada masalah dengan ranah pengguna/grup, Anda menerima kesalahan, Tidak ada akses terkait yang ditemukan untuk grup pengguna.

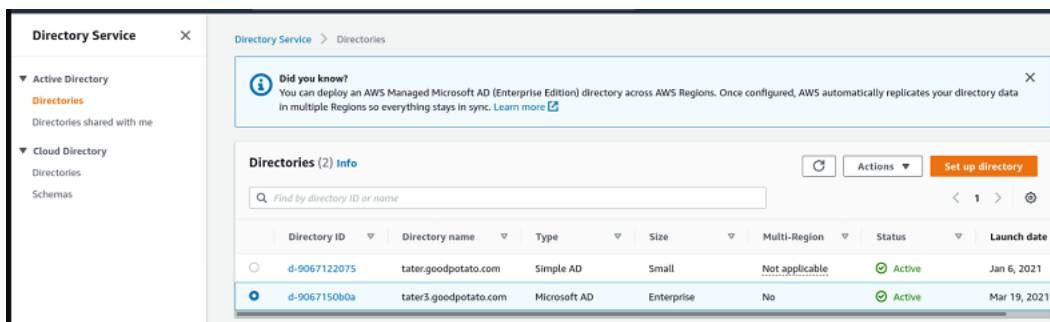
Memilih AWS Managed Microsoft AD sebagai penyedia identitas Anda

Bagian ini menjelaskan cara menggunakan AWS Directory Service for Microsoft Active Directory dengan server.

Untuk digunakan AWS Managed Microsoft AD dengan Transfer Family

1. Masuk ke AWS Management Console dan buka AWS Directory Service konsol di <https://console.aws.amazon.com/directoryservicev2/>.

Gunakan AWS Directory Service konsol untuk mengonfigurasi satu atau beberapa direktori dikelola. Untuk informasi lebih lanjut, lihat [AWS Managed Microsoft AD](#) dalam Panduan Admin AWS Directory Service .



2. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>, dan pilih Buat server.
3. Pada halaman Pilih protokol, pilih satu atau beberapa protokol dari daftar.

Note

Jika Anda memilih FTPS, Anda harus memberikan AWS Certificate Manager sertifikat.

4. Untuk Pilih penyedia identitas, pilih AWS Directory Service.

Choose an identity provider

Identity provider

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

Directory

TATER3 ▼ 

Cancel Previous Next

5. Daftar Direktori berisi semua direktori terkelola yang telah Anda konfigurasi. Pilih direktori dari daftar, dan pilih Berikutnya.

Note

- Direktori Cross-Account dan Shared tidak didukung untuk AWS Managed Microsoft AD
- Untuk menyiapkan server dengan Directory Service sebagai penyedia identitas Anda, Anda perlu menambahkan beberapa AWS Directory Service izin. Untuk detailnya, lihat [Sebelum Anda mulai menggunakan AWS Directory Service for Microsoft Active Directory](#).

6. Untuk menyelesaikan pembuatan server, gunakan salah satu prosedur berikut:

- [Buat server SFTP yang diaktifkan](#)
- [Buat server FTPS yang diaktifkan](#)
- [Buat server FTP yang diaktifkan](#)

Dalam prosedur tersebut, lanjutkan dengan langkah berikut memilih penyedia identitas.

 Important

Anda tidak dapat menghapus direktori Microsoft AD AWS Directory Service jika Anda menggunakannya di server Transfer Family. Anda harus menghapus server terlebih dahulu, dan kemudian Anda dapat menghapus direktori.

Memberikan akses ke grup

Setelah Anda membuat server, Anda harus memilih grup mana di direktori yang harus memiliki akses untuk mengunggah dan mengunduh file melalui protokol yang diaktifkan menggunakan. AWS Transfer Family Anda melakukan ini dengan membuat akses.

 Note

Pengguna harus menjadi bagian langsung dari grup tempat Anda memberikan akses. Misalnya, asumsikan bahwa Bob adalah pengguna dan dia milik GroupA, dan GroupA sendiri termasuk dalam GroupB.

- Jika Anda memberikan akses ke GroupA, Bob diberikan akses.
- Jika Anda memberikan akses ke GroupB (dan bukan ke GroupA), Bob tidak memiliki akses.

Untuk memberikan akses ke grup

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Arahkan ke halaman detail server Anda.
3. Di bagian Accesses, pilih Tambah akses.
4. Masukkan AWS Managed Microsoft AD direktori SID untuk yang ingin Anda akses ke server ini.

 Note

Untuk informasi tentang cara menemukan grup Anda, lihat [the section called “Sebelum Anda mulai menggunakan AWS Directory Service for Microsoft Active Directory”](#). SID

5. Untuk Access, pilih peran AWS Identity and Access Management (IAM) untuk grup.

6. Di bagian Kebijakan, pilih kebijakan. Pengaturan default adalah None.
7. Untuk direktori Home, pilih bucket S3 yang sesuai dengan direktori home grup.

 Note

Anda dapat membatasi bagian bucket yang dilihat pengguna dengan membuat kebijakan sesi. Misalnya, untuk membatasi pengguna ke folder mereka sendiri di bawah /filetest direktori, masukkan teks berikut di dalam kotak.

```
/filetest/${transfer:UserName}
```

Untuk mempelajari selengkapnya tentang membuat kebijakan sesi, lihat [Membuat kebijakan sesi untuk bucket Amazon S3](#).

8. Pilih Tambah untuk membuat asosiasi.
9. Pilih server Anda.
10. Pilih Tambahkan akses.
 - Masukkan SID untuk grup.

 Note

Untuk informasi tentang cara menemukan SID, lihat [the section called “Sebelum Anda mulai menggunakan AWS Directory Service for Microsoft Active Directory”](#).

11. Pilih Tambahkan akses.

Di bagian Accesses, akses untuk server terdaftar.

Endpoint configuration

Availability Zone	Subnet ID	Private IPv4 Address
us-east-1a	subnet-xxxxxx	172.31.80.36

Accesses (1)

Actions Associate access

External Id	Home directory	Role
☒ S...	/padbucket3	ADGuy_S3_And_EFS

Additional details Edit

Logging role [Info](#)
Server activity not logged to Amazon CloudWatch

Server host key [Info](#)

Security Policy [Info](#)
TransferSecurityPolicy-2018-11

Domain
Amazon S3

Menguji pengguna

Anda dapat menguji apakah pengguna memiliki akses ke AWS Managed Microsoft AD direktori untuk server Anda.

Note

Seorang pengguna harus berada dalam satu grup (ID eksternal) yang tercantum di bagian Access pada halaman konfigurasi Endpoint. Jika pengguna tidak berada dalam grup, atau berada di lebih dari satu grup, pengguna tersebut tidak diberikan akses.

Untuk menguji apakah pengguna tertentu memiliki akses

1. Pada halaman detail server, pilih Tindakan, lalu pilih Uji.
2. Untuk pengujian penyedia Identitas, masukkan kredensi masuk untuk pengguna yang berada di salah satu grup yang memiliki akses.
3. Pilih Uji.

Anda melihat tes penyedia identitas yang berhasil, menunjukkan bahwa pengguna yang dipilih telah diberikan akses ke server.

Identity provider testing

User configuration [Info](#)

Username

Password

Response

```

{
  "Response": {
    "homeDirectory": {"/padbucker3"},
    "homeDirectoryDetails": null,
    "homeDirectoryType": "PATH",
    "posixProfile": null,
    "publicKeys": null,
    "role": "arn:aws:iam::195886157073:role/ADGuy_53_Aud_EFS",
    "policy": null,
    "userName": "transferuser1",
    "identityProviderType": null,
    "userConfigMessage": null
  },
  "StatusCode": 200,
  "Message": ""
}

```

Cancel **Test**

Jika pengguna termasuk dalam lebih dari satu grup yang memiliki akses, Anda menerima tanggapan berikut.

```

"Response": "",
"StatusCode": 200,
"Message": "More than one associated access found for user's groups."

```

Menghapus akses server untuk grup

Untuk menghapus akses server untuk grup

1. Pada halaman detail server, pilih Tindakan, lalu pilih Hapus Akses.
2. Di kotak dialog, konfirmasi bahwa Anda ingin menghapus akses untuk grup ini.

Ketika Anda kembali ke halaman detail server, Anda melihat bahwa akses untuk grup ini tidak lagi terdaftar.

Menghubungkan ke server menggunakan SSH (Secure Shell)

Setelah Anda mengkonfigurasi server dan pengguna Anda, Anda dapat terhubung ke server menggunakan SSH dan menggunakan nama pengguna yang sepenuhnya memenuhi syarat untuk pengguna yang memiliki akses.

```
sftp user@active-directory-domain@vpc-endpoint
```

Sebagai contoh:

```
transferuserexample@mycompany.com@vpce-0123456abcdef-789xyz.vpc-  
svc-987654zyxabc.us-east-1.vpce.amazonaws.com.
```

Format ini menargetkan pencarian federasi, membatasi pencarian Direktori Aktif yang berpotensi besar.

Note

Anda dapat menentukan nama pengguna sederhana. Namun, dalam hal ini, kode Active Directory harus mencari semua direktori di federasi. Ini mungkin membatasi pencarian, dan otentikasi mungkin gagal bahkan jika pengguna harus memiliki akses.

Setelah mengautentikasi, pengguna berada di direktori home yang Anda tentukan saat Anda mengonfigurasi pengguna.

Menghubungkan AWS Transfer Family ke Active Directory yang dikelola sendiri menggunakan hutan dan trust

Pengguna di Active Directory (AD) yang dikelola sendiri juga dapat digunakan AWS IAM Identity Center untuk akses masuk tunggal ke dan Transfer Akun AWS Family server. Untuk melakukan itu, AWS Directory Service sediakan opsi berikut:

- Kepercayaan hutan satu arah (keluar dari AWS Managed Microsoft AD dan masuk untuk Active Directory lokal) hanya berfungsi untuk domain root.
- Untuk domain anak, Anda dapat menggunakan salah satu dari berikut ini:
 - Gunakan kepercayaan dua arah antara Active AWS Managed Microsoft AD Directory dan lokal
 - Gunakan kepercayaan eksternal satu arah untuk setiap domain anak.

Saat menghubungkan ke server menggunakan domain tepercaya, pengguna perlu menentukan domain tepercaya, misalnya `transferuserexample@mycompany.com`.

Menggunakan AWS Directory Service untuk Azure Active Directory Domain Services

- Untuk memanfaatkan hutan Active Directory yang ada untuk kebutuhan SFTP Transfer Anda, Anda dapat menggunakan [Active Directory Connector](#).
- Jika Anda menginginkan manfaat Active Directory dan ketersediaan tinggi dalam layanan yang dikelola sepenuhnya, Anda dapat menggunakannya AWS Directory Service for Microsoft Active Directory. Untuk detailnya, lihat [Menggunakan penyedia identitas AWS Directory Service](#).

Topik ini menjelaskan cara menggunakan Active Directory Connector dan [Azure Active Directory Domain Services \(AzureADDS\)](#) untuk mengautentikasi pengguna SFTP Transfer dengan [Azure Active Directory](#).

Topik

- [Sebelum Anda mulai menggunakan AWS Directory Service untuk Azure Active Directory Domain Services](#)
- [Langkah 1: Menambahkan Layanan Domain Direktori Aktif Azure](#)
- [Langkah 2: Membuat akun layanan](#)
- [Langkah 3: Menyiapkan AWS Direktori menggunakan AD Connector](#)
- [Langkah 4: Menyiapkan AWS Transfer Family server](#)
- [Langkah 5: Memberikan akses ke grup](#)
- [Langkah 6: Menguji pengguna](#)

Sebelum Anda mulai menggunakan AWS Directory Service untuk Azure Active Directory Domain Services

Untuk AWS, Anda memerlukan yang berikut ini:

- Virtual private cloud (VPC) di AWS wilayah tempat Anda menggunakan server Transfer Family
- Setidaknya dua subnet pribadi di Anda VPC
- VPC harus memiliki konektivitas internet
- Gateway pelanggan dan gateway pribadi Virtual untuk site-to-site VPN koneksi dengan Microsoft Azure

Untuk Microsoft Azure, Anda memerlukan yang berikut ini:

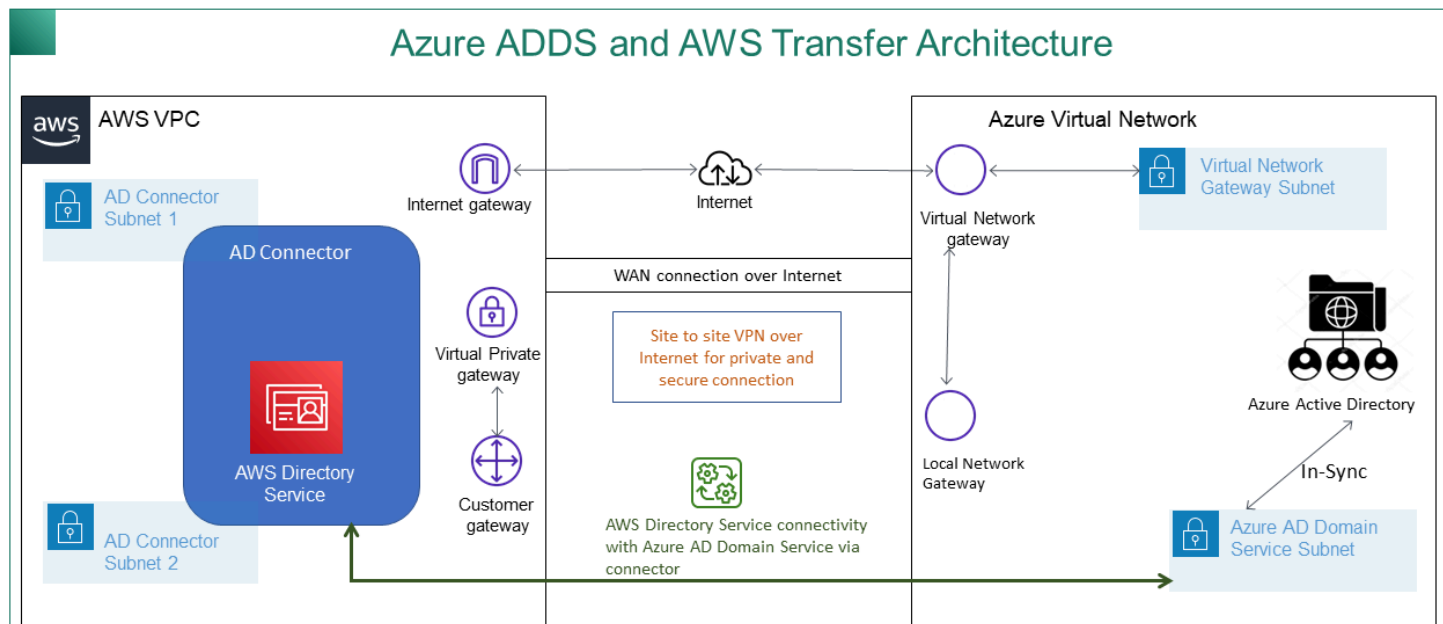
- Layanan domain Direktori Aktif Azure dan Direktori Aktif (ADDS Azure)
- Grup sumber daya Azure
- Jaringan virtual Azure
- VPNkonektivitas antara Amazon Anda VPC dan grup sumber daya Azure Anda

Note

Ini bisa melalui IPSEC terowongan asli atau menggunakan VPN peralatan. Dalam topik ini, kami menggunakan IPSEC terowongan antara gateway jaringan Azure Virtual dan gateway jaringan lokal. Terowongan harus dikonfigurasi untuk memungkinkan lalu lintas antara ADDS titik akhir Azure Anda dan subnet yang menampung Anda. AWS VPC

- Gateway pelanggan dan gateway pribadi Virtual untuk site-to-site VPN koneksi dengan Microsoft Azure

Diagram berikut menunjukkan konfigurasi yang diperlukan sebelum Anda mulai.



Langkah 1: Menambahkan Layanan Domain Direktori Aktif Azure

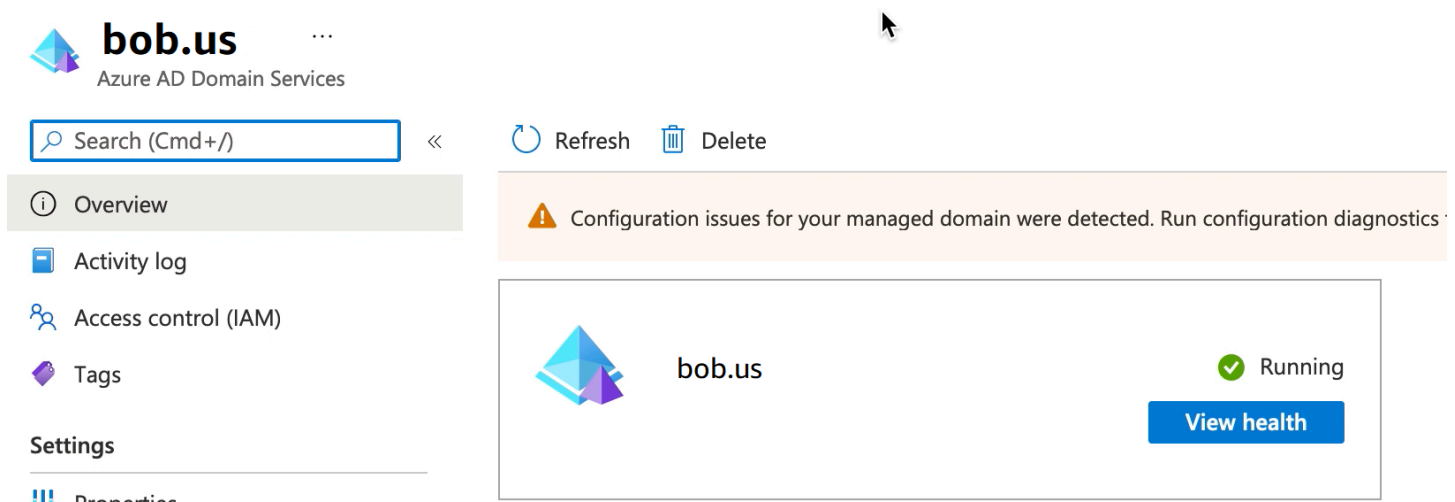
Azure AD tidak mendukung instans penggabungan Domain secara default. Untuk melakukan tindakan seperti Gabung Domain, dan untuk menggunakan alat seperti Kebijakan Grup, administrator harus mengaktifkan Azure Active Directory Domain Services. Jika Anda belum menambahkan Azure

AD DS, atau implementasi yang ada tidak terkait dengan domain yang ingin digunakan oleh server SFTP Transfer, Anda harus menambahkan instance baru.

Untuk informasi tentang mengaktifkan Azure Active Directory Domain Services (AzureADDS), lihat [Tutorial: Membuat dan mengonfigurasi domain terkelola Azure Active Directory Domain Services](#).

Note

Saat Anda mengaktifkan AzureADDS, pastikan itu dikonfigurasi untuk grup sumber daya dan domain Azure AD tempat Anda menghubungkan server SFTP Transfer Anda.



Langkah 2: Membuat akun layanan

Azure AD harus memiliki satu akun layanan yang merupakan bagian dari grup Admin di ADDS Azure. Akun ini digunakan dengan konektor AWS Active Directory. Pastikan akun ini sinkron dengan AzureADDS.

Home > Default Directory > Users > bobatusa

**bobatusa** | Profile ...

User



Reset password



Revoke sessions



Delete



Refresh



Got feedback?

Diagnose and solve problems

Manage

Profile

Assigned roles

Administrative units

Groups

Applications

Licenses

Devices

Azure role assignments

Authentication methods

Activity

Sign-in logs

Audit logs

bobatusa**bobsmith@xyz.com**

User Sign-ins

30

20

10

0

Oct 10 Oct 17 Oct 24 Oct 31

Group memberships

2

Creation time

10/6/2021, 1:32:27 AM

Identity

Name

bobatusa

First name

Bob

Last name

Smith

User Principal Name

bobsmith@xyz.com

User type

Member

Tip

Autentikasi multi-faktor untuk Azure Active Directory tidak didukung untuk server Transfer Family yang menggunakan protokol. SFTP Server Transfer Family tidak dapat memberikan MFA token setelah pengguna mengautentikasi. SFTP Pastikan untuk menonaktifkan MFA sebelum Anda mencoba untuk terhubung.

multi-factor authentication

users service settings

Note: only users licensed to use Microsoft Online Services are eligible for Multi-Factor Authentication. [Learn more about how to license other users.](#)
Before you begin, take a look at the [multi-factor auth deployment guide](#).

View: Sign-in allowed users



Multi-Factor Auth status: Any

bulk update

☐	DISPLAY NAME ^	USER NAME	MULTI-FACTOR AUTH STATUS
☐	Christopher	admin@christopher[REDACTED].com	Disabled
☐	Robert	test@christopher[REDACTED].com	Disabled

Select a user

Langkah 3: Menyiapkan AWS Direktori menggunakan AD Connector

Setelah Anda mengonfigurasi AzureADDS, dan membuat akun layanan dengan IPSEC VPN terowongan antara jaringan Virtual Anda AWS VPC dan Azure, Anda dapat menguji konektivitas dengan melakukan ping ke ADDS DNS alamat IP Azure dari instance apa pun. AWS EC2

Setelah Anda memverifikasi koneksi aktif, Anda dapat melanjutkan di bawah ini.

Untuk mengatur AWS Direktori menggunakan AD Connector

1. Buka konsol [Directory Service](#) dan pilih Directories.
2. Pilih Siapkan direktori.
3. Untuk jenis direktori, pilih AD Connector.
4. Pilih ukuran direktori, pilih Berikutnya, lalu pilih Subnet VPC dan Anda.
5. Pilih Berikutnya, lalu isi kolom sebagai berikut:
 - DNSNama direktori: masukkan nama domain yang Anda gunakan untuk Azure ADDS Anda.
 - DNSAlamat IP: masukkan alamat ADDS IP Azure Anda.
 - Nama pengguna dan kata sandi akun server: masukkan detail untuk akun layanan yang Anda buat di Langkah 2: Buat akun layanan.
6. Lengkapi layar untuk membuat layanan direktori.

Sekarang status direktori harus Aktif, dan siap digunakan dengan server SFTP Transfer.

[Directory Service](#) > Directories

 **Did you know?**
You can deploy an AWS Managed Microsoft AD (Enterprise Edition) directory across AWS Regions. Once configured, AWS automatically replicates your directory data in multiple Regions so everything stays in sync. [Learn more](#)

Directories (1) [Info](#)

Refresh

Actions

Set up directory

< 1 >

Settings

Directory ID	Directory name	Type	Size	Multi-Region	Status	Launch date
 d-906752c0d7		AD Connector	Small	Not applicable	 Active	Nov 3, 2021

Langkah 4: Menyiapkan AWS Transfer Family server

Buat server Transfer Family dengan SFTP protokol, dan jenis penyedia identitas AWS Directory Service. Dari daftar drop-down Directory, pilih direktori yang Anda tambahkan di Langkah 3: Setup AWS Directory menggunakan AD Connector.

Note

Anda tidak dapat menghapus direktori Microsoft AD di AWS Directory Service jika Anda menggunakannya di server Transfer Family. Anda harus menghapus server terlebih dahulu, dan kemudian Anda dapat menghapus direktori.

Langkah 5: Memberikan akses ke grup

Setelah Anda membuat server, Anda harus memilih grup mana di direktori yang harus memiliki akses untuk mengunggah dan mengunduh file melalui protokol yang diaktifkan menggunakan. AWS Transfer Family Anda melakukan ini dengan membuat akses.

Note

Pengguna harus menjadi bagian langsung dari grup tempat Anda memberikan akses. Misalnya, asumsikan bahwa Bob adalah pengguna dan dia milik GroupA, dan GroupA sendiri termasuk dalam GroupB.

- Jika Anda memberikan akses ke GroupA, Bob diberikan akses.
- Jika Anda memberikan akses ke GroupB (dan bukan ke GroupA), Bob tidak memiliki akses.

Untuk memberikan akses, Anda perlu mengambil SID untuk grup.

Gunakan PowerShell perintah Windows berikut untuk mengambil SID untuk grup, menggantikan *YourGroupName* dengan nama grup.

```
Get-ADGroup -Filter {samAccountName -like "YourGroupName*"} -Properties * | Select SamAccountName, ObjectSid
```

```

Windows PowerShell
Copyright (C) 2016 Microsoft Corporation. All rights reserved.

PS C:\Users\bobatusa> Get-ADGroup -Filter {samAccountName -like "AAD DC Administrat
mAccountName,ObjectSid

SamAccountName      ObjectSid
-----
AAD DC Administrators S-1-5-21-375932292-1747164136-3628472596-1104

```

Berikan akses ke grup

1. Buka <https://console.aws.amazon.com/transfer/>.
2. Arahkan ke halaman detail server Anda dan di bagian Akses, pilih Tambahkan akses.
3. Masukkan yang SID Anda terima dari output dari prosedur sebelumnya.
4. Untuk Access, pilih AWS Identity and Access Management peran untuk grup.
5. Di bagian Kebijakan, pilih kebijakan. Nilai defaultnya adalah None.
6. Untuk direktori Home, pilih bucket S3 yang sesuai dengan direktori home grup.
7. Pilih Tambah untuk membuat asosiasi.

Detail dari server Transfer Anda akan terlihat mirip dengan yang berikut ini:

Protocols
Edit

Protocols over which clients can connect to your server's endpoint

- SFTP

Identity provider
Edit

Identity provider type
AWS Directory Service

Directory ID
d-123456789a

Accesses (1)

Actions
Add access

< 1 >

☐	External Id	Home directory	Role
☐	S-1-5-21-375932292-1747164136-3628472596-1104	/s3/transfer	sftp-user-role ↗

Langkah 6: Menguji pengguna

Anda dapat menguji ([Menguji pengguna](#)) apakah pengguna memiliki akses ke AWS Managed Microsoft AD direktori untuk server Anda. Seorang pengguna harus berada dalam satu grup (ID eksternal) yang tercantum di bagian Access pada halaman konfigurasi Endpoint. Jika pengguna tidak berada dalam grup, atau berada di lebih dari satu grup, pengguna tersebut tidak diberikan akses.

Bekerja dengan penyedia identitas khusus

Untuk mengautentikasi pengguna Anda, Anda dapat menggunakan penyedia identitas yang ada dengan AWS Transfer Family. Anda mengintegrasikan penyedia identitas Anda menggunakan AWS Lambda fungsi, yang mengautentikasi dan memberi wewenang kepada pengguna Anda untuk mengakses Amazon S3 atau Amazon Elastic File System (Amazon). EFS Untuk detailnya, lihat [Menggunakan AWS Lambda untuk mengintegrasikan penyedia identitas Anda](#). Anda juga dapat mengakses CloudWatch grafik untuk metrik seperti jumlah file dan byte yang ditransfer di AWS Transfer Family Management Console, memberi Anda satu panel kaca untuk memantau transfer file menggunakan dasbor terpusat.

Atau, Anda dapat menyediakan RESTful antarmuka dengan satu metode Amazon API Gateway. Transfer Family memanggil metode ini untuk terhubung ke penyedia identitas Anda, yang mengautentikasi dan memberi wewenang kepada pengguna Anda untuk mengakses Amazon S3 atau Amazon. EFS Gunakan opsi ini jika Anda memerlukan a RESTful API untuk mengintegrasikan penyedia identitas Anda atau jika Anda ingin menggunakannya untuk memanfaatkan kemampuannya AWS WAF untuk permintaan pemblokiran geografis atau pembatasan laju. Untuk detailnya, lihat [Menggunakan Amazon API Gateway untuk mengintegrasikan penyedia identitas Anda](#).

Dalam kedua kasus, Anda dapat membuat server baru menggunakan [AWS Transfer Family konsol](#) atau [CreateServerAPI](#) operasi.

Note

Kami memiliki lokakarya yang dapat Anda hadiri, di mana Anda dapat membangun solusi transfer file. Solusi ini memanfaatkan FTPS titik akhir AWS Transfer Family yang dikelola dan Amazon SFTP Cognito dan DynamoDB untuk manajemen pengguna. Anda dapat melihat detail untuk lokakarya ini [di sini](#).

AWS Transfer Family menyediakan opsi berikut untuk bekerja dengan penyedia identitas khusus.

- Gunakan AWS Lambda untuk menghubungkan penyedia identitas Anda — Anda dapat menggunakan penyedia identitas yang ada, didukung oleh fungsi Lambda. Anda memberikan nama fungsi Lambda. Untuk informasi selengkapnya, lihat [Menggunakan AWS Lambda untuk mengintegrasikan penyedia identitas Anda](#).
- Gunakan Amazon API Gateway untuk menghubungkan penyedia identitas Anda — Anda dapat membuat metode API Gateway yang didukung oleh fungsi Lambda untuk digunakan sebagai penyedia identitas. Anda menyediakan Amazon API Gateway URL dan peran pemanggilan. Untuk informasi selengkapnya, lihat [Menggunakan Amazon API Gateway untuk mengintegrasikan penyedia identitas Anda](#).

Untuk salah satu opsi, Anda juga dapat menentukan cara mengautentikasi.

- Kata Sandi ATAU Kunci — pengguna dapat mengautentikasi dengan kata sandi atau kunci mereka. Ini adalah nilai default.
- Kata sandi ONLY — pengguna harus memberikan kata sandi mereka untuk terhubung.
- Kunci ONLY — pengguna harus menyediakan kunci pribadi mereka untuk terhubung.
- ANDKunci Kata Sandi — pengguna harus memberikan kunci pribadi dan kata sandi mereka untuk terhubung. Server memeriksa kunci terlebih dahulu, dan kemudian jika kuncinya valid, sistem meminta kata sandi. Jika kunci pribadi yang diberikan tidak cocok dengan kunci publik yang disimpan, otentikasi gagal.

Menggunakan beberapa metode otentikasi untuk mengautentikasi dengan penyedia identitas kustom Anda

Server Transfer Family mengontrol AND logika saat Anda menggunakan beberapa metode otentikasi. Transfer Family memperlakukan ini sebagai dua permintaan terpisah ke penyedia identitas kustom Anda: namun, efeknya digabungkan.

Kedua permintaan harus berhasil dikembalikan dengan respons yang benar untuk memungkinkan otentikasi selesai. Transfer Family mengharuskan dua tanggapan lengkap, artinya berisi semua elemen yang diperlukan (peran, direktori beranda, kebijakan, dan POSIX profil jika Anda menggunakan Amazon EFS untuk penyimpanan). Transfer Family juga mensyaratkan bahwa respons kata sandi tidak boleh menyertakan kunci publik.

Permintaan kunci publik harus memiliki respons terpisah dari penyedia identitas. Perilaku itu tidak berubah saat menggunakan Kata Sandi ATAU Kunci atau AND Kunci Kata Sandi.

SSH/SFTP protokol/menantang klien perangkat lunak terlebih dahulu dengan otentikasi kunci publik, kemudian meminta otentikasi kata sandi. Operasi ini mengamankan keduanya berhasil sebelum pengguna diizinkan untuk menyelesaikan otentikasi.

Topik

- [Menggunakan AWS Lambda untuk mengintegrasikan penyedia identitas Anda](#)
- [Menggunakan Amazon API Gateway untuk mengintegrasikan penyedia identitas Anda](#)

Menggunakan AWS Lambda untuk mengintegrasikan penyedia identitas Anda

Buat AWS Lambda fungsi yang terhubung ke penyedia identitas kustom Anda. Anda dapat menggunakan penyedia identitas kustom apa pun, seperti Okta, Secrets Manager OneLogin, atau penyimpanan data khusus yang menyertakan logika otorisasi dan otentikasi.

Note

Sebelum membuat server Transfer Family yang menggunakan Lambda sebagai penyedia identitas, Anda harus membuat fungsinya. Untuk contoh fungsi Lambda, lihat [Contoh fungsi Lambda](#). Atau, Anda dapat menerapkan CloudFormation tumpukan yang menggunakan salah satu [Template fungsi Lambda](#). Selain itu, pastikan fungsi Lambda Anda menggunakan kebijakan berbasis sumber daya yang mempercayai Transfer Family. Untuk contoh kebijakan, lihat [Kebijakan berbasis sumber daya Lambda](#).

1. Buka [konsol AWS Transfer Family](#).
2. Pilih Buat server untuk membuka halaman Buat server. Untuk Pilih penyedia identitas, pilih Penyedia Identitas Kustom, seperti yang ditunjukkan pada gambar berikut.

Choose an 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

AWS Lambda function

Choose a Lambda function



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

Previous

Next

Note

Pilihan metode otentikasi hanya tersedia jika Anda mengaktifkan SFTP sebagai salah satu protokol untuk server Transfer Family Anda.

3. Pastikan nilai default, Gunakan AWS Lambda untuk menghubungkan penyedia identitas Anda, dipilih.
4. Untuk AWS Lambda fungsi, pilih nama fungsi Lambda Anda.

5. Isi kotak yang tersisa, lalu pilih Buat server. Untuk detail tentang langkah-langkah yang tersisa untuk membuat server, lihat [Mengkonfigurasi titik SFTP akhir, FTPS, atau FTP server](#).

Kebijakan berbasis sumber daya Lambda

Anda harus memiliki kebijakan yang mereferensikan server Transfer Family dan Lambda ARNs. Misalnya, Anda dapat menggunakan kebijakan berikut dengan fungsi Lambda yang terhubung ke penyedia identitas Anda. Kebijakan ini diloloskan JSON sebagai string.

```
"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

Dalam contoh kebijakan di atas, ganti masing-masing *user input placeholder* dengan informasi Anda sendiri.

Struktur pesan peristiwa

Struktur pesan acara dari SFTP server yang dikirim ke fungsi Lambda otorisasi untuk IDP kustom adalah sebagai berikut.

```
{
  'username': 'value',
  'password': 'value',
  'protocol': 'SFTP',
  'serverId': 's-abcd123456',
  'sourceIp': '192.168.0.100'
}
```

Di mana username dan password merupakan nilai untuk kredensial masuk yang dikirim ke server.

Misalnya, Anda memasukkan perintah berikut untuk menghubungkan:

```
sftp bobusa@server_hostname
```

Anda kemudian diminta untuk memasukkan kata sandi Anda:

```
Enter password:
mysecretpassword
```

Anda dapat memeriksa ini dari fungsi Lambda Anda dengan mencetak peristiwa yang diteruskan dari dalam fungsi Lambda. Seharusnya terlihat mirip dengan blok teks berikut.

```
{
  'username': 'bobusa',
  'password': 'mysecretpassword',
  'protocol': 'SFTP',
  'serverId': 's-abcd123456',
  'sourceIp': '192.168.0.100'
}
```

Struktur acara serupa untuk FTP danFTPS: satu-satunya perbedaan adalah nilai-nilai tersebut digunakan untuk `protocol` parameter, bukanSFTP.

Fungsi Lambda untuk otentikasi

Untuk menerapkan strategi otentikasi yang berbeda, edit fungsi Lambda. Untuk membantu Anda memenuhi kebutuhan aplikasi Anda, Anda dapat menerapkan CloudFormation tumpukan.

Untuk informasi selengkapnya tentang Lambda, lihat [Panduan AWS Lambda Pengembang](#) atau Membangun fungsi [Lambda](#) dengan Node.js.

Topik

- [Template fungsi Lambda](#)
- [Nilai Lambda yang valid](#)
- [Contoh fungsi Lambda](#)
- [Menguji konfigurasi Anda](#)

Template fungsi Lambda

Anda dapat menerapkan AWS CloudFormation tumpukan yang menggunakan fungsi Lambda untuk otentikasi. Kami menyediakan beberapa templat yang mengautentikasi dan mengotorisasi pengguna Anda menggunakan kredensi masuk. Anda dapat memodifikasi template atau AWS Lambda kode ini untuk lebih menyesuaikan akses pengguna.

Note

Anda dapat membuat FIPS AWS Transfer Family server yang diaktifkan AWS CloudFormation dengan menentukan kebijakan keamanan FIPS yang diaktifkan di template Anda. Kebijakan keamanan yang tersedia dijelaskan dalam [Kebijakan keamanan untuk AWS Transfer Family](#)

Untuk membuat AWS CloudFormation tumpukan yang akan digunakan untuk otentikasi

1. Buka AWS CloudFormation konsol di <https://console.aws.amazon.com/cloudformation>.
2. Ikuti petunjuk untuk menerapkan AWS CloudFormation tumpukan dari template yang ada di [Memilih template tumpukan](#) di Panduan AWS CloudFormation Pengguna.
3. Gunakan salah satu templat berikut untuk membuat fungsi Lambda yang akan digunakan untuk otentikasi di Transfer Family.
 - [Templat tumpukan klasik \(Amazon Cognito\)](#)

Template dasar untuk membuat AWS Lambda untuk digunakan sebagai penyedia identitas kustom di AWS Transfer Family. Ini mengautentikasi terhadap Amazon Cognito untuk otentikasi berbasis kata sandi dan kunci publik dikembalikan dari bucket Amazon S3 jika

otentikasi berbasis kunci publik digunakan. Setelah penerapan, Anda dapat memodifikasi kode fungsi Lambda untuk melakukan sesuatu yang berbeda.

- [AWS Secrets Manager template tumpukan](#)

Template dasar yang digunakan AWS Lambda dengan AWS Transfer Family server untuk mengintegrasikan Secrets Manager sebagai penyedia identitas. Ini mengotentikasi terhadap entri dalam AWS Secrets Manager format `aws/transfer/server-id/username`. Selain itu, secret harus menyimpan pasangan kunci-nilai untuk semua properti pengguna yang dikembalikan ke Transfer Family. Setelah penerapan, Anda dapat memodifikasi kode fungsi Lambda untuk melakukan sesuatu yang berbeda.

- [Template tumpukan Okta](#): Template dasar yang digunakan AWS Lambda dengan AWS Transfer Family server untuk mengintegrasikan Okta sebagai penyedia identitas khusus.
- [Template tumpukan Okta-MFA: Template](#) dasar yang digunakan AWS Lambda dengan AWS Transfer Family server untuk mengintegrasikan Okta, dengan MultiFactor Otentikasi, sebagai penyedia identitas khusus.
- [Template Azure Active Directory](#): detail untuk tumpukan ini dijelaskan dalam posting blog [Mengautentikasi AWS Transfer Family dengan Azure Active Directory](#) dan. AWS Lambda

Setelah tumpukan digunakan, Anda dapat melihat detailnya di tab Output di CloudFormation konsol.

Menerapkan salah satu tumpukan ini adalah cara termudah untuk mengintegrasikan penyedia identitas kustom ke dalam alur kerja Transfer Family.

Nilai Lambda yang valid

Tabel berikut menjelaskan detail nilai yang diterima Transfer Family untuk fungsi Lambda yang digunakan untuk penyedia identitas kustom.

Nilai	Deskripsi	Diperlukan
Role	Menentukan Nama Sumber Daya Amazon (ARN) IAM peran yang mengontrol akses pengguna ke bucket Amazon S3 atau sistem file	Diperlukan

Nilai	Deskripsi	Diperlukan
	AmazonEFS. Kebijakan yang dilampirkan pada peran ini menentukan tingkat akses yang ingin Anda berikan kepada pengguna saat mentransfer file masuk dan keluar dari sistem file Amazon S3 atau EFS Amazon Anda. IAMPeran tersebut juga harus berisi hubungan kepercayaan yang memungkinkan server mengakses sumber daya Anda saat melayani permintaan transfer pengguna Anda. Untuk detail tentang membangun hubungan kepercayaan, lihatUntuk membangun hubungan kepercayaan.	

Nilai	Deskripsi	Diperlukan
PosixProfile	POSIXIdentitas lengkap, termasuk ID pengguna (Uid), ID grup (Gid), dan grup sekunder IDs (Secondary Gids) apa pun, yang mengontrol akses pengguna ke sistem EFS file Amazon Anda. POSIXIzin yang ditetapkan pada file dan direktori di sistem file Anda menentukan tingkat akses yang didapat pengguna saat mentransfer file masuk dan keluar dari sistem file Amazon EFS Anda.	Diperlukan untuk penyimpanan EFS dukungan Amazon
PublicKeys	Daftar nilai kunci SSH publik yang valid untuk pengguna ini. Daftar kosong menyiratkan bahwa ini bukan login yang valid. Tidak boleh dikembalikan selama otentikasi kata sandi.	Opsional
Policy	Kebijakan sesi untuk pengguna Anda sehingga Anda dapat menggunakan IAM peran yang sama di beberapa pengguna. Kebijakan ini mencakup bawah akses pengguna ke bagian dari bucket Amazon S3 mereka.	Opsional

Nilai	Deskripsi	Diperlukan
HomeDirectoryType	Jenis direktori pendaratan (folder) yang Anda inginkan direktori home pengguna Anda ketika mereka masuk ke server. • Jika Anda mengaturnya <code>yaPATH</code>, pengguna akan melihat bucket Amazon S3 absolut atau EFS jalur Amazon seperti pada klien protokol transfer file mereka. • Jika Anda menyetelnya <code>yaLOGICAL</code>, Anda harus menyediakan pemetaan dalam <code>HomeDirectoryDetails</code> parameter agar jalur Amazon S3 atau EFS Amazon terlihat oleh pengguna Anda.	Opsional

Nilai	Deskripsi	Diperlukan
<code>HomeDirectoryDetails</code>	Pemetaan direktori logis yang menentukan jalur dan kunci Amazon S3 atau EFS Amazon mana yang harus terlihat oleh pengguna Anda dan bagaimana Anda ingin membuatnya terlihat. Anda harus menentukan Entry dan Target memasangkan, di mana Entry menunjukkan bagaimana jalur dibuat terlihat dan Target merupakan jalur Amazon S3 atau Amazon yang sebenarnya. EFS	Diperlukan jika <code>HomeDirectoryType</code> memiliki nilai LOGICAL
<code>HomeDirectory</code>	Direktori pendaratan untuk pengguna ketika mereka masuk ke server menggunakan klien.	Opsional

Note

`HomeDirectoryDetails` adalah representasi string dari JSON peta. Ini berbeda dengan `PosixProfile`, yang merupakan objek JSON peta yang sebenarnya, dan `PublicKeys` yang merupakan JSON array string. Lihat contoh kode untuk detail khusus bahasa.

Contoh fungsi Lambda

Bagian ini menyajikan beberapa contoh fungsi Lambda, baik di NodeJS maupun Python.

Note

Dalam contoh ini, detail direktori pengguna, peran, POSIX profil, kata sandi, dan home semuanya adalah contoh, dan harus diganti dengan nilai aktual Anda.

Logical home directory, NodeJS

Fungsi contoh NodeJS berikut memberikan rincian untuk pengguna yang memiliki direktori home logis.

```
// 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

Fungsi contoh NodeJS berikut memberikan rincian untuk pengguna yang memiliki direktori home berbasis jalur.

```

// 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

Contoh fungsi Python berikut memberikan rincian untuk pengguna yang memiliki direktori [home logis](#).

```
# 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

Contoh fungsi Python berikut memberikan rincian untuk pengguna yang memiliki direktori home berbasis jalur.

```

# 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

```

Menguji konfigurasi Anda

Setelah Anda membuat penyedia identitas kustom Anda, Anda harus menguji konfigurasi Anda.

Console

Untuk menguji konfigurasi Anda dengan menggunakan AWS Transfer Family konsol

1. Buka [konsol AWS Transfer Family](#).
2. Pada halaman Server, pilih server baru Anda, pilih Tindakan, lalu pilih Uji.
3. Masukkan teks untuk Nama Pengguna dan Kata Sandi yang Anda atur saat Anda menerapkan AWS CloudFormation tumpukan. Jika Anda menyimpan opsi default, nama pengguna adalah myuser dan kata sandinyaMySuperSecretPassword.
4. Pilih protokol Server dan masukkan alamat IP untuk IP Sumber, jika Anda mengaturnya saat Anda menerapkan AWS CloudFormation tumpukan.

CLI

Untuk menguji konfigurasi Anda dengan menggunakan AWS CLI

1. Jalankan perintah [test-identity-provider](#). Ganti masing-masing *user input placeholder* dengan informasi Anda sendiri, seperti yang dijelaskan dalam langkah-langkah selanjutnya.

```
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. Masukkan ID server.
3. Masukkan nama pengguna dan kata sandi yang Anda tetapkan saat Anda menerapkan AWS CloudFormation tumpukan. Jika Anda menyimpan opsi default, nama pengguna adalah `myuser` dan kata sandinya `MySuperSecretPassword`.
4. Masukkan protokol server dan alamat IP sumber, jika Anda mengaturnya saat Anda menerapkan AWS CloudFormation tumpukan.

Jika otentikasi pengguna berhasil, pengujian mengembalikan `Status Code: 200 HTTP response`, string kosong `Message: ""` (yang akan berisi alasan kegagalan jika tidak), dan bidang `Response`

Note

Dalam contoh respons di bawah ini, `Response` bidang adalah JSON objek yang telah “dirangkai” (diubah menjadi JSON string datar yang dapat digunakan di dalam program), dan berisi rincian peran dan izin pengguna.

```
{
  "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\": \"\"  
}
```

Menggunakan Amazon API Gateway untuk mengintegrasikan penyedia identitas Anda

Topik ini menjelaskan cara menggunakan AWS Lambda fungsi untuk mendukung metode API Gateway. Gunakan opsi ini jika Anda memerlukan a RESTful API untuk mengintegrasikan penyedia identitas Anda atau jika Anda ingin menggunakannya untuk memanfaatkan kemampuannya AWS WAF untuk permintaan pemblokiran geografis atau pembatasan laju.

Batasan jika menggunakan API Gateway untuk mengintegrasikan penyedia identitas Anda

- Konfigurasi ini tidak mendukung domain kustom.
- Konfigurasi ini tidak mendukung API Gateway pribadiURL.

Jika Anda membutuhkan salah satu dari ini, Anda dapat menggunakan Lambda sebagai penyedia identitas, tanpa API Gateway. Untuk detailnya, lihat [Menggunakan AWS Lambda untuk mengintegrasikan penyedia identitas Anda](#).

Mengautentikasi menggunakan metode API Gateway

Anda dapat membuat metode API Gateway untuk digunakan sebagai penyedia identitas untuk Transfer Family. Pendekatan ini memberikan cara yang sangat aman bagi Anda untuk membuat dan menyediakan APIs. Dengan API Gateway, Anda dapat membuat HTTPS titik akhir sehingga semua API panggilan masuk ditransmisikan dengan keamanan yang lebih besar. Untuk detail selengkapnya tentang layanan API Gateway, lihat [Panduan Pengembang API Gateway](#).

APIGateway menawarkan metode otorisasi bernamaAWS_IAM, yang memberi Anda autentikasi yang sama berdasarkan AWS Identity and Access Management (IAM) yang AWS digunakan secara internal. Jika Anda mengaktifkan autentikasi denganAWS_IAM, hanya penelepon dengan izin eksplisit untuk memanggil yang API dapat mencapai metode Gateway ituAPI. API

Untuk menggunakan metode API Gateway Anda sebagai penyedia identitas khusus untuk Transfer Family, aktifkan IAM metode API Gateway Anda. Sebagai bagian dari proses ini, Anda memberikan IAM peran dengan izin untuk Transfer Family untuk menggunakan gateway Anda.

Note

Untuk meningkatkan keamanan, Anda dapat mengkonfigurasi firewall aplikasi web. AWS WAF adalah firewall aplikasi web yang memungkinkan Anda memantau HTTP dan HTTPS permintaan yang diteruskan ke Amazon API Gateway. Untuk detailnya, lihat [Tambahkan firewall aplikasi web](#).

Untuk menggunakan metode API Gateway Anda untuk autentikasi kustom dengan Transfer Family

1. Buat AWS CloudFormation tumpukan. Untuk melakukannya:

 Note

Templat tumpukan telah diperbarui untuk menggunakan kata sandi yang BASE64 dienkode: untuk detailnya, lihat. [Perbaikan AWS CloudFormation template](#)

- a. Buka AWS CloudFormation konsol di <https://console.aws.amazon.com/cloudformation>.
- b. Ikuti petunjuk untuk menerapkan AWS CloudFormation tumpukan dari template yang ada di [Memilih template tumpukan](#) di Panduan AWS CloudFormation Pengguna.
- c. Gunakan salah satu templat dasar berikut untuk membuat metode API Gateway yang AWS Lambda didukung untuk digunakan sebagai penyedia identitas khusus di Transfer Family.

- [Template tumpukan dasar](#)

Secara default, metode API Gateway Anda digunakan sebagai penyedia identitas khusus untuk mengautentikasi satu pengguna dalam satu server menggunakan kunci atau kata sandi hard-code SSH (Secure Shell). Setelah penerapan, Anda dapat memodifikasi kode fungsi Lambda untuk melakukan sesuatu yang berbeda.

- [AWS Secrets Manager template tumpukan](#)

Secara default, metode API Gateway Anda mengautentikasi terhadap entri di Secrets Manager format `aws/transfer/server-id/username`. Selain itu, secret harus menyimpan pasangan kunci-nilai untuk semua properti pengguna yang dikembalikan ke Transfer Family. Setelah penerapan, Anda dapat memodifikasi kode fungsi Lambda untuk melakukan sesuatu yang berbeda. Untuk informasi selengkapnya, lihat posting blog [Aktifkan otentikasi kata sandi untuk AWS Transfer Family digunakan AWS Secrets Manager](#).

- [Templat tumpukan Okta](#)

Metode API Gateway Anda terintegrasi dengan Okta sebagai penyedia identitas khusus di Transfer Family. Untuk informasi lebih lanjut, lihat posting blog [Menggunakan Okta sebagai penyedia identitas dengan AWS Transfer Family](#).

Menerapkan salah satu tumpukan ini adalah cara termudah untuk mengintegrasikan penyedia identitas kustom ke dalam alur kerja Transfer Family. Setiap tumpukan menggunakan fungsi Lambda untuk mendukung API metode Anda berdasarkan API Gateway. Anda kemudian dapat

menggunakan API metode Anda sebagai penyedia identitas khusus di Transfer Family. Secara default, fungsi Lambda mengotentikasi satu pengguna yang dipanggil `myuser` dengan kata sandi `MySuperSecretPassword`. Setelah penerapan, Anda dapat mengedit kredensi ini atau memperbarui kode fungsi Lambda untuk melakukan sesuatu yang berbeda.

 Important

Kami menyarankan Anda mengedit kredensi pengguna dan kata sandi default.

Setelah tumpukan digunakan, Anda dapat melihat detailnya di tab Output di CloudFormation konsol. Detail ini termasuk Amazon Resource Name (ARN) stack, IAM peran yang dibuat stack, dan URL untuk gateway baru Anda. ARN

 Note

Jika Anda menggunakan opsi penyedia identitas khusus untuk mengaktifkan autentikasi berbasis kata sandi bagi pengguna Anda, dan Anda mengaktifkan pencatatan permintaan dan respons yang disediakan oleh API Gateway, API Gateway mencatat kata sandi pengguna Anda ke Log Amazon Anda. CloudWatch Kami tidak menyarankan menggunakan log ini di lingkungan produksi Anda. Untuk informasi selengkapnya, lihat [Mengatur CloudWatch API login di API Gateway](#) di Panduan Pengembang API Gateway.

2. Periksa konfigurasi metode API Gateway untuk server Anda. Untuk melakukannya:
 - a. Buka konsol API Gateway di <https://console.aws.amazon.com/apigateway/>.
 - b. Pilih template dasar Transfer Custom Identity Provider API yang dihasilkan AWS CloudFormation template. Anda mungkin perlu memilih wilayah Anda untuk melihat gateway Anda.
 - c. Di panel Resources, pilih GET. Tangkapan layar berikut menunjukkan konfigurasi metode yang benar.

The screenshot displays the AWS API Gateway console interface. On the left, a sidebar shows a resource tree with paths: /, /servers, /serverid, /users, /username, and /config. The 'GET' method is selected for the /config path. The main panel shows the 'Method request' settings for this method. At the top, there are tabs for 'Method request', 'Integration request', 'Integration response', 'Method response', and 'Test'. The 'Method request' tab is active, showing 'Method request settings'. The settings include: Authorization (AWS_IAM), API key required (False), Request validator (None), and SDK operation name (Generated based on method and path). Below these are sections for 'Request paths (0)', 'URL query string parameters (2)', 'HTTP request headers (1)', and 'Request body (0)'. Each section has a table with columns for Name, Required, and Caching. The 'Request paths' section shows 'No request paths defined'. The 'URL query string parameters' section shows two parameters: 'protocol' and 'sourcelp', both with Required set to False and Caching set to 'Inactive'. The 'HTTP request headers' section shows one header: 'PasswordBase64', with Required set to False and Caching set to 'Inactive'. The 'Request body' section shows 'No request body defined'.

Pada titik ini, API gateway Anda siap digunakan.

- Untuk Tindakan, pilih Terapkan API. Untuk tahap Deployment, pilih prod, lalu pilih Deploy.

Setelah metode API Gateway berhasil digunakan, lihat kinerjanya di Tahapan> Detail tahap, seperti yang ditunjukkan pada gambar berikut.

Note

Salin URL alamat Invoke yang muncul di bagian atas layar. Anda mungkin membutuhkannya untuk langkah selanjutnya.

The screenshot shows the AWS Transfer Family console interface. At the top, the breadcrumb navigation reads: API Gateway > APIs > Transfer Custom Identity Provider basic template API > Stages. The main heading is 'Stages'. On the left, a sidebar shows a list of stages with 'prod' selected. The main content area is divided into several sections:

- Stage details info**: Contains fields for Stage name (prod), API cache (Inactive), Rate (10000), Burst (5000), Web ACL (-), and Client certificate (-). The 'Invoke URL' field is highlighted with a red box and contains the value: `https://[redacted].execute-api.us-east-1.amazonaws.com/prod`.
- Active deployment**: Shows the deployment 't8aqrm' on December 12, 2023, at 10:49 (UTC-05:00).
- Logs and tracing info**: Contains fields for CloudWatch logs (Error and info logs), Detailed metrics (Inactive), X-Ray tracing (Inactive), and Custom access logging (Inactive).
- Stage variables**: A section with a search bar and a table for variables. The table is currently empty, showing 'No variables' and 'No variables associated with the stage.' A 'Manage variables' button is at the bottom.

4. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
5. Transfer Family seharusnya dibuat untuk Anda, saat Anda membuat tumpukan. Jika tidak, konfigurasi server Anda menggunakan langkah-langkah ini.
 - a. Pilih Buat server untuk membuka halaman Buat server. Untuk Pilih penyedia identitas, pilih Kustom, lalu pilih Gunakan Amazon API Gateway untuk terhubung ke penyedia identitas Anda, seperti yang ditunjukkan pada gambar berikut.

Choose an identity provider

Identity provider

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

Role
IAM role for the service to invoke your Amazon API Gateway URL

- b. Di kotak URL teks Berikan Amazon API Gateway, tempel URL alamat Invoke dari titik akhir API Gateway yang Anda buat di langkah 3 prosedur ini.
- c. Untuk Peran, pilih IAM peran yang dibuat oleh AWS CloudFormation template. Peran ini memungkinkan Transfer Family untuk memanggil metode API gateway Anda.

Peran pemanggilan berisi nama AWS CloudFormation tumpukan yang Anda pilih untuk tumpukan yang Anda buat di langkah 1. Ini memiliki format berikut: *CloudFormation-stack-name*-TransferIdentityProviderRole-*ABC123DEF456GHI*.

- d. Isi kotak yang tersisa, lalu pilih Buat server. Untuk detail tentang langkah-langkah yang tersisa untuk membuat server, lihat [Mengkonfigurasi titik SFTP akhir, FTPS, atau FTP server](#).

Menerapkan metode API Gateway Anda

Untuk membuat penyedia identitas kustom untuk Transfer Family, metode API Gateway Anda harus menerapkan satu metode yang memiliki jalur sumber daya/*servers/serverId*/

users/*username*/config. *username* Nilai *serverId* dan berasal dari jalur RESTful sumber daya. Juga, tambahkan sourceIp dan protocol sebagai Parameter String URL Kueri dalam Permintaan Metode, seperti yang ditunjukkan pada gambar berikut.

Resources

Create resource

/

/servers

/servers/{serverId}

/servers/{serverId}/users

/servers/{serverId}/users/{username}

/servers/{serverId}/users/{username}/config

GET

API actions

Deploy API

/servers/{serverId}/users/{username}/config - GET - Method execution

Update documentation

Delete

ARN

arn:aws:execute-api:us-east-1:123456789012:*/GET/servers/{serverId}/users/{username}/config

Resource ID

aw4ihv

Client

Method request

Method response

Integration request

Integration response

Lambda integration

Method request

Integration request

Integration response

Method response

Test

Method request settings

Edit

Authorization

AWS_IAM

API key required

False

Request validator

None

SDK operation name

Generated based on method and path

Request paths (0)

< 1 >

Name

▼

Caching

▼

No request paths

No request paths defined

URL query string parameters (2)

< 1 >

Name

▼

Required

▼

Caching

▼

protocol

False

Inactive

sourceIp

False

Inactive

 Note

Nama pengguna harus minimal 3 dan maksimal 100 karakter. Anda dapat menggunakan karakter berikut dalam nama pengguna: a—z, A-Z, 0—9, garis bawah (_), tanda hubung (-), titik (.), dan di tanda (@). Namun, nama pengguna tidak dapat dimulai dengan tanda hubung (-), titik (.), atau di tanda (@).

Jika Transfer Family mencoba otentikasi kata sandi untuk pengguna Anda, layanan akan menyediakan bidang Password : header. Jika tidak ada Password : header, Transfer Family mencoba otentikasi kunci publik untuk mengautentikasi pengguna Anda.

Saat Anda menggunakan penyedia identitas untuk mengautentikasi dan mengotorisasi pengguna akhir, selain memvalidasi kredensialnya, Anda dapat mengizinkan atau menolak permintaan akses berdasarkan alamat IP klien yang digunakan oleh pengguna akhir Anda. Anda dapat menggunakan fitur ini untuk memastikan bahwa data yang disimpan di bucket S3 atau sistem EFS file Amazon Anda dapat diakses melalui protokol yang didukung hanya dari alamat IP yang telah Anda tentukan sebagai tepercaya. Untuk mengaktifkan fitur ini, Anda harus menyertakan `sourceIp` dalam string Query.

Jika Anda memiliki beberapa protokol yang diaktifkan untuk server Anda dan ingin memberikan akses menggunakan nama pengguna yang sama melalui beberapa protokol, Anda dapat melakukannya selama kredensial khusus untuk setiap protokol telah diatur di penyedia identitas Anda. Untuk mengaktifkan fitur ini, Anda harus menyertakan *protocol* nilai di jalur RESTful sumber daya.

Metode API Gateway Anda harus selalu mengembalikan kode HTTP status 200. Kode HTTP status lainnya berarti ada kesalahan saat mengakses file. API

Contoh respons Amazon S3

Contoh badan respons adalah JSON dokumen formulir berikut untuk 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

Kebijakan ini diloloskan JSON sebagai string. Sebagai contoh:

```
"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\"}]
  }"

```

Contoh respon berikut menunjukkan bahwa pengguna memiliki tipe direktori home logis.

```

{
  "Role": "arn:aws:iam::123456789012:role/transfer-access-role-s3",
  "HomeDirectoryType": "LOGICAL",
  "HomeDirectoryDetails": "[{\"Entry\": \"\", \"Target\": \"/MY-HOME-BUCKET\"}]",
  "PublicKeys": [""],
}

```

EFSContoh respons Amazon

Contoh badan respons adalah JSON dokumen formulir berikut untuk AmazonEFS.

```

{
  "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"
}

```

RoleBidang menunjukkan bahwa otentikasi berhasil terjadi. Saat melakukan otentikasi kata sandi (saat Anda menyediakan Password: header), Anda tidak perlu memberikan kunci SSH publik. Jika pengguna tidak dapat diautentikasi, misalnya, jika kata sandi salah, metode Anda harus mengembalikan respons tanpa Role disetel. Contoh respons semacam itu adalah JSON objek kosong.

Contoh respon berikut menunjukkan pengguna yang memiliki tipe direktori home logis.

```

{
  "Role": "arn:aws:iam::123456789012:role/transfer-access-role-efs",
  "HomeDirectoryType": "LOGICAL",
  "HomeDirectoryDetails": "[{\"Entry\": \"\\\"/\\\"\", \"Target\": \"\\\"/faa1a123\\\"}]\"",
  "PublicKeys": [""],
  "PosixProfile": {"Uid": 65534, "Gid": 65534}
}

```

Anda dapat menyertakan kebijakan pengguna dalam fungsi Lambda dalam JSON format. Untuk informasi selengkapnya tentang mengonfigurasi kebijakan pengguna di Transfer Family, lihat [Mengelola kontrol akses](#).

Fungsi Lambda default

Untuk menerapkan strategi otentikasi yang berbeda, edit fungsi Lambda yang digunakan gateway Anda. Untuk membantu Anda memenuhi kebutuhan aplikasi Anda, Anda dapat menggunakan contoh fungsi Lambda berikut di Node.js. Untuk informasi selengkapnya tentang Lambda, lihat [Panduan AWS Lambda Pengembang](#) atau Membangun fungsi [Lambda](#) dengan Node.js.

Contoh fungsi Lambda berikut mengambil nama pengguna, kata sandi (jika Anda melakukan otentikasi kata sandi), ID server, protokol, dan alamat IP klien. Anda dapat menggunakan kombinasi input ini untuk mencari penyedia identitas Anda dan menentukan apakah login harus diterima.

Note

Jika Anda memiliki beberapa protokol yang diaktifkan untuk server Anda dan ingin memberikan akses menggunakan nama pengguna yang sama melalui beberapa protokol,

Anda dapat melakukannya selama kredensial khusus untuk protokol telah diatur di penyedia identitas Anda.

Untuk File Transfer Protocol (FTP), kami sarankan untuk mempertahankan kredensial terpisah dari Secure Shell (SSH) File Transfer Protocol (SFTP) dan File Transfer Protocol over SSL (FTPS). Sebaiknya pertahankan kredensial terpisah FTP karena, tidak seperti SFTP dan FTPS, FTP mentransmisikan kredensial dalam teks yang jelas. Dengan mengisolasi FTP kredensial dari SFTP atau FTPS, jika FTP kredensial dibagikan atau diekspos, beban kerja Anda menggunakan atau tetap aman. SFTP FTPS

Fungsi contoh ini mengembalikan peran dan rincian direktori home logis, bersama dengan kunci publik (jika melakukan otentikasi kunci publik).

Saat Anda membuat pengguna yang dikelola layanan, Anda mengatur direktori home mereka, baik logis maupun fisik. Demikian pula, kita membutuhkan hasil fungsi Lambda untuk menyampaikan struktur direktori fisik atau logis pengguna yang diinginkan. Parameter yang Anda tetapkan tergantung pada nilai untuk [HomeDirectoryType](#) lapangan.

- `HomeDirectoryType` disetel ke `PATH` — `HomeDirectory` bidang tersebut kemudian harus berupa awalan bucket Amazon S3 absolut atau jalur EFS absolut Amazon yang dapat dilihat oleh pengguna Anda.
- `HomeDirectoryType` set ke `LOGICAL` - Jangan mengatur `HomeDirectory` bidang. Sebagai gantinya, kami menetapkan `HomeDirectoryDetails` bidang yang menyediakan pemetaan `Entry/Target` yang diinginkan, mirip dengan nilai yang dijelaskan di [HomeDirectoryDetails](#) parameter untuk pengguna yang dikelola layanan.

Contoh fungsi tercantum dalam [Contoh fungsi Lambda](#).

Fungsi Lambda untuk digunakan dengan AWS Secrets Manager

Untuk digunakan AWS Secrets Manager sebagai penyedia identitas Anda, Anda dapat bekerja dengan fungsi Lambda di template sampel AWS CloudFormation . Fungsi Lambda menanyakan layanan Secrets Manager dengan kredensial Anda dan, jika berhasil, mengembalikan rahasia yang ditentukan. Untuk informasi selengkapnya tentang Secrets Manager, lihat [Panduan Pengguna AWS Secrets Manager](#).

Untuk mengunduh contoh AWS CloudFormation template yang menggunakan fungsi Lambda ini, buka bucket [Amazon S3](#) yang disediakan oleh AWS Transfer Family

Perbaikan AWS CloudFormation template

Perbaikan pada antarmuka API Gateway telah dilakukan pada CloudFormation templat yang diterbitkan. Template sekarang menggunakan BASE64 -encoded password dengan Gateway. API Penerapan Anda yang ada terus berfungsi tanpa peningkatan ini, tetapi jangan izinkan kata sandi dengan karakter di luar set karakter AS dasar. ASCII

Perubahan dalam template yang mengaktifkan kemampuan ini adalah sebagai berikut:

- `GetUserConfigRequest` `AWS::ApiGateway::Method` sumber daya harus memiliki `RequestTemplates` kode ini (baris miring adalah baris yang diperbarui)

```
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')"
```

- `GetUserConfigSumber` daya harus diubah untuk menggunakan `PasswordBase64` header (baris miring adalah baris yang diperbarui): `RequestParameters`

```
RequestParameters:
  method.request.header.PasswordBase64: false
  method.request.querystring.protocol: false
  method.request.querystring.sourceIp: false
```

Untuk memeriksa apakah template untuk tumpukan Anda adalah yang terbaru

1. Buka AWS CloudFormation konsol di <https://console.aws.amazon.com/cloudformation>.
2. Dari daftar tumpukan, pilih tumpukan Anda.
3. Dari panel detail, pilih tab Template.
4. Cari yang berikut ini:
 - `CariRequestTemplates`, dan pastikan Anda memiliki baris ini:

```
"password":  
  "$util.escapeJavaScript($util.base64Decode($input.params('PasswordBase64'))).replaceAll(  
    \",\"\", \"'\", \"'\")",
```

- CariRequestParameters, dan pastikan Anda memiliki baris ini:

```
method.request.header.PasswordBase64: false
```

Jika Anda tidak melihat baris yang diperbarui, edit tumpukan Anda. Untuk detail tentang cara memperbarui AWS CloudFormation tumpukan Anda, lihat [Memodifikasi template tumpukan](#) di AWS CloudFormation; Panduan Pengguna.

Menggunakan direktori logis untuk menyederhanakan struktur direktori Transfer Family

Untuk menyederhanakan struktur direktori AWS Transfer Family server Anda, Anda dapat menggunakan direktori logis. Dengan direktori logis, Anda dapat membuat struktur direktori virtual yang menggunakan nama yang mudah digunakan yang dinavigasi pengguna saat mereka terhubung ke bucket Amazon S3 atau sistem file Amazon Anda. EFS Saat menggunakan direktori logis, Anda dapat menghindari pengungkapan jalur direktori absolut, nama bucket Amazon S3, EFS dan nama sistem file kepada pengguna akhir.

Note

Meskipun tergantung pada penggunaan kebijakan sesi dan persyaratan internal lainnya, Anda umumnya tidak memerlukan kebijakan sesi dan direktori logis untuk memastikan bahwa pengguna Anda hanya mengakses file yang Anda inginkan. Jika Anda menyiapkan direktori logis, maka jangan juga membuat kebijakan sesi, karena keduanya dapat menyebabkan kesalahan izin ditolak.

Anda dapat menggunakan direktori logis untuk mengatur direktori root pengguna ke lokasi yang diinginkan dalam hierarki penyimpanan Anda, dengan melakukan apa yang dikenal sebagai chroot operasi. Dalam mode ini, pengguna tidak dapat menavigasi ke direktori di luar direktori home atau root yang telah Anda konfigurasi untuk mereka.

Misalnya, meskipun pengguna Amazon S3 telah dicakup untuk mengakses saja `/mybucket/home/${transfer:UserName}`, beberapa klien mengizinkan pengguna untuk melintasi folder. `/mybucket/home` Dalam situasi ini, pengguna kembali ke direktori home yang dimaksudkan hanya setelah keluar dari dan kembali ke server Transfer Family lagi. Melakukan chroot operasi dapat mencegah situasi ini terjadi.

Anda dapat membuat struktur direktori Anda sendiri di seluruh bucket dan awalan. Fitur ini berguna jika Anda memiliki alur kerja yang mengharapkan struktur direktori tertentu yang tidak dapat Anda tiru melalui awalan bucket. Anda juga dapat menautkan ke beberapa lokasi yang tidak berdekatan dalam Amazon S3, mirip dengan membuat tautan simbolis dalam sistem file Linux di mana jalur direktori Anda mereferensikan lokasi yang berbeda dalam sistem file.

FILEPemetaan direktori logis

Tipe `HomeDirectoryMapEntry` data sekarang menyertakan `Type` parameter. Sebelum parameter ini ada, Anda bisa membuat pemetaan direktori logis di mana targetnya adalah file. Jika sebelumnya Anda telah membuat salah satu dari jenis pemetaan direktori logis ini, Anda harus secara eksplisit menyetelnya `Type` ke `FILE`, atau pemetaan ini tidak akan berfungsi dengan benar di masa mendatang.

Salah satu cara untuk melakukan ini adalah dengan memanggil `UpdateUserAPI`, dan mengatur `Type` ke `FILE` untuk pemetaan yang ada.

Aturan untuk menggunakan direktori logis

Sebelum Anda membangun pemetaan direktori logis Anda, Anda harus memahami aturan berikut:

- `EntryKapan"/`, Anda hanya dapat memiliki satu pemetaan karena jalur yang tumpang tindih tidak diperbolehkan.
- Direktori logis mendukung pemetaan hingga 2,1 MB (untuk pengguna yang dikelola layanan, batas ini adalah 2.000 entri). Artinya, struktur data yang berisi pemetaan memiliki ukuran maksimum 2,1 MB. Jika Anda memiliki banyak pemetaan, Anda dapat menghitung ukuran pemetaan Anda sebagai berikut:
 1. Tuliskan pemetaan khas dalam format `{"Entry": "/entry-path", "Target": "/target-path"}`, di mana `entry-path` dan `target-path` merupakan nilai aktual yang akan Anda gunakan.
 2. Hitung karakter dalam string itu, lalu tambahkan satu (1).

3. Kalikan angka itu dengan perkiraan jumlah pemetaan yang Anda miliki untuk server Anda.

Jika jumlah yang Anda perkirakan pada langkah 3 kurang dari 2,1 MB, maka pemetaan Anda berada dalam batas yang dapat diterima.

- Target dapat menggunakan `${transfer:UserName}` variabel jika bucket atau jalur sistem file telah diparameterisasi berdasarkan nama pengguna.
- Target dapat berupa jalur dalam bucket atau sistem file yang berbeda, tetapi Anda harus memastikan bahwa peran mapped AWS Identity and Access Management (IAM) (`Roleparameter` dalam respons) menyediakan akses ke bucket atau sistem file tersebut.
- Jangan tentukan `HomeDirectory` parameter, karena nilai ini tersirat oleh `Entry Target` pasangan saat Anda menggunakan `LOGICAL` nilai untuk `HomeDirectoryType` parameter tersebut.
- Target harus dimulai dengan karakter garis miring (/), tetapi jangan gunakan garis miring ke depan (/) saat Anda menentukan Target. Misalnya, dapat `/DOC-EXAMPLE-BUCKET/images` diterima, tetapi `DOC-EXAMPLE-BUCKET/images` dan `/DOC-EXAMPLE-BUCKET/images/` tidak.
- Amazon S3 adalah toko objek, yang berarti folder adalah konsep virtual, dan tidak ada hierarki direktori yang sebenarnya. Jika aplikasi Anda mengeluarkan stat operasi dari klien, semuanya diklasifikasikan sebagai file saat Anda menggunakan Amazon S3 untuk penyimpanan. Perilaku ini dijelaskan dalam [Mengatur objek di konsol Amazon S3 menggunakan folder](#) di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon. Jika aplikasi Anda mengharuskan untuk menunjukkan stat secara akurat apakah sesuatu adalah file atau folder, Anda dapat menggunakan Amazon Elastic File System (AmazonEFS) sebagai opsi penyimpanan untuk server Transfer Family Anda.
- Jika Anda menentukan nilai direktori logis untuk pengguna Anda, parameter yang Anda gunakan bergantung pada jenis pengguna:
 - Untuk pengguna yang dikelola layanan, berikan nilai direktori logis di `HomeDirectoryMappings`
 - Untuk pengguna penyedia identitas kustom, berikan nilai direktori logis di `HomeDirectoryDetails`.

Important

Kecuali Anda memilih untuk mengoptimalkan kinerja untuk direktori Amazon S3 Anda (saat Anda membuat atau memperbarui server), direktori root harus ada saat startup. Untuk Amazon S3, ini berarti Anda harus sudah membuat objek zero-byte yang diakhiri dengan

garis miring maju (/) untuk membuat folder root. Menghindari masalah ini adalah alasan untuk mempertimbangkan mengoptimalkan kinerja Amazon S3.

Menerapkan direktori logis dan **chroot**

Untuk menggunakan direktori dan chroot fitur logis, Anda harus melakukan hal berikut:

Aktifkan direktori logis untuk setiap pengguna. Lakukan ini dengan menyetel `HomeDirectoryType` parameter LOGICAL saat Anda membuat atau memperbarui pengguna Anda.

```
"HomeDirectoryType": "LOGICAL"
```

chroot

Untuk chroot, buat struktur direktori yang terdiri dari satu Entry dan Target pasangan untuk setiap pengguna. Folder root adalah Entry intinya, dan lokasi di bucket atau sistem file Anda untuk dipetakan. Target

Example for Amazon S3

```
[{"Entry": "/", "Target": "/mybucket/jane"}]
```

Example for Amazon EFS

```
[{"Entry": "/", "Target": "/fs-faa1a123/jane"}]
```

Anda dapat menggunakan jalur absolut seperti pada contoh sebelumnya, atau Anda dapat menggunakan substitusi dinamis untuk nama pengguna dengan `${transfer:UserName}`, seperti pada contoh berikut.

```
[{"Entry": "/", "Target":  
"/mybucket/${transfer:UserName}"}]
```

Dalam contoh sebelumnya, pengguna dikunci ke direktori root mereka dan tidak dapat melintasi lebih tinggi dalam hierarki.

Struktur direktori virtual

Untuk struktur direktori virtual, Anda dapat membuat beberapa Entry Target pasangan, dengan target di mana saja di bucket atau sistem EFS file S3 Anda, termasuk di beberapa bucket atau sistem file, selama pemetaan IAM peran pengguna memiliki izin untuk mengaksesnya.

Dalam contoh struktur virtual berikut, ketika pengguna login ke AWS SFTP, mereka berada di direktori root dengan sub-direktori/pics,, /doc/reporting, dan. /anotherpath/subpath/financials

Note

Kecuali Anda memilih untuk mengoptimalkan kinerja untuk direktori Amazon S3 Anda (saat Anda membuat atau memperbarui server), baik pengguna atau administrator perlu membuat direktori jika belum ada. Menghindari masalah ini adalah alasan untuk mempertimbangkan mengoptimalkan kinerja Amazon S3.

Untuk AmazonEFS, Anda masih memerlukan administrator untuk membuat pemetaan logis atau direktori. /

```
[
{"Entry": "/pics", "Target": "/bucket1/pics"},
{"Entry": "/doc", "Target": "/bucket1/anotherpath/docs"},
{"Entry": "/reporting", "Target": "/reportingbucket/Q1"},
{"Entry": "/anotherpath/subpath/financials", "Target": "/reportingbucket/financials"}]
```

Note

Anda hanya dapat mengunggah file ke folder tertentu yang Anda petakan. Ini berarti bahwa dalam contoh sebelumnya, Anda tidak dapat mengunggah ke /anotherpath atau anotherpath/subpath direktori; hanyaanotherpath/subpath/financials. Anda juga tidak dapat memetakan ke jalur tersebut secara langsung, karena jalur yang tumpang tindih tidak diperbolehkan.

Misalnya, asumsikan bahwa Anda membuat pemetaan berikut:

```
{
  "Entry": "/pics",
  "Target": "/mybucket/pics"
```

```

},
{
  "Entry": "/doc",
  "Target": "/mybucket/mydocs"
},
{
  "Entry": "/temp",
  "Target": "/mybucket"
}

```

Anda hanya dapat mengunggah file ke bucket tersebut. Ketika Anda pertama kali terhubung `sftp`, Anda dijumpai ke direktori root, `/`. Jika Anda mencoba mengunggah file ke direktori itu, unggahan gagal. Perintah berikut menunjukkan urutan contoh:

```

sftp> pwd
Remote working directory: /
sftp> put file
Uploading file to /file
remote open("/file"): No such file or directory

```

Untuk mengunggah ke salah satu direktori/sub-direktori, Anda harus secara eksplisit memetakan jalur ke file. sub-direktori

Untuk informasi selengkapnya tentang mengonfigurasi direktori logis dan chroot untuk pengguna Anda, termasuk AWS CloudFormation templat yang dapat Anda unduh dan gunakan, lihat [Menyederhanakan AWS SFTP Struktur Anda dengan direktori chroot dan logis](#) di Blog Penyimpanan. AWS

Konfigurasikan contoh direktori logis

Dalam contoh ini, kami membuat pengguna dan menetapkan dua direktori logis. Perintah berikut membuat pengguna baru (untuk server Transfer Family yang ada) dengan direktori logis `pics` dan `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

```

Jika **marymajor** adalah pengguna yang sudah ada dan jenis direktori home nya `PATH`, Anda dapat mengubahnya `LOGICAL` dengan perintah yang sama seperti yang sebelumnya.

```
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\"}]"
```

Perhatikan hal berikut:

- Jika direktori `/DOC-EXAMPLE-BUCKET1/pics` dan `/DOC-EXAMPLE-BUCKET2/test/mydocs` belum ada, pengguna (atau administrator) perlu membuatnya.
- Ketika **marymajor** terhubung ke server, dan menjalankan `ls -l` perintah, dia melihat yang berikut:

```
drwxr--r--  1      -      -      0 Mar 17 15:42 doc
drwxr--r--  1      -      -      0 Mar 17 16:04 pics
```

- **marymajor** tidak dapat membuat file atau direktori apa pun pada level ini. Namun, di dalam `pics` dan `doc`, dia dapat menambahkan sub-direktori.
- File yang dia tambahkan `pics` dan `doc` ditambahkan ke jalur Amazon S3 `/DOC-EXAMPLE-BUCKET1/pics` dan `/DOC-EXAMPLE-BUCKET2/test/mydocs` masing-masing.
- Dalam contoh ini, kami menentukan dua ember berbeda untuk mengilustrasikan kemungkinan itu. Namun, Anda dapat menggunakan bucket yang sama untuk beberapa atau semua direktori logis yang Anda tentukan untuk pengguna.

Konfigurasi direktori logis untuk Amazon EFS

Jika server Transfer Family Anda menggunakan Amazon EFS, direktori home untuk pengguna harus dibuat dengan akses baca dan tulis sebelum pengguna dapat bekerja di direktori home logisnya. Pengguna tidak dapat membuat direktori ini sendiri, karena mereka akan kekurangan izin untuk `mkdir` direktori home logis mereka.

Jika direktori home pengguna tidak ada, dan mereka menjalankan `ls` perintah, sistem merespons sebagai berikut:

```
sftp> ls
```



```
remote readdir ("/"): No such file or directory
```

Seorang pengguna dengan akses administratif ke direktori induk perlu membuat direktori home logis pengguna.

AWS Lambda Tanggapan khusus

Anda dapat menggunakan direktori logis dengan fungsi Lambda yang terhubung ke penyedia identitas kustom Anda. Untuk melakukannya, dalam fungsi Lambda Anda, Anda menentukan `HomeDirectoryType` sebagai **LOGICAL**, dan menambahkan `Entry` dan `Target` nilai untuk parameter. `HomeDirectoryDetails` Sebagai contoh:

```
HomeDirectoryType: "LOGICAL"
HomeDirectoryDetails: "[{"Entry": "\\", "Target": "\/DOC-EXAMPLE-BUCKET/
theRealFolder"}]"
```

Kode berikut adalah contoh respons yang berhasil dari panggilan otentikasi Lambda kustom.

```
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

`"Url"`: Baris dikembalikan hanya jika Anda menggunakan metode API Gateway sebagai penyedia identitas kustom Anda.

AWS Transfer Family SFTPkonektor

AWS Transfer Family SFTPkonektor membangun hubungan untuk mengirim file dan pesan antara penyimpanan Amazon dan mitra eksternal, menggunakan SFTP protokol. Anda dapat mengirim file dari Amazon S3 ke tujuan eksternal milik mitra. Anda juga dapat menggunakan SFTP konektor untuk mengambil file dari SFTP server mitra.

Note

Saat ini, SFTP konektor hanya dapat digunakan untuk terhubung ke SFTP server jarak jauh yang menawarkan titik akhir yang dapat diakses internet.

Lihat [AWS Transfer Family SFTPkonektor](#) untuk pengenalan singkat tentang SFTP konektor Transfer Family.

Topik

- [Konfigurasi SFTP konektor](#)
- [Mengirim dan mengambil file dengan menggunakan konektor SFTP](#)
- [Kelola SFTP konektor](#)

Konfigurasi SFTP konektor

Topik ini menjelaskan cara membuat SFTP konektor, algoritme keamanan yang terkait dengannya, cara menyimpan rahasia untuk menyimpan kredensial, detail tentang memformat kunci pribadi, dan instruksi untuk menguji konektor Anda.

Topik

- [Buat SFTP konektor](#)
- [SFTP algoritma konektor](#)
- [Simpan rahasia untuk digunakan dengan SFTP konektor](#)
- [Menghasilkan dan memformat kunci pribadi SFTP konektor](#)
- [Uji SFTP konektor](#)

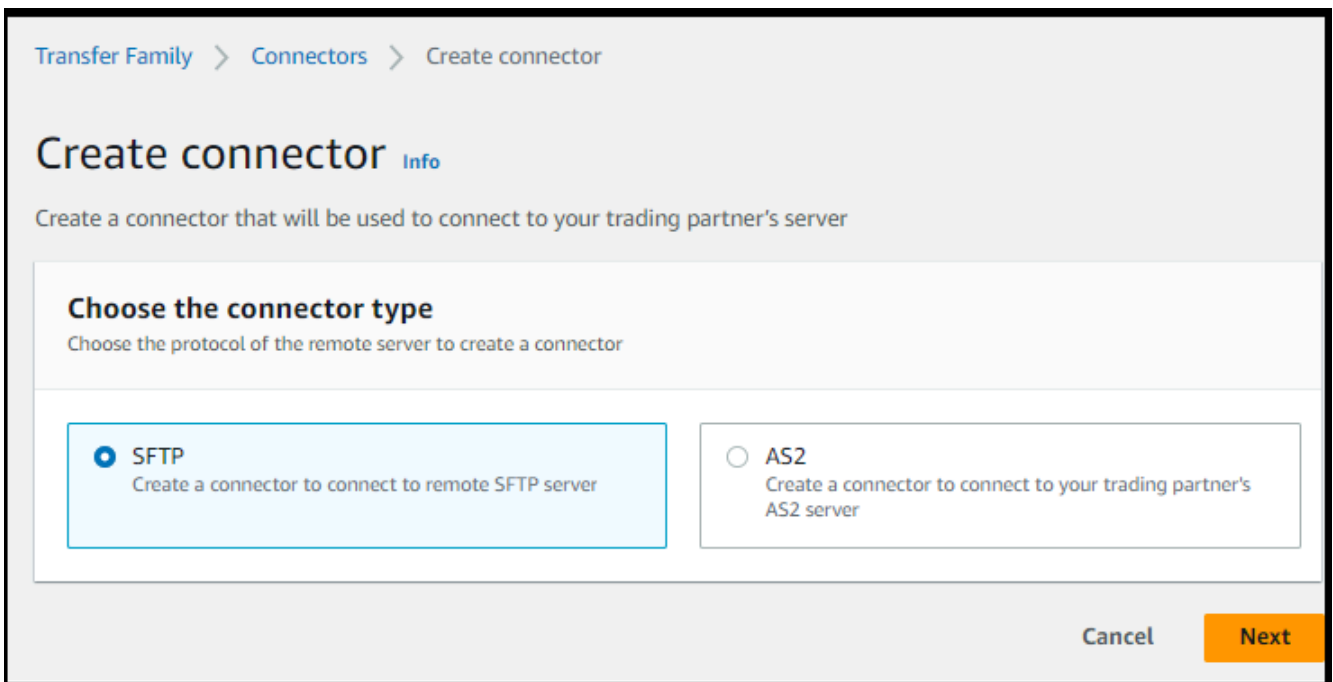
Buat SFTP konektor

Prosedur ini menjelaskan cara membuat SFTP konektor dengan menggunakan AWS Transfer Family konsol atau AWS CLI.

Console

Untuk membuat SFTP konektor

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Di panel navigasi kiri, pilih Konektor, lalu pilih Buat konektor.
3. Pilih SFTP jenis konektor untuk membuat SFTP konektor, lalu pilih Berikutnya.



4. Di bagian Konfigurasi konektor, berikan informasi berikut:
 - Untuk URL, masukkan URL untuk SFTP server jarak jauh. Ini URL harus diformat sebagai `sftp://partner-SFTP-server-url`, misalnya `sftp://AnyCompany.com`.

Note

Secara opsional, Anda dapat memberikan nomor port di URL. Formatnya adalah `sftp://partner-SFTP-server-url:port-number`. Nomor port default (bila tidak ada port yang ditentukan) adalah port 22.

- Untuk peran Access, pilih Amazon Resource Name (ARN) dari peran AWS Identity and Access Management (IAM) yang akan digunakan.
- Pastikan bahwa peran ini menyediakan akses baca dan tulis ke direktori induk dari lokasi file yang digunakan dalam `StartFileTransfer` permintaan.
- Pastikan bahwa peran ini memberikan izin `secretsmanager:GetSecretValue` untuk mengakses rahasia.

 Note

Dalam kebijakan, Anda harus menentukan ARN rahasianya. ARN berisi nama rahasia, tetapi menambahkan nama dengan enam, acak, karakter alfanumerik. An ARN for a secret memiliki format berikut.

```
arn:aws:secretsmanager:region:account-id:secret:aws/
transfer/SecretName-6RandomCharacters
```

- Pastikan peran ini berisi hubungan kepercayaan yang memungkinkan konektor mengakses sumber daya Anda saat melayani permintaan transfer pengguna Anda. Untuk detail tentang membangun hubungan kepercayaan, lihat [Untuk membangun hubungan kepercayaan](#).

Contoh berikut memberikan izin yang diperlukan untuk mengakses **DOC-EXAMPLE-BUCKET** di Amazon S3, dan rahasia yang ditentukan disimpan di Secrets Manager.

```
{
  "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

Untuk peran akses, contoh memberikan akses ke satu rahasia. Namun, Anda dapat menggunakan karakter wildcard, yang dapat menyimpan pekerjaan jika Anda ingin menggunakan kembali IAM peran yang sama untuk beberapa pengguna dan rahasia. Misalnya, pernyataan sumber daya berikut memberikan izin untuk semua rahasia yang memiliki nama yang dimulai dengan `aws/transfer`

```

"Resource": "arn:aws:secretsmanager:region:account-id:secret:aws/transfer/*"

```

Anda juga dapat menyimpan rahasia yang berisi SFTP kredensi Anda di tempat lain. Akun AWS Untuk detail tentang mengaktifkan akses rahasia lintas akun, lihat [Izin untuk AWS Secrets Manager rahasia bagi pengguna di](#) akun lain.

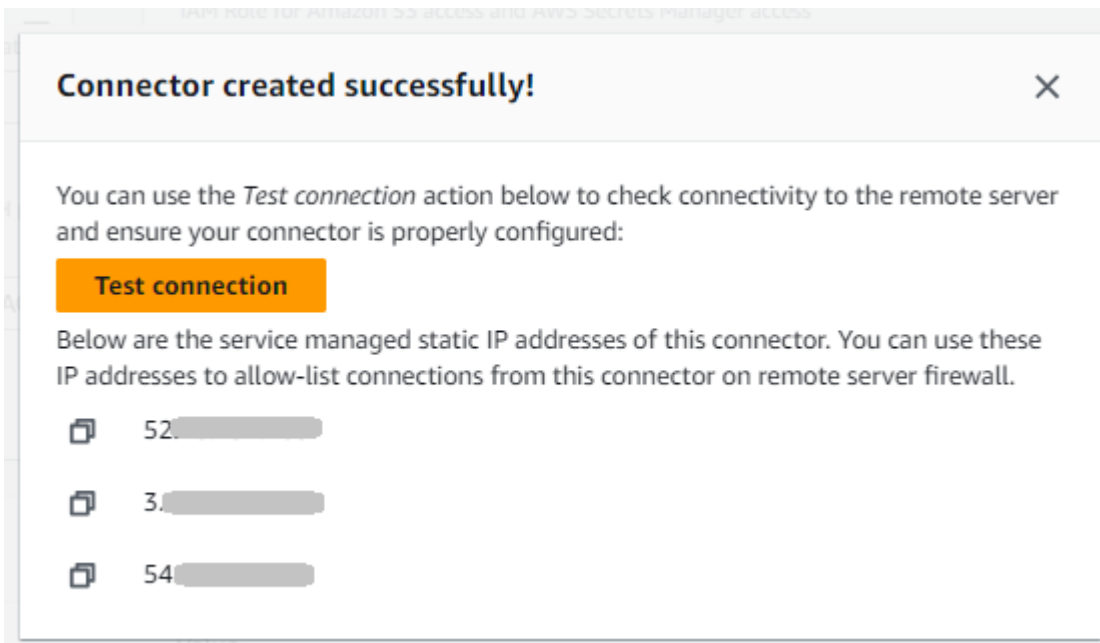
- (Opsional) Untuk peran Logging, pilih IAM peran konektor yang akan digunakan untuk mendorong peristiwa ke CloudWatch log Anda. Contoh kebijakan berikut mencantumkan izin yang diperlukan untuk mencatat peristiwa untuk SFTP konektor.

```
{
  "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. Di bagian SFTPKonfigurasi, berikan informasi berikut:

- Untuk kredensi Connector, dari daftar dropdown, pilih nama rahasia AWS Secrets Manager yang berisi kunci pribadi atau kata SFTP sandi pengguna. Anda harus membuat rahasia dan menyimpannya dengan cara tertentu. Untuk detailnya, lihat [Simpan rahasia untuk digunakan dengan SFTP konektor](#).
- Kunci host tepercaya - Tempel di bagian publik dari kunci host yang digunakan untuk mengidentifikasi server eksternal. Anda dapat menambahkan lebih dari satu kunci, dengan memilih Tambahkan kunci host tepercaya untuk menambahkan kunci tambahan. Anda dapat menggunakan ssh-keyscan perintah terhadap SFTP server untuk mengambil kunci yang diperlukan. Untuk detail tentang format dan jenis kunci host tepercaya yang didukung Transfer Family, lihat [SFTPConnectorConfig](#).

6. (Opsional) Di bagian Tag, untuk Kunci dan Nilai, masukkan satu atau beberapa tag sebagai pasangan nilai kunci.
7. Setelah Anda mengkonfirmasi semua pengaturan Anda, pilih Buat konektor untuk membuat SFTP konektor. Jika konektor berhasil dibuat, layar muncul dengan daftar alamat IP statis yang ditetapkan dan tombol koneksi Uji. Gunakan tombol untuk menguji konfigurasi konektor baru Anda.



Halaman Konektor muncul, dengan ID SFTP konektor baru Anda ditambahkan ke daftar. Untuk melihat detail konektor Anda, lihat [Lihat detail SFTP konektor](#).

CLI

Anda menggunakan [create-connector](#) perintah untuk membuat konektor. Untuk menggunakan perintah ini untuk membuat SFTP konektor, Anda harus memberikan informasi berikut.

- URL Untuk SFTP server jarak jauh. Ini URL harus diformat sebagai `ftp://partner-SFTP-server-url`, misalnya `ftp://AnyCompany.com`.
- Peran akses. Pilih Amazon Resource Name (ARN) dari peran AWS Identity and Access Management (IAM) yang akan digunakan.
 - Pastikan bahwa peran ini menyediakan akses baca dan tulis ke direktori induk dari lokasi file yang digunakan dalam `StartFileTransfer` permintaan.
 - Pastikan bahwa peran ini memberikan izin `secretsmanager:GetSecretValue` untuk mengakses rahasia.

Note

Dalam kebijakan, Anda harus menentukan ARN rahasianya. ARN berisi nama rahasia, tetapi menambahkan nama dengan enam, acak, karakter alfanumerik. An ARN for a secret memiliki format berikut.

```
arn:aws:secretsmanager:region:account-id:secret:aws/
transfer/SecretName-6RandomCharacters
```

- Pastikan peran ini berisi hubungan kepercayaan yang memungkinkan konektor mengakses sumber daya Anda saat melayani permintaan transfer pengguna Anda. Untuk detail tentang membangun hubungan kepercayaan, lihat [Untuk membangun hubungan kepercayaan](#).

Contoh berikut memberikan izin yang diperlukan untuk mengakses *DOC-EXAMPLE-BUCKET* di Amazon S3, dan rahasia yang ditentukan disimpan di Secrets Manager.

```
{
  "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

Untuk peran akses, contoh memberikan akses ke satu rahasia. Namun, Anda dapat menggunakan karakter wildcard, yang dapat menyimpan pekerjaan jika Anda ingin menggunakan kembali IAM peran yang sama untuk beberapa pengguna dan rahasia. Misalnya, pernyataan sumber daya berikut memberikan izin untuk semua rahasia yang memiliki nama yang dimulai dengan. `aws/transfer`

```

"Resource": "arn:aws:secretsmanager:region:account-id:secret:aws/transfer/
*"

```

Anda juga dapat menyimpan rahasia yang berisi SFTP kredensi Anda di tempat lain. Akun AWS Untuk detail tentang mengaktifkan akses rahasia lintas akun, lihat [Izin untuk AWS Secrets Manager rahasia bagi pengguna di](#) akun lain.

- (Opsional) Pilih IAM peran konektor yang akan digunakan untuk mendorong peristiwa ke CloudWatch log Anda. Contoh kebijakan berikut mencantumkan izin yang diperlukan untuk mencatat peristiwa untuk SFTP konektor.

```

{
  "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/*"
    ]
  }]
}
```

- Berikan informasi SFTP konfigurasi berikut.
 - Rahasia AWS Secrets Manager yang berisi kunci pribadi atau kata sandi SFTP pengguna. ARN
 - Bagian publik dari kunci host yang digunakan untuk mengidentifikasi server eksternal. Anda dapat memberikan beberapa kunci host tepercaya jika Anda mau.

Cara termudah untuk memberikan SFTP informasi adalah dengan menyimpannya ke file. Misalnya, salin contoh teks berikut ke file bernama `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="
  ]
}
```

Note

SecretId bisa berupa keseluruhan ARN atau nama rahasia (`example-username-key` dalam daftar sebelumnya).

Kemudian jalankan perintah berikut untuk membuat konektor.

```
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
```

SFTP algoritma konektor

Saat Anda membuat SFTP konektor, algoritme keamanan berikut terpasang ke konektor.

Tipe	Algoritme
SSHcipher	aes256-gcm@openssh.com aes128-gcm@openssh.com aes256-ctr aes192-ctr
SSHmetode pertukaran kunci (KEX)	kurva25519-sha256 curve25519-sha256@libssh.org diffie-hellman-group16-sha512 diffie-hellman-group18-sha512 diffie-hellman-group-exchange-sha256
SSH MAC	hmac-sha2-512-etm@openssh.com hmac-sha2-256-etm@openssh.com hmac-sha2-512 hmac-sha2-256
SSHkunci host	ecdsa-sha2-nistp256 ecdsa-sha2-nistp384 ecdsa-sha2-nistp521 rsa-sha2-512 rsa-sha2-256

Simpan rahasia untuk digunakan dengan SFTP konektor

Anda dapat menggunakan Secrets Manager untuk menyimpan kredensi pengguna untuk konektor AndaSFTP. Ketika Anda membuat rahasia Anda, Anda harus memberikan nama pengguna. Selain itu, Anda dapat memberikan kata sandi, kunci pribadi, atau keduanya. Untuk detailnya, lihat [Kuota untuk konektor SFTP](#).

Note

Ketika Anda menyimpan rahasia di Secrets Manager, Anda Akun AWS dikenakan biaya. Untuk informasi tentang harga, lihat [AWS Secrets Manager Harga](#).

Untuk menyimpan kredensi pengguna di Secrets Manager untuk konektor SFTP

1. Masuk ke AWS Management Console dan buka AWS Secrets Manager konsol di <https://console.aws.amazon.com/secretsmanager/>.
2. Pada panel navigasi kiri, pilih Rahasia.
3. Pada halaman Rahasia, pilih Simpan rahasia baru.
4. Pada halaman Pilih jenis rahasia, untuk tipe Rahasia, pilih Jenis rahasia lainnya.
5. Di bagian pasangan kunci/Nilai, pilih tab kunci/Nilai.
 - Kunci — Masukkan **Username**.
 - nilai — Masukkan nama pengguna yang berwenang untuk terhubung ke server mitra.
6. Jika Anda ingin memberikan kata sandi, pilih Tambahkan baris, dan di bagian pasangan kunci/nilai, pilih tab kunci/Nilai.

Pilih Tambah baris, dan di bagian pasangan kunci/Nilai, pilih tab kunci/Nilai.

- Kunci — Masukkan **Password**.
 - nilai — Masukkan kata sandi untuk pengguna.
7. Jika Anda ingin memberikan kunci pribadi, lihat [Menghasilkan dan memformat kunci pribadi SFTP konektor](#), yang menjelaskan cara memasukkan data kunci pribadi.

 Note

Data kunci pribadi yang Anda masukkan harus sesuai dengan kunci publik yang disimpan untuk pengguna ini di SFTP server jarak jauh.

8. Pilih Berikutnya.
9. Pada halaman Konfigurasi rahasia, masukkan nama dan deskripsi untuk rahasia Anda. Kami menyarankan Anda menggunakan awalan **aws/transfer/** untuk nama tersebut. Misalnya, Anda bisa menyebutkan rahasia Anda **aws/transfer/connector-1**.
10. Pilih Berikutnya, dan kemudian terima default pada halaman Konfigurasi rotasi. Lalu pilih Berikutnya.
11. Pada halaman Review, pilih Store untuk membuat dan menyimpan rahasia.

Menghasilkan dan memformat kunci pribadi SFTP konektor

Detail lengkap untuk menghasilkan public/private key pair dijelaskan dalam [Membuat SSH kunci di macOS, Linux, atau Unix](#)

Sebagai contoh, untuk menghasilkan kunci pribadi untuk digunakan dengan SFTP konektor, perintah contoh berikut menghasilkan jenis kunci yang benar (ganti *key_name* dengan nama file aktual untuk key pair Anda):

```
ssh-keygen -t rsa -b 4096 -m PEM -f key_name -N ""
```

 Note

Saat Anda membuat key pair untuk digunakan dengan SFTP konektor, jangan gunakan frasa sandi. Frasa sandi kosong diperlukan agar SFTP konfigurasi berfungsi dengan benar.

Perintah ini menciptakan sebuah RSA key pair, dengan ukuran kunci 4096 bit. Kunci dihasilkan dalam PEM format lama, yang diperlukan oleh Transfer Family untuk digunakan dengan rahasia SFTP konektor. Kunci disimpan di *key_name* (kunci pribadi) dan *key_name*.pub (kunci publik) di direktori saat ini: yaitu, direktori tempat Anda menjalankan ssh-keygen perintah.

 Note

Transfer Family tidak mendukung SSH format Open (-----BEGIN OPENSSH PRIVATE KEY-----) untuk kunci yang digunakan untuk SFTP konektor Anda. Kuncinya harus dalam PEM format lama (-----BEGIN RSA PRIVATE KEY-----atau-----BEGIN EC PRIVATE KEY-----). Anda dapat menggunakan ssh-keygen alat ini untuk mengonversi kunci Anda, dengan menyediakan -m PEM opsi saat Anda menjalankan perintah.

Setelah Anda membuat kunci, Anda harus memastikan bahwa kunci pribadi diformat dengan karakter baris baru yang disematkan ("\\n") dalam format. JSON

Gunakan perintah untuk mengonversi kunci pribadi Anda yang ada ke format yang benar— JSON format dengan karakter baris baru yang disematkan. Di sini kami memberikan contoh untuk jq dan Powershell. Anda dapat menggunakan alat atau perintah apa pun yang ingin Anda ubah kunci pribadi menjadi JSON format dengan karakter baris baru yang disematkan.

jq command

Contoh ini menggunakan jq perintah, yang tersedia untuk diunduh dari [Download jq](#).

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

Misalnya, jika file kunci pribadi Anda berada di ~/.ssh/my_private_key, perintahnya adalah sebagai berikut.

```
jq -sR . ~/.ssh/my_private_key
```

Ini menghasilkan kunci dalam format yang benar (dengan karakter baris baru yang disematkan) ke output standar.

PowerShell

Jika Anda menggunakan Windows, Anda dapat menggunakan PowerShell untuk mengonversi kunci ke format yang benar. Perintah Powershell berikut mengonversi kunci pribadi ke format yang benar.

```
Get-Content -Raw path-to-private-key-file | ConvertTo-Json
```

Untuk menambahkan data kunci pribadi ke rahasia untuk digunakan dengan SFTP konektor

1. Di konsol Secrets Manager, saat menyimpan jenis rahasia lainnya, pilih tab Plaintext. Teks harus kosong, dengan hanya tanda kurung pembuka dan penutup, {}.
2. Tempel nama pengguna, data kunci pribadi, dan/atau kata sandi Anda menggunakan format berikut. Untuk data kunci pribadi Anda, tempel output dari perintah yang Anda jalankan di langkah 1.

```
{"Username": "SFTP-USER", "Password": "SFTP-USER-PASSWORD", "PrivateKey": "PASTE-PRIVATE-KEY-DATA-HERE"}
```



Jika Anda menempelkan data kunci pribadi dengan benar, Anda akan melihat yang berikut saat memilih tab kunci/Nilai. Perhatikan bahwa data kunci pribadi ditampilkan line-by-line, bukan sebagai string teks kontinu.

Secret value [Info](#)
Retrieve and view the secret value.

Key/value | Plaintext

Secret key	Secret value
Username	SFTP-USER
Password	SFTP-USER-PASSWORD
PrivateKey	-----BEGIN RSA PRIVATE KEY----- MITOMERAMU... g... a... U... G... g... T... a... I... W... I... A... e... 5... 7... H... i... By...

3. Lanjutkan prosedur [Simpan rahasia untuk digunakan dengan SFTP konektor](#) di langkah 8, dan ikuti prosedur itu sampai akhir.

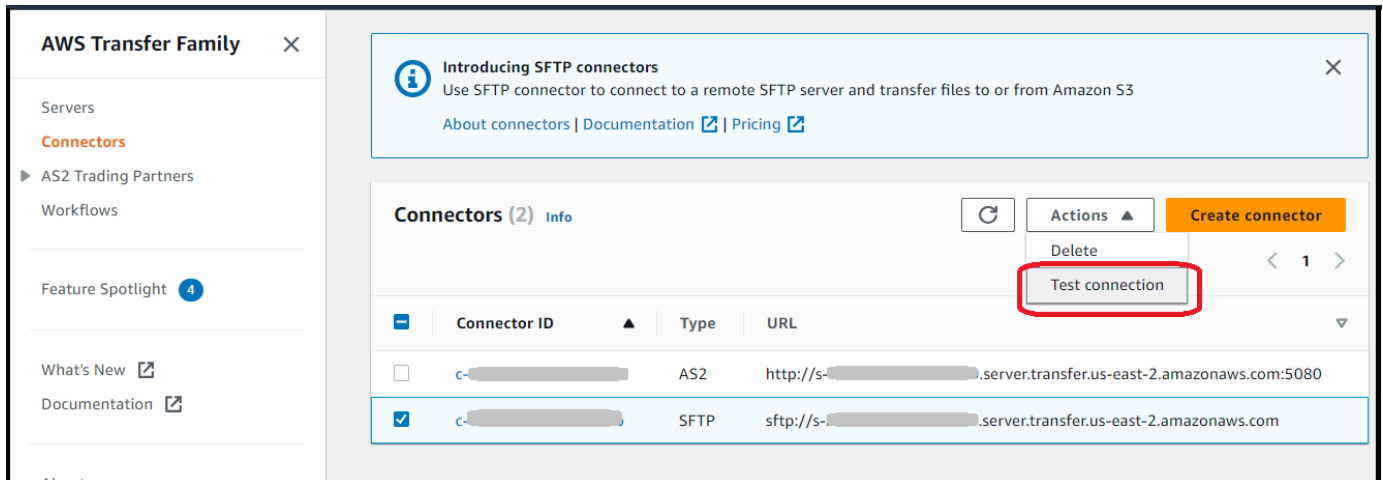
Uji SFTP konektor

Setelah Anda membuat SFTP konektor, kami sarankan Anda mengujinya sebelum mencoba mentransfer file apa pun menggunakan konektor baru Anda.

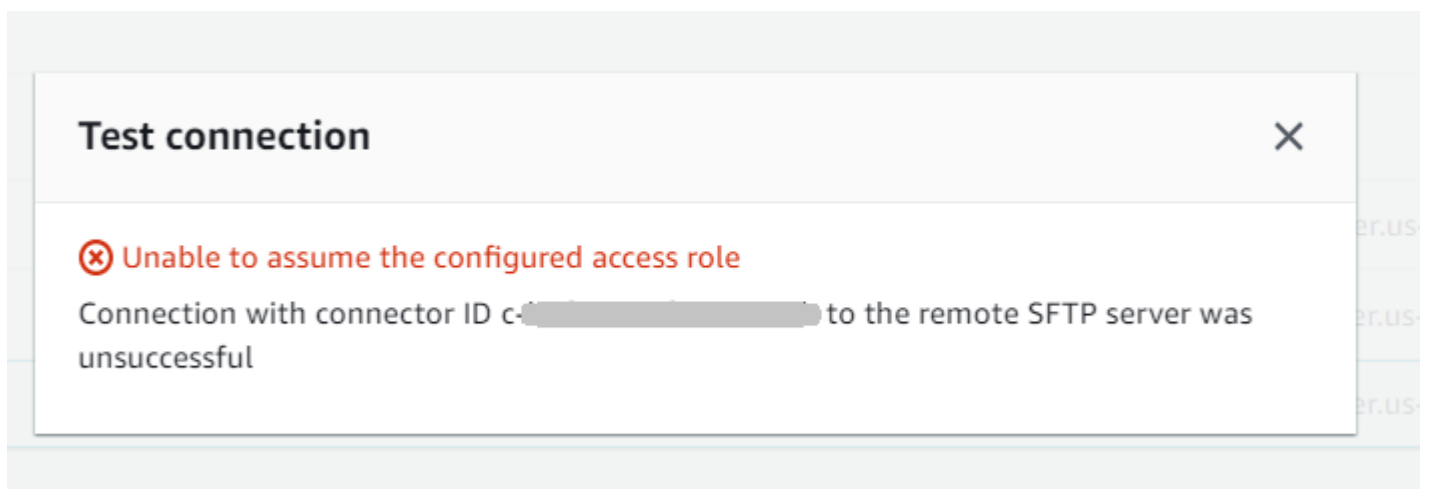
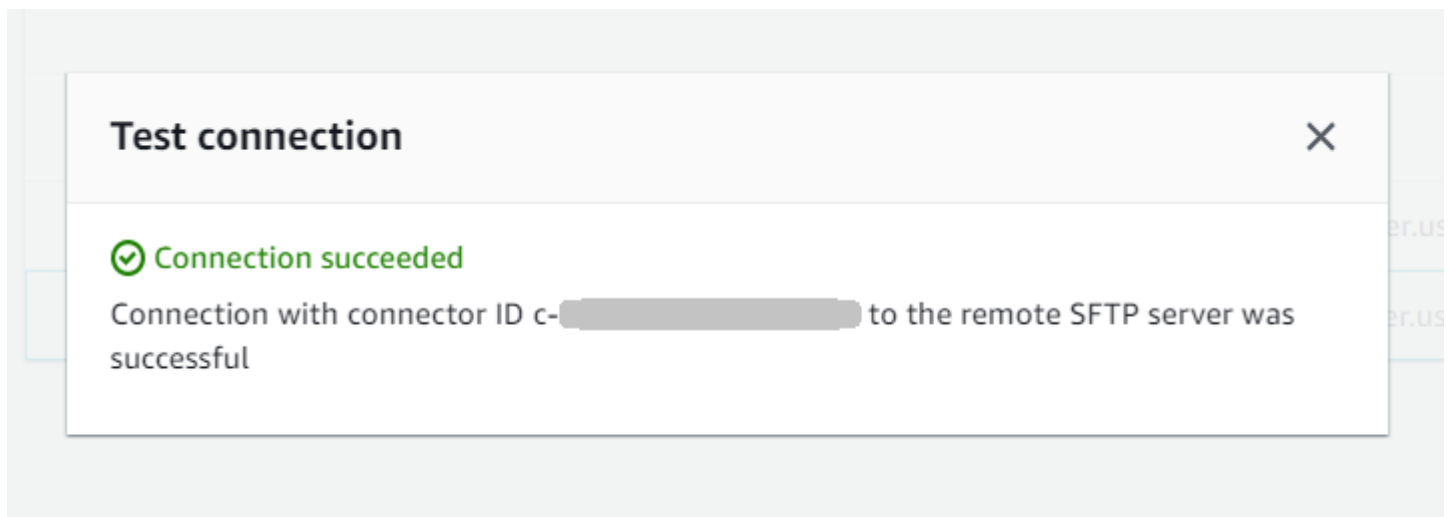
Untuk menguji SFTP konektor

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Di panel navigasi kiri, pilih Konektor, dan pilih konektor.

3. Dari menu Tindakan, pilih Uji koneksi.



Sistem mengembalikan pesan, menunjukkan apakah tes lulus atau gagal. Jika tes gagal, sistem memberikan pesan kesalahan berdasarkan alasan pengujian gagal.



 Note

Untuk menggunakan API untuk menguji konektor Anda, lihat [TestConnectionAPI](#) dokumentasi.

Mengirim dan mengambil file dengan menggunakan konektor SFTP

SFTPkonektor memperluas kemampuan AWS Transfer Family untuk berkomunikasi dengan server jarak jauh baik di cloud maupun di tempat. Anda dapat mengintegrasikan data yang dihasilkan dan disimpan dalam sumber jarak jauh dengan gudang data yang AWS dihosting untuk analitik, aplikasi bisnis, pelaporan, dan audit. Untuk memulai transfer file ke SFTP server jauh, Anda menggunakan [StartFileTransfer](#) API operasi, yang menggunakan SFTP konektor untuk melakukan transfer. Setiap `StartFileTransfer` permintaan dapat berisi 10 jalur berbeda.

Anda dapat memantau transfer file Anda dengan memeriksa log server Anda. Aktivitas konektor dicatat ke aliran log yang memiliki format `aws/transfer/connector-id`, misalnya, `aws/transfer/c-1234567890abcdef0`. Jika Anda tidak melihat log apa pun untuk konektor Anda, pastikan Anda telah menentukan peran logging dengan izin yang benar untuk konektor Anda.

Untuk detail tentang membuat konektor, lihat [Konfigurasi SFTP konektor](#).

Untuk mengirim dan mengambil file dengan menggunakan SFTP konektor, Anda menggunakan perintah `start-file-transfer` AWS Command Line Interface (AWS CLI). Anda menentukan parameter berikut, tergantung apakah Anda mengirim file (transfer keluar) atau menerima file (transfer masuk).

- Transfer keluar
 - `send-file-paths` berisi dari satu hingga sepuluh jalur file sumber, untuk file yang akan ditransfer ke SFTP server mitra.
 - `remote-directory-path` adalah jalur jarak jauh untuk mengirim file ke SFTP server pelanggan.
- Transfer masuk
 - `retrieve-file-paths` berisi dari satu hingga sepuluh jalur jarak jauh. Setiap jalur menentukan lokasi untuk mentransfer file dari server mitra ke SFTP server Transfer Family Anda.

- `local-directory-path` adalah lokasi Amazon S3 (bucket dan awalan opsional) tempat file Anda disimpan.

Untuk mengirim file, Anda menentukan `remote-directory-path` parameter `send-file-paths` dan. Anda dapat menentukan hingga 10 file untuk `send-file-paths` parameter. Contoh perintah berikut mengirimkan file bernama `/DOC-EXAMPLE-SOURCE-BUCKET/file1.txt` dan `/DOC-EXAMPLE-SOURCE-BUCKET/file2.txt`, terletak di penyimpanan Amazon S3, ke `/tmp` direktori di server mitra Anda. SFTP Untuk menggunakan perintah contoh ini, ganti `DOC-EXAMPLE-SOURCE-BUCKET` dengan bucket Anda sendiri.

```
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
```

Untuk menerima file, Anda menentukan `local-directory-path` parameter `retrieve-file-paths` dan. Contoh berikut mengambil file `/my/remote/file1.txt` dan `/my/remote/file2.txt` di SFTP server mitra, dan menemukannya di lokasi Amazon S3 `/DOC-EXAMPLE-BUCKET/prefix`. Untuk menggunakan perintah contoh ini, ganti `user input placeholders` dengan informasi Anda sendiri.

```
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
```

Contoh sebelumnya menentukan jalur absolut di SFTP server. Anda juga dapat menggunakan jalur relatif: yaitu jalur yang relatif terhadap direktori home SFTP pengguna. Misalnya, jika SFTP pengguna `marymajor` dan direktori home mereka di SFTP server adalah `/users/marymajor/`, perintah berikut mengirimkan `/DOC-EXAMPLE-SOURCE-BUCKET/file1.txt` ke `/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
```

Kelola SFTP konektor

Topik ini menjelaskan cara melihat dan memperbarui SFTP konektor, dan mencantumkan kuota yang relevan untuk SFTP konektor.

Topik

- [Perbarui SFTP konektor](#)
- [Lihat detail SFTP konektor](#)
- [Kuota untuk konektor SFTP](#)

Perbarui SFTP konektor

Untuk mengubah nilai parameter yang ada untuk konektor Anda, Anda dapat menjalankan `update-connector` perintah. Perintah berikut memperbarui rahasia untuk konektor *connector-id*, di Wilayah *region-id* ke *secret-ARN*. Untuk menggunakan contoh perintah ini, ganti *user input placeholders* dengan informasi Anda sendiri.

```
aws transfer update-connector --sftp-config '{"UserSecretId":"secret-ARN"}' \
  --connector-id connector-id --region region-id
```

Lihat detail SFTP konektor

Anda dapat menemukan daftar detail dan properti untuk SFTP konektor di AWS Transfer Family konsol.

Untuk melihat detail konektor

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Di panel navigasi kiri, pilih Konektor.
3. Pilih pengenal di kolom Connector ID untuk melihat halaman detail untuk konektor yang dipilih.

Anda dapat mengubah properti untuk SFTP konektor dengan memilih Edit pada halaman detail konektor.

[Transfer Family](#) > [Connectors](#) > c-
C- Delete

Connector configuration Info Edit

URL sftp://	Access role transfer-s3	Logging role role

SFTP configuration Edit

Connector credentials arn:aws:secretsmanager:us-	Trusted host keys 1. SHA256

Egress IP details Info

Service managed static IP addresses of this connector

	52.
	3.
	54.

Tags (0) Manage tags

Key

Value

Note

Anda bisa mendapatkan banyak informasi ini, meskipun dalam format yang berbeda, dengan menjalankan perintah AWS Command Line Interface (AWS CLI) berikut. Untuk menggunakan contoh perintah ini, ganti *user input placeholders* dengan informasi Anda sendiri.

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

Untuk informasi selengkapnya, silakan lihat [DescribeConnector](#) dalam API referensi.

Kuota untuk konektor SFTP

Kuota berikut tersedia untuk SFTP konektor. Kuota untuk AS2 konektor dijelaskan dalam [AS2 kuota dan batasan](#). Untuk meminta kenaikan kuota yang dapat disesuaikan, lihat [Layanan AWS kuota](#) di Referensi Umum AWS

SFTP kuota konektor

Nama	Default	Dapat disesuaikan
Transaksi koneksi uji maksimum per detik (TPS)	1 permintaan per detik, per akun	Tidak
Maksimum StartFile Transfer TPS	3 permintaan per detik, per akun	Ya
Ukuran antrian maksimum untuk transfer file yang tertunda	1000	Tidak
Ukuran maksimum file	50 gibibyte (GiB)	Tidak
Waktu transfer maksimum per file	6 jam	Tidak
Waktu tunggu permintaan maksimum per file	6 jam	Tidak
Durasi minimum AccessRole atau LoggingRole sesi	60 menit	Tidak
Transfer file bersamaan maksimum	1 transfer file bersamaan per konektor	Tidak
Jumlah maksimum permintaan transfer file per detik per akun	3	Ya
Jumlah maksimum konektor per akun (keduanya SFTP dan	100	Ya

Nama	Default	Dapat disesuaikan
AS2 konektor berkontribusi pada hitungan ini)		
Bandwidth maksimum untuk konektor per akun (keduanya SFTP dan AS2 konektor berkontribusi pada nilai ini)	50 MBps	Tidak

Untuk menyimpan kredensi untuk SFTP konektor, ada kuota yang terkait dengan setiap rahasia Secrets Manager. Jika Anda menggunakan rahasia yang sama untuk menyimpan beberapa jenis kunci, untuk berbagai tujuan, Anda mungkin menemukan kuota ini.

- Total panjang untuk satu rahasia: 12.000 karakter
- Panjang maksimum **Password** string: 1024 karakter
- Panjang maksimum **PrivateKey** string: 8192 karakter
- Panjang maksimum **Username** string: 100 karakter

AWS Transfer Family untuk AS2

Pernyataan Penerapan 2 (AS2) adalah spesifikasi transmisi file RFC yang ditentukan yang mencakup perlindungan pesan dan mekanisme verifikasi yang kuat. AS2Protokol sangat penting untuk alur kerja dengan persyaratan kepatuhan yang bergantung pada perlindungan data dan fitur keamanan yang dibangun ke dalam protokol.

Note

AS2 untuk Transfer Family bersertifikat [Drummond](#).

Pelanggan di industri seperti ritel, ilmu hayati, manufaktur, layanan keuangan, dan utilitas yang bergantung pada AS2 rantai pasokan, logistik, dan alur kerja pembayaran dapat menggunakan AWS Transfer Family AS2 titik akhir untuk bertransaksi dengan aman dengan mitra bisnis mereka. Data yang ditransaksikan dapat diakses secara native AWS untuk pemrosesan, analisis, dan pembelajaran mesin. Data ini juga tersedia untuk integrasi dengan perencanaan sumber daya perusahaan (ERP) dan manajemen hubungan pelanggan (CRM) sistem yang berjalan AWS. Dengan AS2, pelanggan dapat menjalankan transaksi business-to-business (B2B) mereka dalam skala besar AWS sambil mempertahankan integrasi dan kepatuhan mitra bisnis yang ada.

Jika Anda adalah pelanggan Transfer Family yang ingin bertukar file dengan mitra yang memiliki server AS2 berkemampuan dikonfigurasi, penyiapannya melibatkan pembuatan satu key pair publik-pribadi untuk enkripsi dan satu lagi untuk menandatangani dan menukar kunci publik dengan mitra.

Melindungi AS2 muatan dalam perjalanan biasanya melibatkan penggunaan Sintaks Pesan Kriptografi (CMS) dan umumnya menggunakan enkripsi dan tanda tangan digital untuk memberikan perlindungan data dan otentikasi rekan. Payload tanggapan Pemberitahuan Disposisi Pesan (MDN) yang ditandatangani memberikan verifikasi (non-penolakan) bahwa pesan diterima dan berhasil didekripsi.

Transportasi CMS muatan dan MDN respons ini terjadi. HTTP

Note

HTTPSAS2Titik akhir server saat ini tidak didukung. TLS penghentian saat ini menjadi tanggung jawab pelanggan.

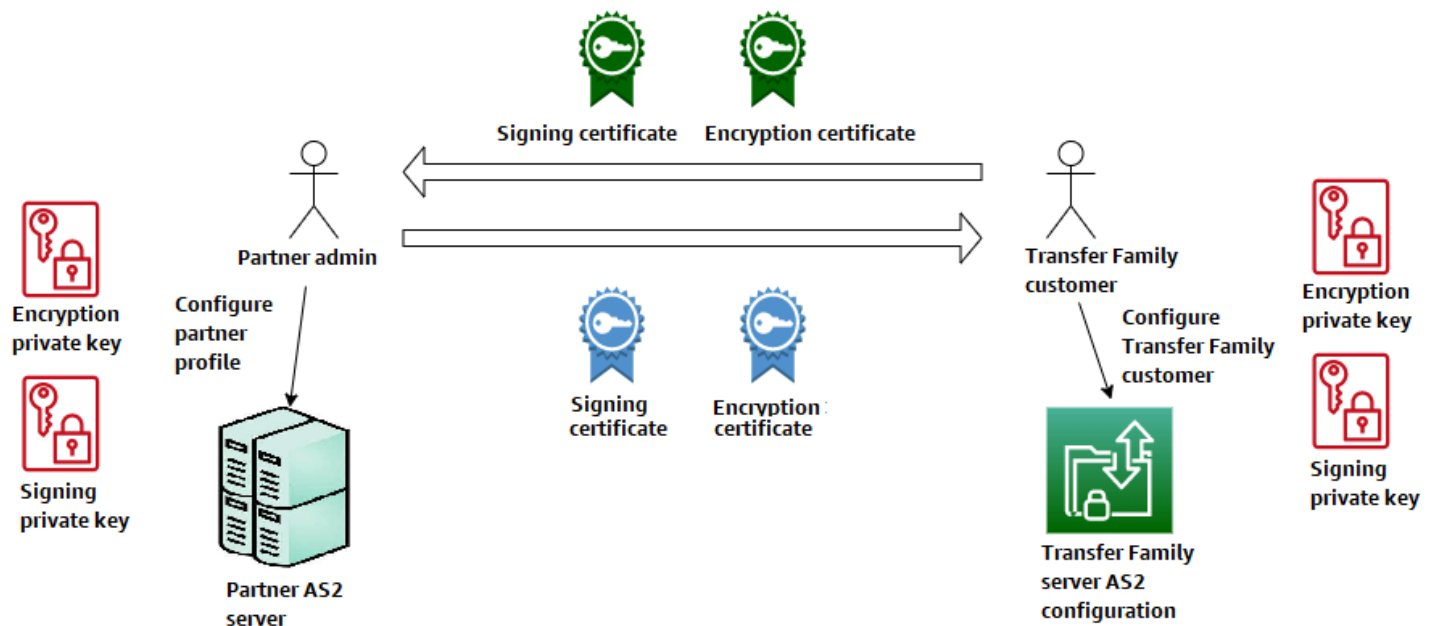
Untuk detail, step-by-step panduan pengaturan konfigurasi Applicability Statement 2 (AS2), lihat tutorialnya,. [Menyiapkan AS2 konfigurasi](#)

Topik

- [Kasus penggunaan AS2](#)
- [Mengkonfigurasi AS2](#)
- [Konfigurasi AS2 konektor](#)
- [Kelola AS2 mitra](#)
- [Mengirim dan menerima AS2 pesan](#)
- [Memantau penggunaan AS2](#)

Kasus penggunaan AS2

Jika Anda adalah AWS Transfer Family pelanggan yang ingin bertukar file dengan mitra yang memiliki AS2 server yang dikonfigurasi, bagian paling kompleks dari pengaturan melibatkan pembuatan satu public-private key pair untuk enkripsi dan satu lagi untuk menandatangani dan bertukar kunci publik dengan mitra.



Pertimbangkan variasi berikut untuk digunakan AWS Transfer Family dengan AS2.

 Note

Mitra dagang adalah mitra yang terkait dengan profil mitra tersebut.
Semua penyebutan MDN dalam tabel berikut berasumsi ditandatangani MDNs.

Kasus penggunaan AS2

Kasus penggunaan khusus masuk

- Transfer AS2 pesan terenkripsi dari mitra dagang ke server Transfer Family.

Dalam kasus ini, lakukan hal berikut:

1. Buat profil untuk mitra dagang Anda dan diri Anda sendiri.
2. Buat server Transfer Family yang menggunakan AS2 protokol.
3. Buat perjanjian dan tambahkan ke server Anda.
4. Impor sertifikat dengan kunci pribadi dan tambahkan ke profil Anda, lalu impor kunci publik ke profil mitra Anda untuk enkripsi.
5. Setelah Anda memiliki barang-barang ini, kirimkan kunci publik untuk sertifikat Anda ke mitra dagang Anda.

Sekarang pasangan Anda dapat mengirimi Anda pesan terenkripsi dan Anda dapat mendekripsi dan menyimpannya di ember Amazon S3 Anda.

- Transfer AS2 pesan terenkripsi dari mitra dagang ke server Transfer Family dan tambahkan penandatanganan.

Dalam skenario ini, Anda masih hanya melakukan transfer masuk, tetapi sekarang Anda ingin pasangan Anda menandatangani pesan yang mereka kirim. Dalam hal ini, impor kunci publik penandatanganan mitra dagang (sebagai sertifikat penandatanganan yang ditambahkan ke profil mitra Anda).

- Transfer AS2 pesan terenkripsi dari mitra dagang ke server Transfer Family dan tambahkan penandatanganan dan pengiriman tanggapan. MDN

Dalam skenario ini, Anda masih hanya melakukan transfer masuk, tetapi sekarang, selain menerima muatan yang ditandatangani, mitra dagang Anda ingin menerima tanggapan yang ditandatangani MDN.

1. Impor kunci penandatanganan publik dan pribadi Anda (sebagai sertifikat penandatanganan ke profil Anda).
2. Kirim kunci penandatanganan publik ke mitra dagang Anda.

Kasus penggunaan khusus keluar

- Transfer AS2 pesan terenkripsi dari server Transfer Family ke mitra dagang.

Kasus ini mirip dengan kasus penggunaan transfer inbound-only, kecuali bahwa alih-alih menambahkan perjanjian ke AS2 server Anda, Anda membuat konektor. Dalam hal ini, Anda mengimpor kunci publik mitra dagang Anda ke profil mereka.

- Transfer AS2 pesan terenkripsi dari server Transfer Family ke mitra dagang dan tambahkan penandatanganan.

Anda masih hanya melakukan transfer keluar, tetapi sekarang mitra dagang Anda ingin Anda menandatangani pesan yang Anda kirim kepada mereka.

1. Impor kunci pribadi penandatanganan Anda (sebagai sertifikat penandatanganan yang ditambahkan ke profil Anda).
 2. Kirimkan kunci publik kepada mitra dagang Anda.
- Transfer AS2 pesan terenkripsi dari server Transfer Family ke mitra dagang dan tambahkan penandatanganan dan kirim MDN tanggapan.

Anda masih hanya melakukan transfer keluar, tetapi sekarang, selain mengirim muatan yang ditandatangani, Anda ingin menerima MDN tanggapan yang ditandatangani dari mitra dagang Anda.

1. Mitra dagang Anda mengirimkan Anda kunci penandatanganan publik mereka.
2. Impor kunci publik mitra dagang Anda (sebagai sertifikat penandatanganan yang ditambahkan ke profil mitra Anda).

Kasus penggunaan masuk dan keluar

- Transfer AS2 pesan terenkripsi di kedua arah antara server Transfer Family dan mitra dagang.

Dalam kasus ini, lakukan hal berikut:

1. Buat profil untuk mitra dagang Anda dan diri Anda sendiri.
2. Buat server Transfer Family yang menggunakan AS2 protokol.
3. Buat perjanjian dan tambahkan ke server Anda.
4. Buat konektor.
5. Impor sertifikat dengan kunci pribadi dan tambahkan ke profil Anda, lalu impor kunci publik ke profil mitra Anda untuk enkripsi.
6. Terima kunci publik dari mitra dagang Anda dan tambahkan ke profil mereka untuk enkripsi.
7. Setelah Anda memiliki barang-barang ini, kirimkan kunci publik untuk sertifikat Anda ke mitra dagang Anda.

Sekarang Anda dan mitra dagang Anda dapat bertukar pesan terenkripsi, dan Anda berdua dapat mendekripsi mereka. Anda dapat menyimpan pesan yang Anda terima di bucket Amazon S3 Anda, dan pasangan Anda dapat mendekripsi dan menyimpan pesan yang Anda kirim kepada mereka.

- Transfer AS2 pesan terenkripsi di kedua arah antara server Transfer Family dan mitra dagang dan tambahkan penandatanganan.

Sekarang Anda dan pasangan Anda ingin pesan yang ditandatangani.

1. Impor kunci pribadi penandatanganan Anda (sebagai sertifikat penandatanganan yang ditambahkan ke profil Anda).
 2. Kirimkan kunci publik kepada mitra dagang Anda.
 3. Impor kunci publik penandatanganan mitra dagang Anda dan tambahkan ke profil mereka.
- Transfer AS2 pesan terenkripsi di kedua arah antara server Transfer Family dan mitra dagang dan tambahkan penandatanganan dan kirim MDN tanggapan.

Sekarang, Anda ingin menukar muatan yang ditandatangani, dan Anda dan mitra dagang Anda menginginkan MDN tanggapan.

1. Mitra dagang Anda mengirimkan Anda kunci penandatanganan publik mereka.
2. Impor kunci publik mitra dagang Anda (sebagai sertifikat penandatanganan ke profil mitra Anda).

3. Kirim kunci publik Anda ke mitra dagang Anda.

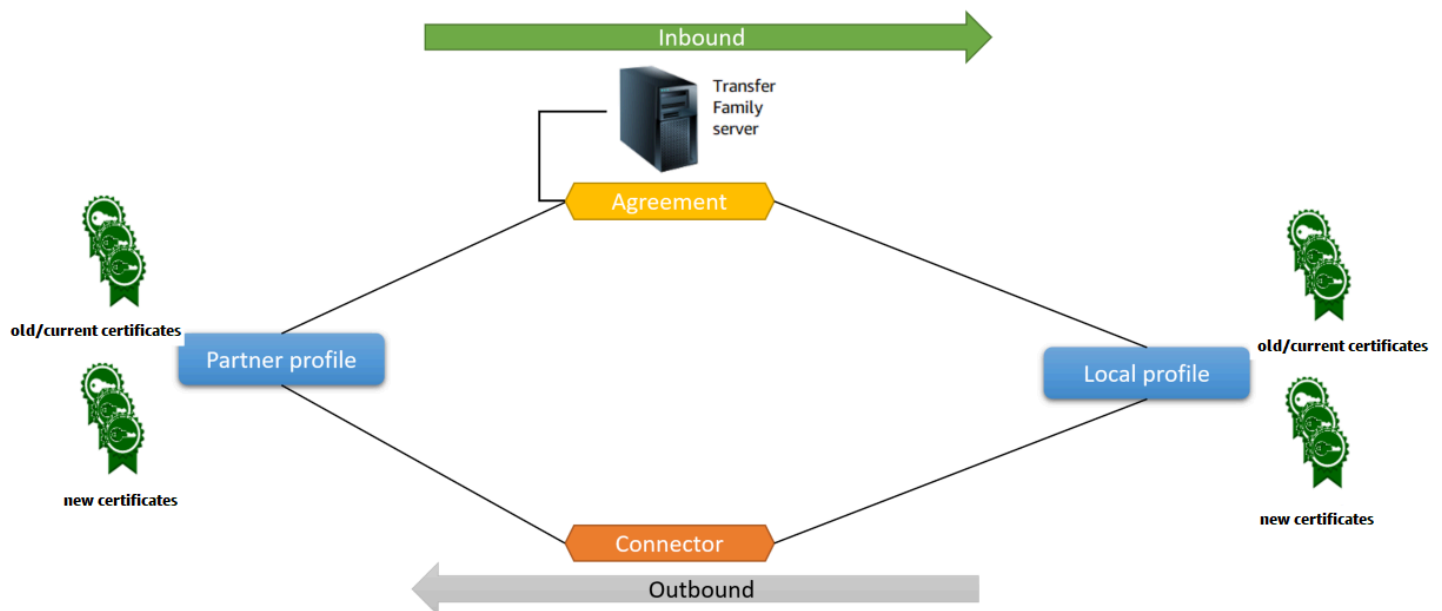
Mengkonfigurasi AS2

Untuk membuat AS2 server yang diaktifkan, Anda juga harus menentukan komponen berikut:

- **Perjanjian** — Perjanjian mitra dagang bilateral, atau kemitraan, menentukan hubungan antara kedua pihak yang bertukar pesan (file). Untuk menentukan perjanjian, Transfer Family menggabungkan server, profil lokal, profil mitra, dan informasi sertifikat. Transfer Family AS2 - proses masuk menggunakan perjanjian.
- **Sertifikat** — Sertifikat kunci publik (X.509) digunakan dalam AS2 komunikasi untuk enkripsi dan verifikasi pesan. Sertifikat juga digunakan untuk titik akhir konektor.
- **Profil lokal dan profil mitra** — Profil lokal mendefinisikan organisasi atau “pesta” lokal (AS2-enabled Transfer Family server). Demikian pula, profil mitra mendefinisikan organisasi mitra jarak jauh, eksternal untuk Transfer Family.

Meskipun tidak diperlukan untuk semua AS2 server yang diaktifkan, untuk transfer keluar, Anda memerlukan konektor. Konektor menangkap parameter untuk koneksi keluar. Konektor diperlukan untuk mengirim file ke eksternal pelanggan, non AWS server.

Diagram berikut menunjukkan hubungan antara AS2 objek yang terlibat dalam proses masuk dan keluar.



Untuk AS2 konfigurasi end-to-end contoh, lihat [Menyiapkan AS2 konfigurasi](#).

Topik

- [Membuat AS2 server menggunakan konsol Transfer Family](#)
- [Gunakan template untuk membuat AS2 tumpukan Transfer Family demo](#)
- [AS2 konfigurasi dan batasan](#)
- [AS2 fitur dan kemampuan](#)

Membuat AS2 server menggunakan konsol Transfer Family

Prosedur ini menjelaskan cara membuat server AS2 yang diaktifkan dengan menggunakan konsol Transfer Family. Jika Anda ingin menggunakan AWS CLI sebagai gantinya, lihat [the section called “Langkah 2: Buat server Transfer Family yang menggunakan AS2 protokol”](#).

Untuk membuat server AS2 yang diaktifkan

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Di panel navigasi kiri, pilih Server, lalu pilih Buat server.
3. Pada halaman Pilih protokol, pilih AS2(Pernyataan Penerapan 2), lalu pilih Berikutnya.

Choose protocols

Select the protocols you want to enable [Info](#)

Choose one or more file transfer protocols over which clients can connect to your server's endpoint

- ☐ SFTP (SSH File Transfer Protocol) - file transfer over Secure Shell
- ☒ AS2 (Applicability Statement 2) - messaging protocol for exchanging business-to-business data [Info](#)
- ☐ FTPS (File Transfer Protocol Secure) - file transfer protocol with TLS encryption
- ☐ FTP (File Transfer Protocol) - unencrypted file transfer protocol

Cancel Next

4. Pada halaman Pilih penyedia identitas, pilih Berikutnya.

Note

Untuk AS2, Anda tidak dapat memilih penyedia identitas karena otentikasi dasar tidak didukung untuk AS2 protokol. Sebagai gantinya, Anda mengontrol akses melalui grup keamanan virtual private cloud (VPC).

5. Pada halaman Pilih titik akhir, lakukan hal berikut:

Choose an endpoint

Endpoint configuration [Info](#)

Endpoint type
Select whether the endpoint will be publicly accessible or hosted inside your VPC

☐ Publicly accessible
Accessible over the internet

☒ **VPC hosted** [Info](#)
Access controlled using Security Groups

Access [Info](#)

☒ **Internal**

☐ Internet Facing

VPC
Select a VPC ID

Select a VPC ID ▼ [Create a VPC](#)

FIPS Enabled
Select whether the endpoint should comply with Federal Information Processing Standards (FIPS)

☐ FIPS Enabled endpoint

[Cancel](#) [Previous](#) [Next](#)

- a. Untuk jenis Endpoint, pilih VPCHost untuk meng-host endpoint server Anda. Untuk informasi tentang menyiapkan titik akhir yang VPC di-host, lihat. [Buat server di cloud pribadi virtual](#)

 Note

Titik akhir yang dapat diakses publik tidak didukung untuk protokol. AS2 Untuk membuat VPC titik akhir Anda dapat diakses melalui internet, pilih Internet Facing di bawah Access, dan kemudian berikan alamat IP Elastic Anda.

b. Untuk Access, pilih salah satu opsi berikut:

- Internal — Pilih opsi ini untuk menyediakan akses dari dalam lingkungan Anda VPC dan lingkungan VPC yang terhubung, seperti pusat data lokal di atas AWS Direct Connect atau. VPN
- Menghadapi Internet — Pilih opsi ini untuk menyediakan akses melalui internet dan dari dalam lingkungan Anda VPC dan VPC yang terhubung, seperti pusat data lokal di atas AWS Direct Connect atau. VPN

Jika Anda memilih Internet Facing, berikan alamat IP Elastis Anda saat diminta.

- c. Untuk VPC, pilih yang sudah ada VPC atau pilih Buat VPC untuk membuat yang baruVPC.
- d. Untuk FIPSDiaktifkan, biarkan kotak centang FIPSEnabled endpoint tetap bersih.

 Note

FIPSEndpoint yang diaktifkan tidak didukung untuk protokol. AS2

e. Pilih Berikutnya.

6. Pada halaman Pilih domain, pilih Amazon S3 untuk menyimpan dan mengakses file Anda sebagai objek dengan menggunakan protokol yang dipilih.

Pilih Berikutnya.

7. Pada halaman Konfigurasi detail tambahan, pilih pengaturan yang Anda butuhkan.

 Note

Jika Anda mengonfigurasi protokol lain bersamaAS2, semua pengaturan detail tambahan berlaku. Namun, untuk AS2 protokol, satu-satunya pengaturan yang berlaku adalah yang ada di bagian CloudWatch logging dan Tags.

Meskipun pengaturan peran CloudWatch logging bersifat opsional, kami sangat menyarankan untuk mengaturnya sehingga Anda dapat melihat status pesan dan memecahkan masalah konfigurasi.

8. Pada halaman Tinjau dan buat, tinjau pilihan Anda untuk memastikannya benar.

- Jika Anda ingin mengedit pengaturan apa pun, pilih Edit di samping langkah yang ingin Anda ubah.

Note

Jika Anda mengedit langkah, kami sarankan Anda meninjau setiap langkah setelah langkah yang Anda pilih untuk diedit.

- Jika Anda tidak memiliki perubahan, pilih Buat server untuk membuat server Anda. Anda dibawa ke halaman Server, ditampilkan berikut, di mana server baru Anda terdaftar.

Servers (1)							
					Actions ▼	Add user	Create server
< 1 >							
☐	Hostname ▲	Server ID ▼	State ▼	Service managed users ▼		Endpoint type ▼	Domain ▼
☐	-	s-	Starting	-		VPC	Amazon S3

Diperlukan waktu beberapa menit sebelum status server baru Anda berubah menjadi Online. Pada saat itu, server Anda dapat melakukan operasi file untuk pengguna Anda.

Gunakan template untuk membuat AS2 tumpukan Transfer Family demo

Kami menyediakan AWS CloudFormation template mandiri untuk membuat server Transfer Family AS2 yang diaktifkan dengan cepat. Template mengonfigurasi server dengan VPC titik akhir Amazon publik, sertifikat, profil lokal dan mitra, perjanjian, dan konektor.

Sebelum menggunakan template ini, perhatikan hal berikut:

- Jika Anda membuat tumpukan dari template ini, Anda akan ditagih untuk AWS sumber daya yang digunakan.
- Template membuat beberapa sertifikat dan menempatkannya AWS Secrets Manager untuk menyimpannya dengan aman. Anda dapat menghapus sertifikat ini dari Secrets Manager jika mau, karena Anda dikenakan biaya untuk menggunakan layanan ini. Menghapus sertifikat ini di Secrets

Manager tidak akan menghapusnya dari server Transfer Family. Oleh karena itu, fungsionalitas tumpukan demo tidak terpengaruh. Namun, untuk sertifikat yang akan Anda gunakan dengan AS2 server produksi, Anda mungkin ingin menggunakan Secrets Manager untuk mengelola dan memutar sertifikat tersimpan secara berkala.

- Kami menyarankan Anda menggunakan template sebagai basis saja, dan terutama untuk tujuan demonstrasi. Jika Anda ingin menggunakan tumpukan demo ini dalam produksi, kami sarankan Anda memodifikasi YAML kode template untuk membuat tumpukan yang lebih kuat. Misalnya, buat sertifikat tingkat produksi, dan buat AWS Lambda fungsi yang dapat Anda gunakan dalam produksi.

Untuk membuat server Transfer Family AS2 -enabled dari template CloudFormation

1. Buka AWS CloudFormation konsol di <https://console.aws.amazon.com/cloudformation>.
2. Di panel navigasi sebelah kiri, pilih Tumpukan.
3. Pilih Buat tumpukan, lalu pilih Dengan sumber daya baru (standar).
4. Di bagian Prasyarat - Siapkan templat, pilih Template sudah siap.
5. Salin tautan ini, [templat AS2 demo](#), dan tempel ke bidang Amazon S3 URL.
6. Pilih Berikutnya.
7. Pada halaman Tentukan detail tumpukan, beri nama tumpukan Anda, lalu tentukan parameter berikut:
 - Di bawah AS2, masukkan nilai untuk AS2ID Lokal dan AS2ID Mitra, atau terima default, `local` dan `partner`, masing-masing.
 - Di bawah Jaringan, masukkan nilai untuk CIDRIP masuk grup Keamanan, atau terima default, `0.0.0.0/0`.

 Note

Nilai ini, dalam CIDR format, menentukan alamat IP mana yang diizinkan untuk lalu lintas masuk ke server. AS2 Nilai default, `0.0.0.0/0`, memungkinkan semua alamat IP.

- Di bawah Umum, masukkan nilai untuk Awalan, atau terima default, `transfer-as2`. Awalan ini ditempatkan sebelum nama sumber daya apa pun yang dibuat oleh tumpukan. Misalnya, jika Anda menggunakan awalan default, bucket Amazon S3 Anda diberi nama. `transfer-as2-TransferS3BucketName`

8. Pilih Berikutnya. Pada halaman Configure stack options, pilih Next lagi.
9. Tinjau detail tumpukan yang Anda buat, lalu pilih Buat tumpukan.

 Note

Di bagian bawah halaman, di bawah Kemampuan, Anda harus mengakui bahwa AWS CloudFormation mungkin membuat AWS Identity and Access Management (IAM) sumber daya.

Setelah tumpukan dibuat, Anda dapat mengirim AS2 pesan pengujian dari server mitra ke server Transfer Family lokal Anda dengan menggunakan AWS Command Line Interface (AWS CLI). Contoh AWS CLI perintah untuk mengirim pesan pengujian dibuat bersama dengan semua sumber daya lain di tumpukan.

Untuk menggunakan perintah sampel ini, buka tab Output dari tumpukan Anda, dan salin `TransferExampleAs2Command`. Anda kemudian dapat menjalankan perintah dengan menggunakan AWS CLI. Jika Anda belum menginstal AWS CLI, lihat [Menginstal atau memperbarui versi terbaru dari AWS CLI](#) Panduan AWS Command Line Interface Pengguna.

Perintah sampel memiliki format berikut:

```
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

Versi Anda dari perintah ini berisi nilai aktual untuk *TransferS3BucketName* dan *TransferConnectorId* sumber daya di tumpukan Anda.

Perintah sampel ini terdiri dari dua perintah terpisah yang dirantai bersama dengan menggunakan `&&` string.

Perintah pertama membuat file teks baru yang kosong di bucket Anda:

```
aws s3api put-object --bucket TransferS3BucketName --key test.txt
```

Kemudian, perintah kedua menggunakan konektor untuk mengirim file dari profil mitra ke profil lokal. Server Transfer Family memiliki pengaturan perjanjian yang memungkinkan profil lokal menerima pesan dari profil mitra.

```
aws transfer start-file-transfer --region aws-region --connector-id TransferConnectorId
--send-file-paths /TransferS3BucketName/test.txt
```

Setelah menjalankan perintah, Anda dapat pergi ke bucket Amazon S3 (*TransferS3BucketName*) dan melihat isinya. Jika perintah berhasil, Anda akan melihat objek berikut di ember Anda:

- *processed/*— Folder ini berisi JSON file yang menjelaskan file yang ditransfer dan MDN responsnya.
- *processing/*— Folder ini sementara berisi file saat sedang diproses, tetapi setelah transfer selesai, folder ini harus kosong.
- *server-id/*— Folder ini diberi nama berdasarkan ID server Transfer Family Anda. Ini berisi *from-partner* (folder ini dinamai secara dinamis, berdasarkan AS2 ID mitra), yang berisi *sendrifailed/*, *processed/*, dan *processing/* folder. */server-id/from-partner/processed/* Folder berisi salinan file teks yang ditransfer, dan yang sesuai JSON dan MDN file.
- *test.txt*— Objek ini adalah file (kosong) yang ditransfer.

AS2konfigurasi dan batasan

Topik ini menjelaskan konfigurasi, fitur, dan kemampuan yang didukung untuk transfer yang menggunakan protokol Applicability Statement 2 (AS2), termasuk cipher dan digest yang diterima. Bagian ini juga menjelaskan batasan dan masalah yang diketahui untuk AS2 transfer.

Topik

- [AS2konfigurasi yang didukung](#)
- [AS2kuota dan batasan](#)

AS2konfigurasi yang didukung

Penandatanganan, enkripsi, kompresi, MDN

Untuk transfer masuk dan keluar, item berikut wajib atau opsional:

- Enkripsi — Diperlukan (untuk HTTP transportasi, yang merupakan satu-satunya metode transportasi yang saat ini didukung). Pesan yang tidak terenkripsi hanya diterima jika diteruskan oleh proxy TLS -terminating seperti Application Load Balancer () dan header ada. ALB X-Forwarded-Proto: https
- Penandatanganan - Opsional
- Kompresi - Opsional (satu-satunya algoritma kompresi yang didukung saat ini adalah ZLIB)
- Pemberitahuan Disposisi Pesan (MDN) - Opsional

Cipher

Cipher berikut didukung untuk transfer masuk dan keluar:

- AES128_CBC
- AES192_CBC
- AES256_CBC
- 3 DES (hanya untuk kompatibilitas mundur)

Intisari

Intisari berikut didukung:

- Penandatanganan masuk dan MDN —SHA1,,SHA256, SHA384 SHA512
- Penandatanganan keluar dan MDN —SHA1,,SHA256, SHA384 SHA512

MDN

Untuk MDN tanggapan, jenis tertentu didukung, sebagai berikut:

- Transfer masuk — Sinkron dan asinkron
- Transfer keluar — Hanya sinkron
- Protokol Transfer Surat Sederhana (SMTP) (emailMDN) - Tidak didukung

Transportasi

- Transfer masuk — HTTP adalah satu-satunya transportasi yang didukung saat ini, dan Anda harus menentukannya secara eksplisit.

Note

Jika Anda perlu menggunakan HTTPS untuk transfer masuk, Anda dapat mengakhiri TLS pada Application Load Balancer atau Network Load Balancer. Ini dijelaskan dalam [Menerima AS2 pesan di atas HTTPS](#).

- Transfer keluar — Jika Anda memberikan HTTPURL, Anda juga harus menentukan algoritma enkripsi. Jika Anda memberikan HTTPSURL, Anda memiliki opsi NONE untuk menentukan algoritma enkripsi Anda.

AS2kuota dan batasan

Bagian ini membahas kuota dan batasan untuk AS2

Topik

- [AS2kuota](#)
- [Kuota untuk menangani rahasia](#)
- [Keterbatasan yang Sudah Diketahui](#)

AS2kuota

Kuota berikut tersedia untuk transfer AS2 file. Untuk meminta kenaikan kuota yang dapat disesuaikan, lihat [Layanan AWS kuota](#) di. Referensi Umum AWS

AS2kuota

Nama	Default	Dapat disesuaikan
AS2Permintaan masuk per server	25 per detik	Tidak
AS2Permintaan masuk sedang berlangsung per server	100	Tidak
Jumlah maksimum file per permintaan transfer file	10	Tidak

Nama	Default	Dapat disesuaikan
AS2Permintaan keluar sedang berlangsung per konektor	100	Tidak
Ukuran file maksimum (terkompresi atau tidak terkompresi)	50 MiB	Ya
Batas waktu tidak aktif	350 detik	Tidak
Jumlah maksimum profil mitra per akun	1000 (hingga 10 sertifikat per profil mitra: tidak dapat disesuaikan)	Ya
Jumlah maksimum sertifikat per akun	1000	Ya
Jumlah maksimum permintaan transfer file per detik per akun	3	Ya
Jumlah maksimum konektor per akun (keduanya SFTP dan AS2 konektor berkontribusi pada hitungan ini)	100	Ya
Bandwidth maksimum untuk konektor per akun (keduanya SFTP dan AS2 konektor berkontribusi pada nilai ini)	50 MBps	Tidak
Jumlah maksimum perjanjian per server	100	Ya

Kuota untuk menangani rahasia

AWS Transfer Family melakukan panggilan ke AWS Secrets Manager atas nama AS2 pelanggan yang menggunakan otentikasi Dasar. Selain itu Secrets Manager membuat panggilan ke AWS KMS.

Note

Kuota ini tidak spesifik untuk penggunaan rahasia Anda untuk Transfer Family: mereka dibagi di antara semua layanan di Akun AWS.

Untuk Secrets Manager `GetSecretValue`, kuota yang berlaku adalah Combined rate of `DescribeSecret` dan `GetSecretValue` API request, seperti yang dijelaskan dalam [AWS Secrets Manager kuota](#).

Secrets Manager `GetSecretValue`

Nama	Nilai	Deskripsi
Tingkat gabungan <code>DescribeSecret</code> dan <code>GetSecretValue</code> API permintaan	Setiap Wilayah yang didukung: 10.000 per detik	Transaksi maksimum per detik untuk <code>DescribeSecret</code> dan <code>GetSecretValue</code> API permintaan digabungkan.

Untuk AWS KMS, kuota berikut berlaku untuk `Decrypt`. Untuk detailnya, lihat [Meminta kuota untuk setiap operasi AWS KMS API](#)

AWS KMS `Decrypt`

Nama kuota	Nilai default (permintaan per detik)
Tingkat permintaan operasi kriptografi (simetris)	Kuota bersama ini bervariasi dengan Wilayah AWS dan jenis AWS KMS kunci yang digunakan dalam permintaan. Setiap kuota dihitung secara terpisah. • 5.500 (bersama) • 10.000 (bersama) di Wilayah berikut: • US East (Ohio) us-east-2 • Asia Pacific (Singapore) ap-southeast-1 • Asia Pacific (Sydney), ap-southeast-2 • Asia Pacific (Tokyo), ap-northeast-1

Nama kuota	Nilai default (permintaan per detik)
	• Europe (Frankfurt), eu-central-1 • Europe (London), eu-west-2 • 50.000 (bersama) di Wilayah berikut: • US East (N. Virginia), us-east-1 • US West (Oregon), us-west-2 • Europe (Ireland), eu-west-1
Kuota permintaan toko kunci kustom  Note Kuota ini hanya berlaku jika Anda menggunakan toko kunci eksternal.	Kuota permintaan toko kunci khusus dihitung secara terpisah untuk setiap toko kunci khusus. • 1.800 (dibagikan) untuk setiap toko AWS CloudHSM kunci • 1.800 (dibagikan) untuk setiap toko kunci eksternal

Keterbatasan yang Sudah Diketahui

- TCPKeep-alive sisi server tidak didukung. Waktu koneksi habis setelah 350 detik tidak aktif kecuali klien mengirim paket keep-alive.
- Agar perjanjian aktif diterima oleh layanan dan muncul di CloudWatch log Amazon, pesan harus berisi AS2 header yang valid.
- [Server yang menerima pesan dari AWS Transfer Family for AS2 harus mendukung atribut perlindungan algoritma Cryptographic Message Syntax \(CMS\) untuk memvalidasi tanda tangan pesan, seperti yang didefinisikan dalam 6211. RFC](#) Atribut ini tidak didukung di beberapa produk IBM Sterling yang lebih tua.
- Pesan duplikat IDs menghasilkan pesan diproses/peringatan: dokumen duplikat.
- Panjang kunci untuk AS2 sertifikat harus setidaknya 2048 bit, dan paling banyak 4096.
- Saat mengirim AS2 pesan atau asinkron MDNs ke HTTPS titik akhir mitra dagang, pesan atau MDNs harus menggunakan sertifikat yang valid yang ditandatangani oleh otoritas SSL sertifikat tepercaya publik (CA). Sertifikat yang ditandatangani sendiri saat ini tidak didukung.

- Titik akhir harus mendukung protokol TLS versi 1.2 dan algoritma kriptografi yang diizinkan oleh kebijakan keamanan (seperti yang dijelaskan dalam [Kebijakan keamanan untuk AWS Transfer Family](#)).
- Mutual TLS (mTLS) saat ini tidak didukung.
- Beberapa lampiran dan perpesanan pertukaran sertifikat (CEM) dari AS2 versi 1.2 saat ini tidak didukung.
- Autentikasi dasar saat ini hanya didukung untuk pesan keluar.

AS2 fitur dan kemampuan

Tabel berikut mencantumkan fitur dan kemampuan yang tersedia untuk sumber daya Transfer Family yang digunakan AS2.

AS2 fitur

Transfer Family menawarkan fitur-fitur berikut untuk AS2.

Fitur	Didukung oleh AWS Transfer Family
Sertifikasi Drummond	Ya
AWS CloudFormation dukungan	Ya
CloudWatch Metrik Amazon	Ya
SHA-2 algoritma kriptografi	Ya
Support untuk Amazon S3	Ya
Support untuk Amazon EFS	Tidak
Pesan Terjadwal	Ya ¹
AWS Transfer Family Alur Kerja Terkelola	Tidak
Pesan Pertukaran Sertifikat (CEM)	Tidak
Saling Bersama TLS (mTLS)	Tidak

1. Pesan Terjadwal Keluar tersedia dengan [menjadwalkan AWS Lambda fungsi menggunakan Amazon EventBridge](#)

AS2 mengirim dan menerima kemampuan

Tabel berikut menyediakan daftar kemampuan AWS Transfer Family AS2 mengirim dan menerima.

Kemampuan	Inbound: Menerima dengan server	Keluar: Mengirim dengan konektor
TLSTransportasi Terenkripsi () HTTPS	Ya ¹	Ya
TLSNon-Transportasi (HTTP)	Ya	Ya ²
Sinkron MDN	Ya	Ya
Kompresi Pesan	Ya	Ya
Asinkron MDN	Ya	Tidak
Alamat IP Statis	Ya	Ya
Bawa Alamat IP Anda Sendiri	Ya	Tidak
Beberapa Lampiran File	Tidak	Tidak
Otentikasi Dasar	Tidak	Ya
AS2Mulai ulang	Tidak berlaku	Tidak
Subjek Kustom per Pesan	Tidak berlaku	Tidak

1. Transportasi TLS Terenkripsi Masuk tersedia dengan Network Load Balancer () NLB

2. Outbound non- TLS Transport hanya tersedia saat enkripsi diaktifkan

Konfigurasi AS2 konektor

Tujuan dari konektor adalah untuk membangun hubungan antara mitra dagang untuk transfer keluar — mengirim AS2 file dari server Transfer Family ke tujuan eksternal milik mitra. Untuk konektor, Anda

menentukan pihak lokal, mitra jarak jauh, dan sertifikat mereka (dengan membuat profil lokal dan mitra). Setelah Anda memiliki konektor, Anda dapat mentransfer informasi ke mitra dagang Anda.

Note

Ukuran pesan yang diterima oleh mitra dagang tidak akan cocok dengan ukuran objek di Amazon S3. Perbedaan ini terjadi karena AS2 pesan membungkus file dalam amplop sebelum dikirim. Jadi, ukuran file mungkin meningkat, bahkan jika file dikirim dengan kompresi. Oleh karena itu, pastikan bahwa ukuran file maksimum mitra dagang lebih besar dari ukuran file yang Anda kirim.

Pernyataan Penerapan 2 (AS2) konektor membangun hubungan antara mitra dagang untuk transfer keluar. Untuk informasi lebih lanjut tentang AS2 konektor, lihat [Konfigurasi AS2 konektor](#).

Buat AS2 konektor

Prosedur ini menjelaskan cara membuat AS2 konektor dengan menggunakan AWS Transfer Family konsol. Jika Anda ingin menggunakan AWS CLI sebagai gantinya, lihat [the section called “Langkah 6: Buat konektor antara Anda dan pasangan”](#).

Untuk membuat AS2 konektor

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Di panel navigasi kiri, pilih Konektor, lalu pilih Buat konektor.
3. Di bagian Konfigurasi konektor, tentukan informasi berikut:
 - URL— Masukkan URL untuk koneksi keluar.
 - Peran akses — Pilih Amazon Resource Name (ARN) dari peran AWS Identity and Access Management (IAM) yang akan digunakan. Pastikan peran ini menyediakan akses baca dan tulis ke direktori induk lokasi file yang digunakan dalam `StartFileTransfer` permintaan. Selain itu, pastikan bahwa peran tersebut menyediakan akses baca dan tulis ke direktori induk dari file yang ingin Anda kirim `StartFileTransfer`.

Note

Jika Anda menggunakan otentikasi Dasar untuk konektor Anda, peran akses memerlukan `secretsmanager:GetSecretValue` izin untuk rahasia tersebut.

Jika rahasia dienkripsi dengan menggunakan kunci yang dikelola pelanggan alih-alih Kunci yang dikelola AWS masuk AWS Secrets Manager, maka peran tersebut juga memerlukan kms:Decrypt izin untuk kunci itu. Jika Anda memberi nama rahasia Anda dengan awalan `aws/transfer/`, Anda dapat menambahkan izin yang diperlukan dengan karakter wildcard (*), seperti yang ditunjukkan dalam [izin Contoh untuk membuat](#) rahasia.

- Peran logging (opsional) - Pilih IAM peran konektor yang akan digunakan untuk mendorong peristiwa ke CloudWatch log Anda.
4. Di bagian AS2konfigurasi, pilih profil lokal dan mitra, algoritma enkripsi dan penandatanganan, dan apakah akan mengompres informasi yang ditransfer. Perhatikan hal berikut:
- Untuk algoritma enkripsi, jangan memilih DES_EDE3_CBC kecuali Anda harus mendukung klien lama yang membutuhkannya, karena ini adalah algoritma enkripsi yang lemah.
 - Subjek digunakan sebagai atribut subject HTTP header dalam AS2 pesan yang sedang dikirim dengan konektor.
 - Jika Anda memilih untuk membuat konektor tanpa algoritma enkripsi, Anda harus menentukan HTTPS sebagai protokol Anda.
5. Di bagian MDNkonfigurasi, tentukan informasi berikut:
- Permintaan MDN — Anda memiliki opsi untuk meminta mitra dagang Anda mengirimi Anda pesan MDN setelah mereka berhasil menerima pesan AndaAS2.
 - Ditandatangani MDN — Anda memiliki opsi untuk MDNs mewajibkannya ditandatangani. Opsi ini hanya tersedia jika Anda telah memilih Permintaan MDN.
6. Di bagian otentikasi dasar, tentukan informasi berikut.
- Untuk mengirim kredensi masuk bersama dengan pesan keluar, pilih Aktifkan otentikasi Dasar. Jika Anda tidak ingin mengirim kredensi apa pun dengan pesan keluar, biarkan Aktifkan otentikasi Dasar tetap bersih.
 - Jika Anda menggunakan otentikasi, pilih atau buat rahasia.
 - Untuk membuat rahasia baru, pilih Buat rahasia baru lalu masukkan nama pengguna dan kata sandi. Kredensi ini harus sesuai dengan pengguna yang terhubung ke titik akhir mitra.

Basic authentication [Info](#)

☒ **Enable Basic authentication - optional**
Select this option to authenticate with your trading partner's host using username and password credentials.

Basic authentication credentials [Info](#)
Choose the username and password credentials that will be used to authenticate with your trading partner's host.

☒ Create a new secret

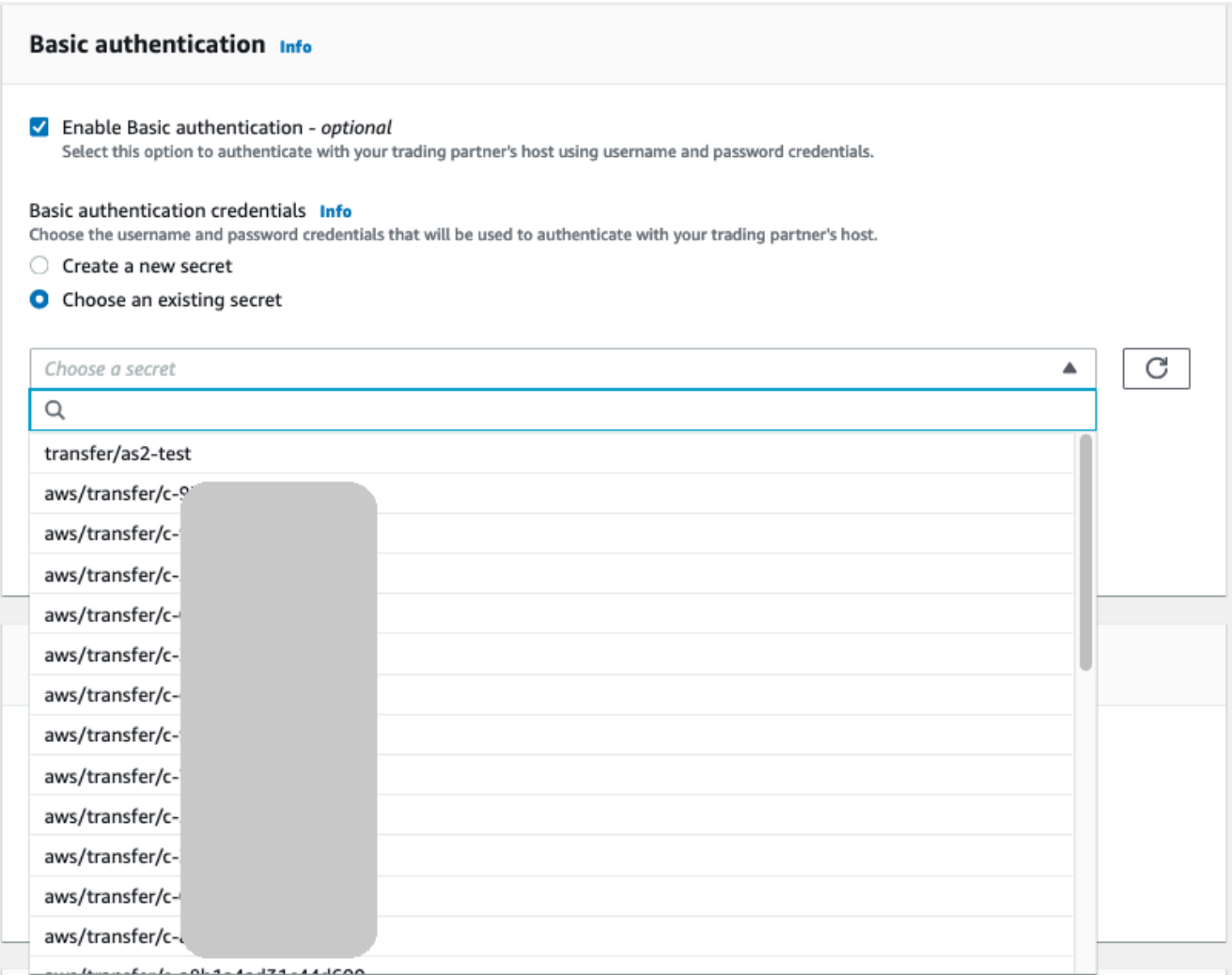
☐ Choose an existing secret

Username

Password

 Update the access role associated with your connector to provide AWS Transfer Family with permission to read the secret containing your Basic authentication credentials.

- Untuk menggunakan rahasia yang ada, pilih Pilih rahasia yang ada, lalu pilih rahasia dari menu tarik-turun. Untuk detail tentang membuat rahasia yang diformat dengan benar di Secrets Manager, lihat [Aktifkan otentikasi dasar untuk konektor AS2](#).



7. Setelah Anda mengonfirmasi semua pengaturan Anda, pilih **Buat konektor** untuk membuat konektor.

Halaman Konektor muncul, dengan ID konektor baru Anda ditambahkan ke daftar. Untuk melihat detail konektor Anda, lihat [Lihat detail AS2 konektor](#).

AS2 algoritma konektor

Saat Anda membuat AS2 konektor, algoritme keamanan berikut terpasang ke konektor.

Tipe	Algoritme
TLSCipher	TLS _ ECDHE _ ECDSA WITH _ AES GCM _128 _ SHA256

Tipe	Algoritme
	TLS_ _ ECDHE _ RSA WITH _ AES GCM _128_ _ SHA256
	TLS_ _ ECDHE _ ECDSA WITH _ AES CBC _128_ _ SHA256
	TLS_ _ ECDHE _ RSA WITH _ AES CBC _128_ _ SHA256
	TLS_ _ ECDHE _ ECDSA WITH _ AES GCM _256_ _ SHA384
	TLS_ _ ECDHE _ RSA WITH _ AES GCM _256_ _ SHA384
	TLS_ _ ECDHE _ ECDSA WITH _ AES CBC _256_ _ SHA384
	TLS_ _ ECDHE _ RSA WITH _ AES CBC _256_ _ SHA384

Otentikasi dasar untuk konektor AS2

Saat membuat atau memperbarui server Transfer Family yang menggunakan AS2 protokol, Anda dapat menambahkan otentikasi Dasar untuk pesan keluar. Anda melakukan ini dengan menambahkan informasi otentikasi ke konektor.

Note

Otentikasi dasar hanya tersedia jika Anda menggunakan HTTPS.

Untuk menggunakan otentikasi untuk konektor Anda, pilih Aktifkan otentikasi Dasar di bagian Autentikasi dasar. Setelah mengaktifkan otentikasi Dasar, Anda dapat memilih untuk membuat rahasia baru, atau menggunakan yang sudah ada. Dalam kedua kasus, kredensial dalam rahasia dikirim dengan pesan keluar yang menggunakan konektor ini. Kredensialnya harus sesuai dengan pengguna yang mencoba terhubung ke titik akhir jarak jauh mitra dagang.

Tangkapan layar berikut menunjukkan Aktifkan otentikasi Dasar yang dipilih, dan Buat rahasia baru yang dipilih. Setelah membuat pilihan ini, Anda dapat memasukkan nama pengguna dan kata sandi untuk rahasianya.

Basic authentication [Info](#)

☒ **Enable Basic authentication - optional**
Select this option to authenticate with your trading partner's host using username and password credentials.

Basic authentication credentials [Info](#)
Choose the username and password credentials that will be used to authenticate with your trading partner's host.

☒ **Create a new secret**

☐ Choose an existing secret

Username

Password

 Update the access role associated with your connector to provide AWS Transfer Family with permission to read the secret containing your Basic authentication credentials.

Tangkapan layar berikut menunjukkan Aktifkan otentikasi dasar dipilih, dan Pilih rahasia yang ada dipilih. Rahasia Anda harus dalam format yang benar, seperti yang dijelaskan dalam [Aktifkan otentikasi dasar untuk konektor AS2](#).

[illegible]

Aktifkan otentikasi dasar untuk konektor AS2

Saat Anda mengaktifkan otentikasi Dasar untuk AS2 konektor, Anda dapat membuat rahasia baru di konsol Transfer Family, atau Anda dapat menggunakan rahasia yang Anda buat. AWS Secrets Manager Dalam kedua kasus tersebut, rahasia Anda disimpan di Secrets Manager.

Topik

- Buat rahasia baru di konsol
- Gunakan secret yang sudah ada
- Buat rahasia di AWS Secrets Manager

Buat rahasia baru di konsol

Saat Anda membuat konektor di konsol, Anda dapat membuat rahasia baru.

Untuk membuat rahasia baru, pilih Buat rahasia baru lalu masukkan nama pengguna dan kata sandi. Kredensi ini harus sesuai dengan pengguna yang terhubung ke titik akhir mitra.

Basic authentication [Info](#)

☒ **Enable Basic authentication - optional**
Select this option to authenticate with your trading partner's host using username and password credentials.

Basic authentication credentials [Info](#)
Choose the username and password credentials that will be used to authenticate with your trading partner's host.

☒ Create a new secret

☐ Choose an existing secret

Username

Password

 Update the access role associated with your connector to provide AWS Transfer Family with permission to read the secret containing your Basic authentication credentials.

Note

Saat Anda membuat rahasia baru di konsol, nama rahasia mengikuti konvensi penamaan ini: **/aws/transfer/connector-id**, di mana **connector-id** adalah ID konektor yang Anda buat. Pertimbangkan ini ketika Anda mencoba menemukan rahasianya AWS Secrets Manager.

Gunakan secret yang sudah ada

Saat Anda membuat konektor di konsol, Anda dapat menentukan rahasia yang ada.

Untuk menggunakan rahasia yang ada, pilih Pilih rahasia yang ada, lalu pilih rahasia dari menu tarik-turun. Untuk detail tentang membuat rahasia yang diformat dengan benar di Secrets Manager, lihat [Buat rahasia di AWS Secrets Manager](#).

[illegible]

Buat rahasia di AWS Secrets Manager

Prosedur berikut menjelaskan cara membuat rahasia yang sesuai untuk digunakan dengan AS2 konektor Anda.

 Note

Otentikasi dasar hanya tersedia jika Anda menggunakan HTTPS.

Untuk menyimpan kredensi pengguna di Secrets Manager untuk AS2 otentikasi Dasar

1. Masuk ke AWS Management Console dan buka AWS Secrets Manager konsol di <https://console.aws.amazon.com/secretsmanager/>.
2. Pada panel navigasi kiri, pilih Rahasia.
3. Pada halaman Rahasia, pilih Simpan rahasia baru.
4. Pada halaman Pilih jenis rahasia, untuk tipe Rahasia, pilih Jenis rahasia lainnya.
5. Di bagian pasangan kunci/Nilai, pilih tab kunci/Nilai.
 - Kunci — Masukkan **Username**.
 - nilai — Masukkan nama pengguna yang berwenang untuk terhubung ke server mitra.
6. Jika Anda ingin memberikan kata sandi, pilih Tambahkan baris, dan di bagian pasangan kunci/nilai, pilih tab kunci/Nilai.

Pilih Tambah baris, dan di bagian pasangan kunci/Nilai, pilih tab kunci/Nilai.

 - Kunci — Masukkan **Password**.
 - nilai — Masukkan kata sandi untuk pengguna.
7. Jika Anda ingin memberikan kunci pribadi, pilih Tambah baris, dan di bagian pasangan kunci/nilai, pilih tab kunci/Nilai.
 - Kunci — Masukkan **PrivateKey**.
 - nilai — Masukkan kunci pribadi untuk pengguna. Nilai ini harus disimpan dalam SSH format Terbuka, dan harus sesuai dengan kunci publik yang disimpan untuk pengguna ini di server jarak jauh.
8. Pilih Berikutnya.
9. Pada halaman Konfigurasi rahasia, masukkan nama dan deskripsi untuk rahasia Anda. Kami menyarankan Anda menggunakan awalan **aws/transfer/** untuk nama tersebut. Misalnya, Anda bisa menyebutkan rahasia Anda **aws/transfer/connector-1**.
10. Pilih Berikutnya, dan kemudian terima default pada halaman Konfigurasi rotasi. Lalu pilih Berikutnya.
11. Pada halaman Review, pilih Store untuk membuat dan menyimpan rahasia.

Setelah Anda membuat rahasia, Anda dapat memilihnya saat Anda membuat konektor (lihat [Konfigurasi AS2 konektor](#)). Pada langkah di mana Anda mengaktifkan otentikasi Dasar, pilih rahasia dari daftar dropdown rahasia yang tersedia.

Lihat detail AS2 konektor

Anda dapat menemukan daftar detail dan properti untuk AS2 AWS Transfer Family konektor di AWS Transfer Family konsol. Properti AS2 konektor mencakup URL, peran, profil, tagMDNs, dan metrik pemantauannya.

Ini adalah prosedur untuk melihat detail konektor.

Untuk melihat detail konektor

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Di panel navigasi kiri, pilih Konektor.
3. Pilih pengenal di kolom Connector ID untuk melihat halaman detail untuk konektor yang dipilih.

Anda dapat mengubah properti untuk AS2 konektor pada halaman detail konektor dengan memilih Edit.

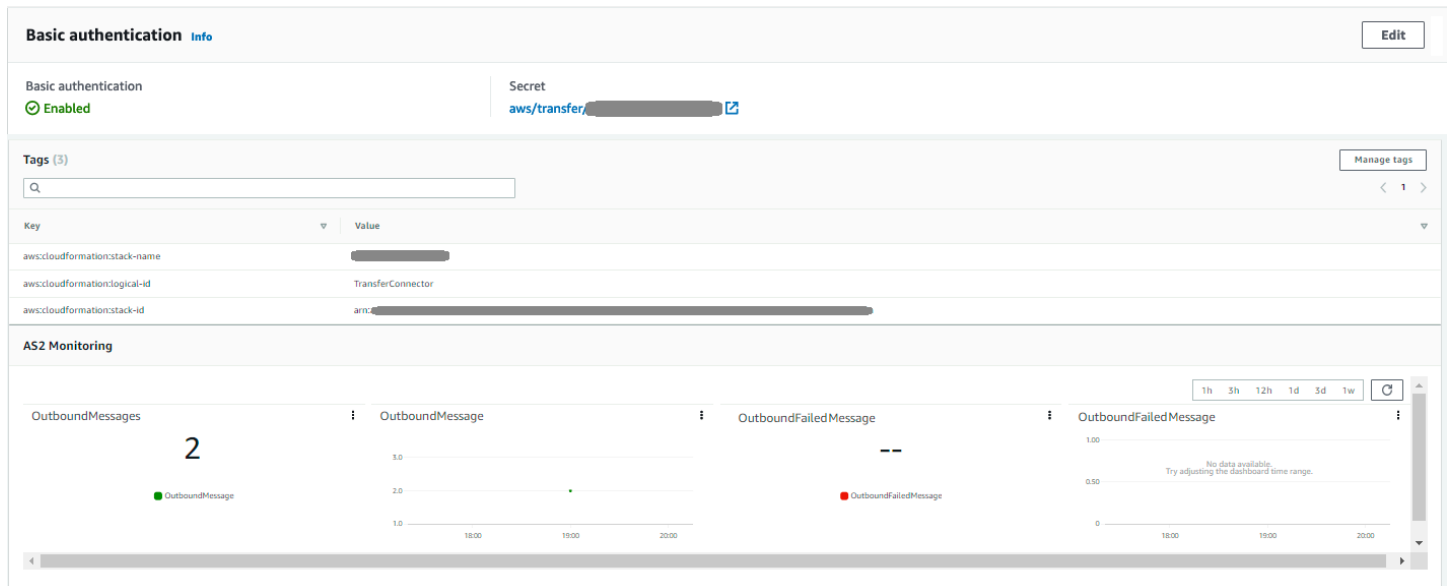
The screenshot displays the AWS Transfer Family console interface for a specific connector. The breadcrumb navigation at the top reads 'Transfer Family > Connectors > c-'. The connector ID is partially visible as 'c-'. The 'Connector configuration' section includes fields for 'URL' (http://), 'Access role', and 'Logging role', each with an external link icon. The 'Communication settings' section shows 'AS2-From header' (partner-test) and 'AS2-To header' (local-test). The 'AS2 configuration' section lists 'Local profile' (partner-test), 'Partner profile' (local-test), 'Compression' (Disabled), 'Message Subject' (View), 'Encryption algorithm' (AES256_CBC), and 'Signing algorithm' (SHA256). The 'MDN configuration' section shows 'Request MDN' (Enabled), 'Signed MDN' (Default to message signing algorithm: SHA256), and 'Synchronization' (Enabled).

Connector configuration		
URL	Access role	Logging role
http://		

Communication settings	
AS2-From header	AS2-To header
partner-test	local-test

AS2 configuration		
Local profile	Compression	Encryption algorithm
partner-test	Disabled	AES256_CBC
Partner profile	Message Subject	Signing algorithm
local-test	View	SHA256

MDN configuration		
Request MDN	Signed MDN	Synchronization
Enabled	Default to message signing algorithm: SHA256	Enabled



Note

Anda bisa mendapatkan banyak informasi ini, meskipun dalam format yang berbeda, dengan menjalankan perintah berikut AWS Command Line Interface (AWS CLI) :

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

Untuk informasi selengkapnya, silakan lihat [DescribeConnector](#) dalam API referensi.

Kelola AS2 mitra

Topik ini membahas cara mengelola AS2 sertifikat, profil, dan perjanjian.

Impor AS2 sertifikat

AS2Proses Transfer Family menggunakan kunci sertifikat untuk enkripsi dan penandatanganan informasi yang ditransfer. Mitra dapat menggunakan kunci yang sama untuk kedua tujuan, atau kunci terpisah untuk masing-masing. Jika Anda memiliki kunci enkripsi umum yang disimpan di escrow oleh pihak ketiga tepercaya sehingga data dapat didekripsi jika terjadi bencana atau pelanggaran keamanan, sebaiknya Anda memiliki kunci penandatanganan terpisah. Dengan menggunakan kunci penandatanganan terpisah (yang tidak Anda escrow), Anda tidak mengkompromikan fitur non-penolakan tanda tangan digital Anda.

 Note

Panjang kunci untuk AS2 sertifikat harus setidaknya 2048 bit, dan paling banyak 4096.

Poin-poin berikut merinci bagaimana AS2 sertifikat digunakan selama proses berlangsung.

- Masuk AS2
 - Mitra dagang mengirimkan kunci publik mereka untuk sertifikat penandatanganan, dan kunci ini diimpor ke profil mitra.
 - Pihak lokal mengirimkan kunci publik untuk enkripsi dan sertifikat penandatanganan mereka. Mitra kemudian mengimpor kunci pribadi atau kunci. Pihak lokal dapat mengirim kunci sertifikat terpisah untuk penandatanganan dan enkripsi, atau dapat memilih untuk menggunakan kunci yang sama untuk kedua tujuan tersebut.
- Keluar AS2
 - Mitra mengirimkan kunci publik untuk sertifikat enkripsi mereka, dan kunci ini diimpor ke profil mitra.
 - Pihak lokal mengirimkan kunci publik untuk sertifikat untuk ditandatangani, dan mengimpor kunci pribadi sertifikat untuk ditandatangani.

Untuk detail tentang cara membuat sertifikat, lihat [the section called “Langkah 1: Buat sertifikat untuk AS2”](#).

Prosedur ini menjelaskan cara mengimpor sertifikat dengan menggunakan konsol Transfer Family. Jika Anda ingin menggunakan AWS CLI sebagai gantinya, lihat [the section called “Langkah 3: Impor sertifikat sebagai sumber sertifikat Transfer Family”](#).

Untuk menentukan sertifikat AS2 -enabled

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Di panel navigasi kiri, di bawah Mitra AS2 Dagang, pilih Sertifikat.
3. Pilih Impor sertifikat.
4. Di bagian Deskripsi sertifikat, masukkan nama yang mudah diidentifikasi untuk sertifikat. Pastikan Anda dapat mengidentifikasi tujuan sertifikat dengan deskripsinya. Selain itu, pilih peran untuk sertifikat.

5. Di bagian Konten Sertifikat, berikan sertifikat publik dari mitra dagang, atau kunci publik dan pribadi untuk sertifikat lokal.
6. Di bagian Penggunaan sertifikat, pilih tujuan untuk sertifikat ini. Ini dapat digunakan untuk enkripsi, penandatanganan, atau keduanya.

 Note

Jika Anda memilih Enkripsi dan menandatangani untuk penggunaan, Transfer Family akan membuat dua sertifikat identik (masing-masing memiliki ID sendiri): satu dengan nilai penggunaan ENCRYPTION dan satu dengan nilai penggunaan SIGNING.

7. Isi bagian Isi sertifikat dengan detail yang sesuai.
 - Jika Anda memilih Sertifikat yang ditandatangani sendiri, Anda tidak memberikan rantai sertifikat.
 - Tempel di isi sertifikat.
 - Jika sertifikat bukan sertifikat yang ditandatangani sendiri, berikan rantai sertifikat.
 - Jika sertifikat ini adalah sertifikat lokal, tempel di kunci pribadinya.
8. Pilih Impor sertifikat untuk menyelesaikan proses dan menyimpan rincian untuk sertifikat yang diimpor.

AS2rotasi sertifikat

Seringkali, sertifikat berlaku untuk jangka waktu enam bulan hingga satu tahun. Anda mungkin telah menyiapkan profil yang ingin Anda pertahankan untuk durasi yang lebih lama. Untuk memfasilitasi hal ini, Transfer Family menyediakan rotasi sertifikat. Anda dapat menentukan beberapa sertifikat untuk profil, memungkinkan Anda untuk tetap menggunakan profil selama beberapa tahun. Transfer Family menggunakan sertifikat untuk penandatanganan (opsional) dan enkripsi (wajib). Anda dapat menentukan satu sertifikat untuk kedua tujuan, jika Anda mau.

Rotasi sertifikat adalah proses penggantian sertifikat lama yang kedaluwarsa dengan sertifikat yang lebih baru. Transisi adalah transisi bertahap untuk menghindari gangguan transfer di mana mitra dalam perjanjian belum mengonfigurasi sertifikat baru untuk transfer keluar atau mungkin mengirim muatan yang ditandatangani atau dienkripsi dengan sertifikat lama selama periode ketika sertifikat yang lebih baru mungkin juga digunakan. Periode menengah di mana sertifikat lama dan baru valid disebut sebagai masa tenggang.

Sertifikat X.509 memiliki `Not Before` dan tanggal. `Not After` Namun, parameter ini mungkin tidak memberikan kontrol yang cukup untuk administrator. `Transfer Family Inactive Date` menyediakan `Active Date` dan mengatur untuk mengontrol sertifikat mana yang digunakan untuk muatan keluar dan mana yang diterima untuk muatan masuk.

Pemilihan sertifikat keluar menggunakan nilai maksimum yang sebelum tanggal transfer sebagai `Inactive Date`. Proses masuk menerima sertifikat dalam kisaran `Not Before` dan `Not After` dan dalam kisaran `Active Date` dan `Inactive Date`.

Tabel berikut menjelaskan satu cara yang mungkin untuk mengkonfigurasi dua sertifikat untuk satu profil.

Dua sertifikat dalam rotasi

Nama	NOT BEFORE (dikendalikan oleh otoritas sertifikat)	ACTIVE DATE (diatur oleh Transfer Family)	INACTIVE DATE (diatur oleh Transfer Family)	NOT AFTER (ditetapkan oleh otoritas sertifikat)
Cert1 (sertifikat lama)	2019-11-01	2020-01-01	2020-12-31	2024-01-01
Cert2 (sertifikat yang lebih baru)	01/11/2020	2020-06-01	2021-06-01	2025-01-01

Perhatikan hal berikut:

- Saat Anda menentukan `Active Date` dan `Inactive Date` untuk sertifikat, rentang harus berada di dalam rentang antara `Not Before` dan `Not After`.
- Kami menyarankan Anda mengonfigurasi beberapa sertifikat untuk setiap profil, memastikan bahwa rentang tanggal aktif untuk semua sertifikat yang digabungkan mencakup jumlah waktu yang ingin Anda gunakan profil.
- Kami menyarankan Anda menentukan waktu tenggang antara saat sertifikat lama Anda menjadi tidak aktif dan ketika sertifikat Anda yang lebih baru menjadi aktif. Dalam contoh sebelumnya, sertifikat pertama tidak menjadi tidak aktif hingga 2020-12-31, sedangkan sertifikat kedua menjadi aktif pada 2020-06-01, memberikan masa tenggang 6 bulan. Selama periode 2020-06-01 hingga 2020-12-31, kedua sertifikat aktif.

Buat AS2 profil

Gunakan prosedur ini untuk membuat profil lokal dan mitra. Prosedur ini menjelaskan cara membuat AS2 profil dengan menggunakan konsol Transfer Family. Jika Anda ingin menggunakan AWS CLI sebagai gantinya, lihat [the section called “Langkah 4: Buat profil untuk Anda dan mitra dagang Anda”](#).

Untuk membuat AS2 profil

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Di panel navigasi kiri, di bawah Mitra AS2 Dagang, pilih Profil, lalu pilih Buat profil.
3. Di bagian Konfigurasi profil, masukkan AS2 ID untuk profil. Nilai ini digunakan untuk HTTP header AS2 khusus protokol as2-from dan as2-to untuk mengidentifikasi kemitraan perdagangan, yang menentukan sertifikat yang akan digunakan, dan sebagainya.
4. Di bagian Jenis profil, pilih Profil lokal atau Profil mitra.
5. Di bagian Sertifikat, pilih satu atau beberapa sertifikat dari menu tarik-turun.

Note

Jika Anda ingin mengimpor sertifikat yang tidak tercantum dalam menu tarik-turun, pilih Impor Sertifikat baru. Ini membuka jendela browser baru di layar Impor sertifikat. Untuk prosedur tentang mengimpor sertifikat lihat [Impor AS2 sertifikat](#).

6. (Opsional) Di tag bagian, tentukan satu atau beberapa pasangan nilai kunci untuk membantu mengidentifikasi profil ini.
7. Pilih Buat profil untuk menyelesaikan proses dan menyimpan profil baru.

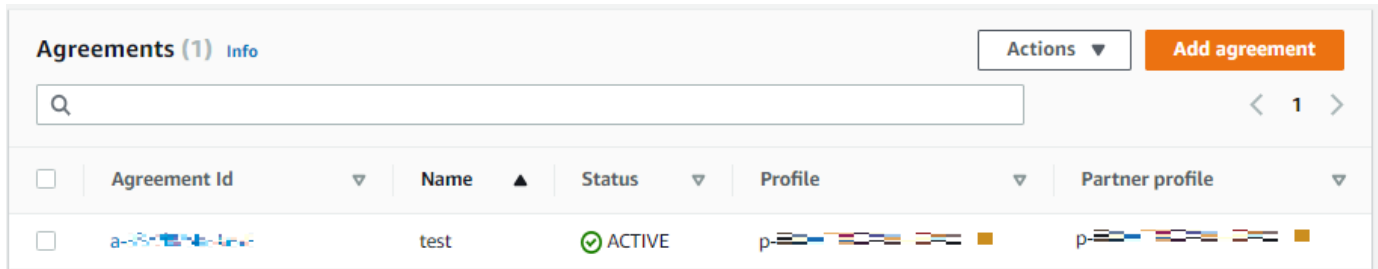
Buat AS2 perjanjian

Perjanjian terkait dengan server Transfer Family. Mereka menentukan detail untuk mitra dagang yang menggunakan AS2 protokol untuk bertukar pesan atau file dengan menggunakan Transfer Family, untuk transfer masuk—mengirim AS2 file dari sumber eksternal milik mitra ke server Transfer Family.

Prosedur ini menjelaskan cara membuat AS2 perjanjian dengan menggunakan konsol Transfer Family. Jika Anda ingin menggunakan AWS CLI sebagai gantinya, lihat [the section called “Langkah 5: Buat kesepakatan antara Anda dan pasangan”](#).

Untuk membuat perjanjian untuk server Transfer Family

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Di panel navigasi kiri, pilih Server, lalu pilih server yang menggunakan AS2 protokol.
3. Pada halaman detail server, gulir ke bawah ke bagian Perjanjian.



Agreements (1) Info					
< 1 >					
☐	Agreement Id	Name	Status	Profile	Partner profile
☐	a-...	test	ACTIVE	p-...	p-...

4. Pilih Tambahkan perjanjian.
5. Isi parameter perjanjian, sebagai berikut:
 - a. Di bagian Konfigurasi Perjanjian, masukkan nama deskriptif. Pastikan bahwa Anda dapat mengidentifikasi tujuan perjanjian dengan namanya. Juga, atur Status untuk perjanjian: Aktif (dipilih secara default) atau Tidak Aktif.
 - b. Di bagian Konfigurasi komunikasi, pilih profil lokal dan profil mitra.
 - c. Di bagian konfigurasi folder Kotak Masuk, pilih bucket Amazon S3 untuk menyimpan file masuk dan peran IAM yang dapat mengakses bucket. Secara opsional, Anda dapat memasukkan awalan (folder) yang akan digunakan untuk menyimpan file di ember.

Misalnya, jika Anda memasukkan **DOC-EXAMPLE-BUCKET** bucket dan **incoming** awalan, file masuk Anda akan disimpan ke folder/DOC-EXAMPLE-BUCKET/incoming.
 - d. (Opsional) Tambahkan tag di bagian Tag.
 - e. Setelah Anda memasukkan semua informasi untuk perjanjian, pilih Buat perjanjian.

Perjanjian baru muncul di bagian Perjanjian pada halaman detail server.

Mengirim dan menerima AS2 pesan

Bagian ini menjelaskan proses untuk mengirim dan menerima AS2 pesan. Ini juga memberikan rincian tentang nama file dan lokasi yang terkait dengan pesan. AS2

Topik

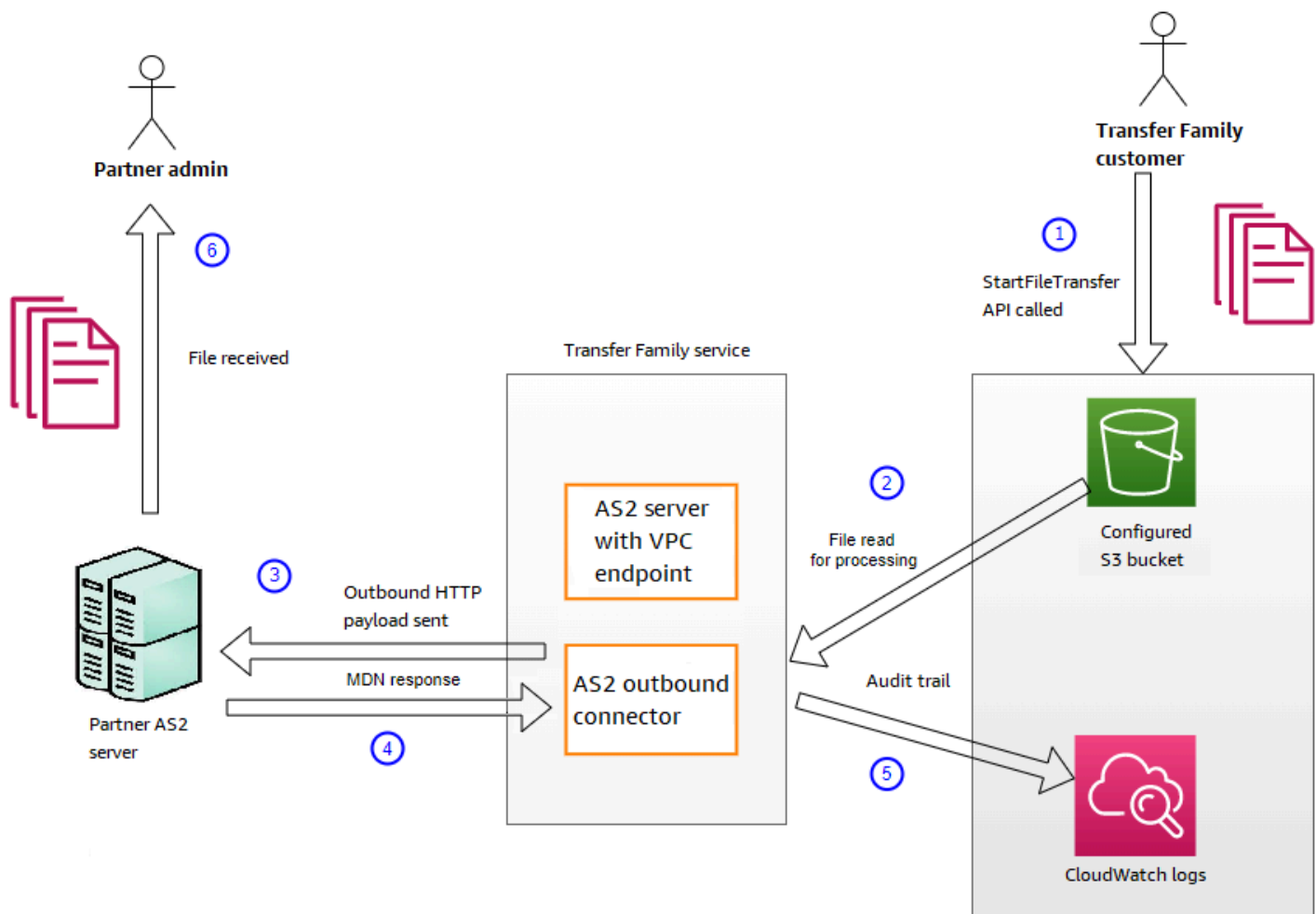
- [Proses kirim AS2 pesan](#)

- [Menerima proses AS2 pesan](#)
- [Mengirim dan menerima AS2 pesan HTTPS](#)
- [Mentransfer file dengan menggunakan konektor AS2](#)
- [Nama dan lokasi file](#)
- [Kode status](#)
- [Contoh JSON file](#)

Proses kirim AS2 pesan

Proses keluar didefinisikan sebagai pesan atau file yang dikirim dari AWS ke klien atau layanan eksternal. Urutan pesan keluar adalah sebagai berikut:

1. Admin memanggil perintah `start-file-transfer` AWS Command Line Interface (AWS CLI) atau `StartFileTransfer` API operasi. Operasi ini mereferensikan `connector` konfigurasi.
2. Transfer Family mendeteksi permintaan file baru dan menemukan file tersebut. File dikompresi, ditandatangani, dan dienkripsi.
3. HTTPKlien transfer melakukan HTTP POST permintaan untuk mengirimkan muatan ke AS2 server mitra.
4. Proses mengembalikan MDN respons yang ditandatangani, sejalan dengan HTTP respons (MDNsinkron).
5. Saat file bergerak di antara berbagai tahap transmisi, proses memberikan tanda terima MDN respons dan detail pemrosesan kepada pelanggan.
6. AS2Server jarak jauh membuat file yang didekripsi dan diverifikasi tersedia untuk admin mitra.



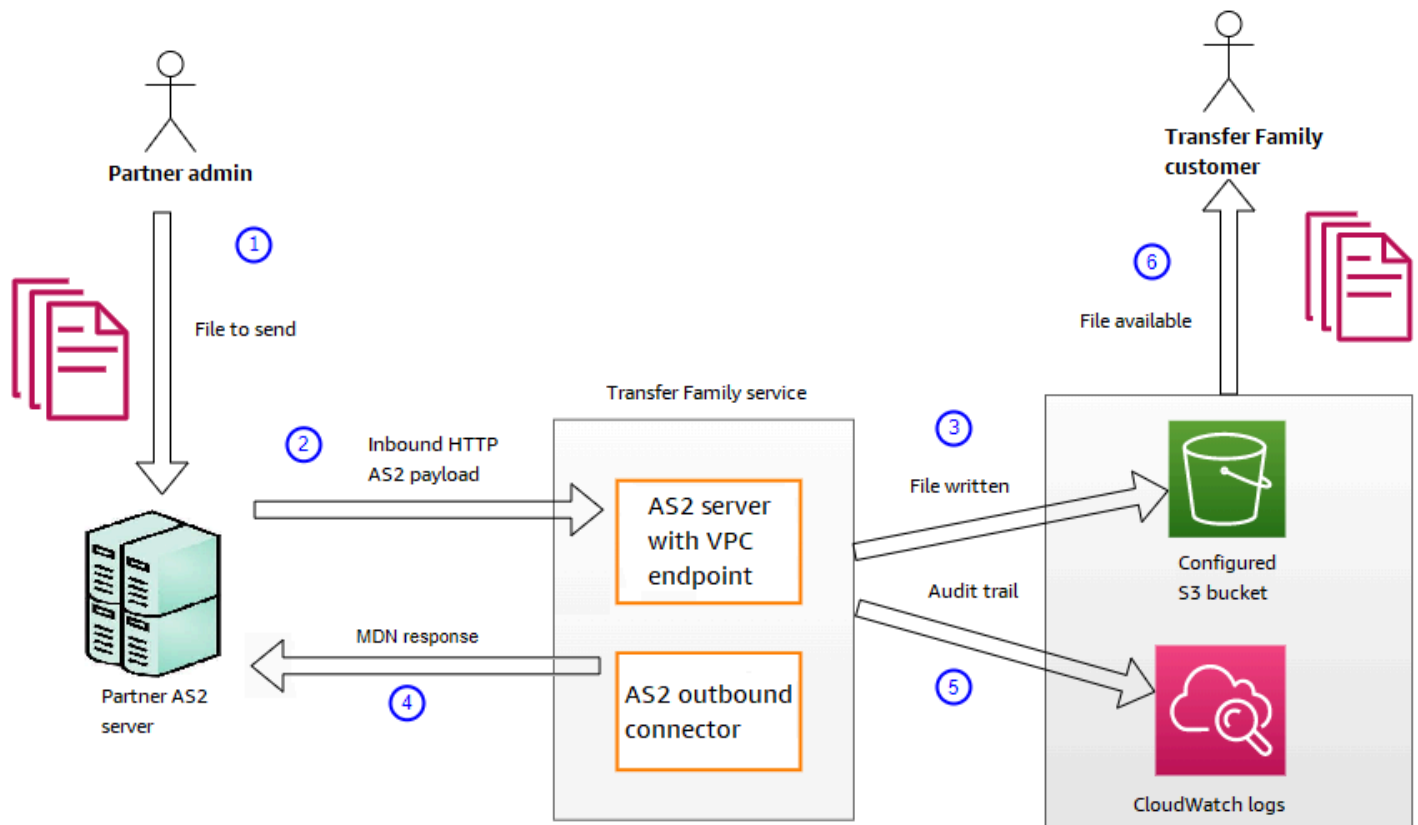
AS2Processing mendukung banyak dari RFC 4130 protokol, dengan fokus pada kasus penggunaan umum dan integrasi dengan implementasi server AS2 -enabled yang ada. Untuk detail konfigurasi yang didukung, lihat [AS2konfigurasi yang didukung](#).

Menerima proses AS2 pesan

Proses inbound didefinisikan sebagai pesan atau file yang sedang ditransfer ke AWS Transfer Family server Anda. Urutan pesan masuk adalah sebagai berikut:

1. Admin atau proses otomatis memulai transfer AS2 file di AS2 server jarak jauh mitra.
2. AS2Server jarak jauh mitra menandatangani dan mengenkripsi konten file, lalu mengirimkan HTTP POST permintaan ke titik akhir AS2 masuk yang dihosting di Transfer Family.
3. Menggunakan nilai yang dikonfigurasi untuk server, mitra, sertifikat, dan perjanjian, Transfer Family mendekripsi dan memverifikasi muatan. AS2 Isi file disimpan di toko file Amazon S3 yang dikonfigurasi.

4. MDNRespons yang ditandatangani dikembalikan baik sejalan dengan HTTP respons, atau secara asinkron melalui HTTP POST permintaan terpisah kembali ke server asal.
5. Jejak audit ditulis ke Amazon CloudWatch dengan rincian tentang pertukaran.
6. File yang didekripsi tersedia dalam folder bernama. `inbox/processed`



Mengirim dan menerima AS2 pesan HTTPS

Bagian ini menjelaskan cara mengonfigurasi server Transfer Family yang menggunakan AS2 protokol untuk mengirim dan menerima pesan HTTPS.

Kirim AS2 pesan HTTPS

Untuk mengirim AS2 pesan menggunakan HTTPS, buat konektor dengan informasi berikut:

- Untuk URL, tentukan HTTPS URL
- Untuk algoritma enkripsi, tentukan NONE.
- Berikan nilai yang tersisa untuk konektor seperti yang dijelaskan dalam [Konfigurasi AS2 konektor](#).

Menerima AS2 pesan di atas HTTPS

AWS Transfer Family AS2server saat ini hanya menyediakan HTTP transportasi melalui port 5080. Namun, Anda dapat menghentikan TLS penyeimbang beban di depan VPC endpoint server Transfer Family Anda dengan menggunakan port dan sertifikat pilihan Anda. Dengan pendekatan ini, Anda dapat menggunakan AS2 HTTPS pesan masuk.

Prasyarat

- VPC harus sama Wilayah AWS dengan server Transfer Family Anda.
- Subnet Anda VPC harus berada dalam Availability Zones tempat Anda ingin menggunakan server Anda.

Note

Setiap server Transfer Family dapat mendukung hingga tiga Availability Zone.

- Alokasikan hingga tiga alamat IP Elastis di Wilayah yang sama dengan server Anda. Atau, Anda dapat memilih untuk membawa rentang alamat IP Anda sendiri (BYOIP).

Note

Jumlah alamat IP Elastis harus sesuai dengan jumlah Availability Zone yang Anda gunakan dengan endpoint server Anda.

Konfigurasi Network Load Balancer

Siapkan Network Load Balancer NLB () yang menghadap ke internet di file Anda. VPC

Untuk membuat Network Load Balancer dan mendefinisikan VPC endpoint server sebagai target load balancer

1. Buka konsol Amazon Elastic Compute Cloud di <https://console.aws.amazon.com/ec2/>.
2. Dari panel navigasi, pilih Load Balancers, lalu pilih Create load balancer.
3. Di bawah Penyeimbang Beban Jaringan, pilih Buat.
4. Di bagian Konfigurasi dasar, masukkan informasi berikut:
 - Untuk Nama, masukkan nama deskriptif untuk penyeimbang beban.

- Untuk Skema, pilih yang menghadap ke Internet.
 - Untuk jenis alamat IP, pilih IPv4.
5. Di bagian Pemetaan jaringan, masukkan informasi berikut:
- Untuk VPC, pilih virtual private cloud (VPC) yang Anda buat.
 - Di bawah Pemetaan, pilih Availability Zones yang terkait dengan subnet publik yang tersedia sama dengan VPC yang Anda gunakan dengan endpoint server Anda.
 - Untuk IPv4alamat setiap subnet, pilih salah satu alamat IP elastis yang Anda alokasikan.
6. Di bagian Pendengar dan perutean, masukkan informasi berikut:
- Untuk Protokol, pilih TLS.
 - Untuk Port, masukkan **5080**.
 - Untuk tindakan Default, pilih Buat grup target. Untuk detail pembuatan grup target baru, lihat[Untuk membuat grup target](#).

Setelah Anda membuat grup target, masukkan namanya di bidang Tindakan default.

7. Di bagian Pengaturan pendengar aman, pilih sertifikat Anda di area SSLTLSDefault/certificate.
8. Pilih Buat penyeimbang beban untuk membuatNLB.
9. (Opsional, tetapi disarankan) Aktifkan log akses untuk Network Load Balancer untuk mempertahankan jejak audit penuh, seperti yang dijelaskan dalam [log Access untuk Network Load Balancer Anda](#).

Kami merekomendasikan langkah ini karena TLS koneksi diakhiri di. NLB Oleh karena itu, alamat IP sumber yang tercermin dalam grup AS2 CloudWatch log Transfer Family Anda adalah alamat IP pribadi, bukan alamat IP eksternal mitra dagang Anda. NLB

Setelah Anda mengatur penyeimbang beban, klien berkomunikasi dengan penyeimbang beban melalui pendengar port kustom. Kemudian, penyeimbang beban berkomunikasi dengan server melalui port 5080.

Untuk membuat grup target

1. Setelah Anda memilih Buat grup target dalam prosedur sebelumnya, Anda akan dibawa ke halaman Tentukan rincian grup untuk grup target baru.
2. Di bagian Konfigurasi dasar, masukkan informasi berikut.

- Untuk Pilih jenis target, pilih alamat IP.
 - Untuk Name, masukkan nama untuk grup target.
 - Untuk Protokol, pilih TCP.
 - Untuk Port, masukkan **5080**.
 - Untuk jenis alamat IP, pilih IPv4.
 - Untuk VPC, pilih VPC yang Anda buat untuk AS2 server Transfer Family Anda.
3. Di bagian Pemeriksaan Kesehatan, TCP pilih protokol pemeriksaan Kesehatan.
 4. Pilih Berikutnya.
 5. Pada halaman Daftar target, masukkan informasi berikut:
 - Untuk Jaringan, konfirmasikan VPC bahwa yang Anda buat untuk AS2 server Transfer Family Anda ditentukan.
 - Untuk IPv4 alamat, masukkan IPv4 alamat pribadi titik akhir AS2 server Transfer Family Anda.

Jika Anda memiliki lebih dari satu titik akhir untuk server Anda, pilih Tambah IPv4 alamat untuk menambahkan baris lain untuk memasukkan IPv4 alamat lain. Ulangi proses ini sampai Anda memasukkan alamat IP pribadi untuk semua titik akhir server Anda.

 - Pastikan bahwa Port diatur ke **5080**.
 - Pilih Sertakan sebagai tertunda di bawah ini untuk menambahkan entri Anda ke bagian Tinjau target.
 6. Di bagian Tinjau target, tinjau target IP Anda.
 7. Pilih Buat grup target, lalu kembali ke prosedur sebelumnya untuk membuat Anda NLB dan masukkan grup target baru di mana ditunjukkan.

Uji akses ke server dari alamat IP Elastis

Connect ke server melalui port kustom dengan menggunakan alamat IP Elastis atau DNS nama Network Load Balancer.

 Important

Kelola akses ke server Anda dari alamat IP klien dengan menggunakan [daftar kontrol akses jaringan \(jaringanACLs\)](#) untuk subnet yang dikonfigurasi pada penyeimbang beban. ACL izin jaringan ditetapkan pada tingkat subnet, sehingga aturan berlaku untuk semua sumber daya yang menggunakan subnet. Anda tidak dapat mengontrol akses dari alamat IP klien dengan

menggunakan grup keamanan, karena jenis target penyeimbang beban disetel ke alamat IP, bukan Instans. Oleh karena itu, penyeimbang beban tidak mempertahankan alamat IP sumber. Jika [pemeriksaan kesehatan Network Load Balancer](#) gagal, ini berarti penyeimbang beban tidak dapat terhubung ke titik akhir server. Untuk memecahkan masalah ini, periksa hal berikut:

- Konfirmasikan bahwa [grup keamanan terkait titik akhir](#) server memungkinkan koneksi masuk dari subnet yang dikonfigurasi pada penyeimbang beban. Load balancer harus dapat terhubung ke endpoint server melalui port 5080.
- Konfirmasikan bahwa Status server sedang Online.

Mentransfer file dengan menggunakan konektor AS2

AS2konektor membangun hubungan antara mitra dagang untuk transfer AS2 pesan dari server Transfer Family ke tujuan eksternal milik mitra.

Anda dapat menggunakan Transfer Family untuk mengirim AS2 pesan dengan mereferensikan ID konektor dan jalur ke file, seperti yang diilustrasikan dalam perintah `start-file-transfer` AWS Command Line Interface (AWS CLI) berikut:

```
aws transfer start-file-transfer --connector-id c-1234567890abcdef0 \  
--send-file-paths "/DOC-EXAMPLE-SOURCE-BUCKET/myfile1.txt" "/DOC-EXAMPLE-SOURCE-BUCKET/  
myfile2.txt"
```

Untuk mendapatkan detail konektor Anda, jalankan perintah berikut:

```
aws transfer list-connectors
```

`list-connectors` Perintah mengembalikan konektorIDs, URLs, dan Amazon Resource Names (ARNs) untuk konektor Anda.

Untuk mengembalikan properti konektor tertentu, jalankan perintah berikut dengan ID yang ingin Anda gunakan:

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

`describe-connector` Perintah mengembalikan semua properti untuk konektor, termasuk, peran, profilURL, Pemberitahuan Disposisi Pesan (MDNs), tag, dan metrik pemantauan.

Anda dapat mengonfirmasi bahwa mitra berhasil menerima file dengan melihat JSON dan MDN file. File-file ini diberi nama sesuai dengan konvensi yang dijelaskan dalam [Nama dan lokasi file](#). Jika Anda mengonfigurasi peran logging saat membuat konektor, Anda juga dapat memeriksa CloudWatch log untuk status AS2 pesan.

Untuk melihat detail AS2 konektor, lihat [Lihat detail AS2 konektor](#). Untuk informasi selengkapnya tentang membuat AS2 konektor, lihat [Konfigurasi AS2 konektor](#).

Nama dan lokasi file

Bagian ini membahas konvensi penamaan file untuk transfer. AS2

Untuk transfer file masuk, perhatikan hal berikut:

- Anda menentukan direktori dasar dalam perjanjian. Direktori dasar adalah nama bucket Amazon S3 yang dikombinasikan dengan awalan, jika ada. Misalnya, /DOC-EXAMPLE-BUCKET/AS2-folder.
- Jika file yang masuk berhasil diproses, file (dan JSON file yang sesuai) disimpan ke /processed folder. Misalnya, /DOC-EXAMPLE-BUCKET/AS2-folder/processed.

JSONFile berisi bidang-bidang berikut:

- 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`
- Jika file yang masuk tidak dapat diproses dengan sukses, file (dan JSON file yang sesuai) disimpan ke `/failed` folder. Misalnya, `/DOC-EXAMPLE-BUCKET/AS2-folder/failed`.
- File yang ditransfer disimpan dalam `processed` folder sebagai `original_filename.messageId.original_extension`. Artinya, ID pesan untuk transfer ditambahkan ke nama file, sebelum ekstensi aslinya.
- JSONFile dibuat dan disimpan sebagai `original_filename.messageId.original_extension.json`. Selain ID pesan yang ditambahkan, string `.json` ditambahkan ke nama file yang ditransfer.
- File Message Disposition Notice (MDN) dibuat dan disimpan sebagai `original_filename.messageId.original_extension.mdn`. Selain ID pesan yang ditambahkan, string `.mdn` ditambahkan ke nama file yang ditransfer.
- Jika ada file inbound bernama `ExampleFileInS3Payload.dat`, file-file berikut dibuat:
 - Berkas —
`ExampleFileInS3Payload.c4d6b6c7-23ea-4b8c-9ada-0cb811dc8b35@44313c54b0a46a36.`
 - JSON –
`ExampleFileInS3Payload.c4d6b6c7-23ea-4b8c-9ada-0cb811dc8b35@44313c54b0a46a36.`
 - MDN –
`ExampleFileInS3Payload.c4d6b6c7-23ea-4b8c-9ada-0cb811dc8b35@44313c54b0a46a36.`

Untuk transfer keluar, penamaannya serupa, dengan perbedaan bahwa tidak ada file pesan masuk, dan juga, ID transfer untuk pesan yang ditransfer ditambahkan ke nama file. ID transfer dikembalikan oleh `StartFileTransfer` API operasi (atau ketika proses atau skrip lain memanggil operasi ini).

- `transfer-id` ini adalah pengidentifikasi yang terkait dengan transfer file. Semua permintaan yang merupakan bagian dari `StartFileTransfer` panggilan berbagi `transfer-id`.
- Direktori dasar sama dengan jalur yang Anda gunakan untuk file sumber. Artinya, direktori dasar adalah jalur yang Anda tentukan dalam `StartFileTransfer` API operasi atau `start-file-transfer` AWS CLI perintah. Sebagai contoh:

```
aws transfer start-file-transfer --send-file-paths /DOC-EXAMPLE-BUCKET/AS2-folder/
file-to-send.txt
```

Jika Anda menjalankan perintah ini, MDN dan JSON file disimpan di /DOC-EXAMPLE-BUCKET/AS2-folder/processed (untuk transfer yang berhasil), atau /DOC-EXAMPLE-BUCKET/AS2-folder/failed (untuk transfer yang gagal).

- JSONFile dibuat dan disimpan sebagai `fileoriginal_filename.transferId.messageId.original_extension.json`.
- MDNFile dibuat dan disimpan sebagai `fileoriginal_filename.transferId.messageId.original_extension.mdn`.
- Jika ada file keluar bernama `ExampleFileOutTestOutboundSyncMdn.dat`, file-file berikut dibuat:
 - JSON – `ExampleFileOutTestOutboundSyncMdn.dedf4601-4e90-4043-b16b-579af35e0d83.fbe18db8-7361-42ff-8ab6-49ec1e435f34@c9c705f0baaaabaa.dat.json`
 - MDN – `ExampleFileOutTestOutboundSyncMdn.dedf4601-4e90-4043-b16b-579af35e0d83.fbe18db8-7361-42ff-8ab6-49ec1e435f34@c9c705f0baaaabaa.dat.mdn`

Anda juga dapat memeriksa CloudWatch log untuk melihat detail transfer Anda, termasuk yang gagal.

Kode status

Tabel berikut mencantumkan semua kode status yang dapat dicatat ke CloudWatch log saat Anda atau pasangan mengirim AS2 pesan. Langkah-langkah pemrosesan pesan yang berbeda berlaku untuk jenis pesan yang berbeda dan dimaksudkan untuk pemantauan saja. FAILEDStatus COMPLETED dan mewakili langkah terakhir dalam pemrosesan, dan terlihat dalam JSON file.

Kode	Deskripsi	Pemrosesan selesai?
PROCESSING	Pesan sedang dalam proses dikonversi ke format akhirnya. Misalnya, langkah dekompresi dan dekripsi keduanya memiliki status ini.	Tidak

Kode	Deskripsi	Pemrosesan selesai?
MDN_TRANSMIT	Pemrosesan pesan adalah mengirimkan MDN respons.	Tidak
MDN_RECEIVE	Pemrosesan pesan menerima MDN respons.	Tidak
COMPLETED	Pemrosesan pesan telah selesai dengan sukses. Status ini mencakup kapan MDN dikirim untuk pesan masuk atau untuk MDN verifikasi pesan keluar.	Ya
FAILED	Pemrosesan pesan telah gagal. Untuk daftar kode kesalahan, lihat Kode kesalahan AS2 .	Ya

Contoh JSON file

Bagian ini mencantumkan JSON file sampel untuk transfer masuk dan keluar, termasuk file sampel untuk transfer yang berhasil dan transfer yang gagal.

Contoh file keluar yang berhasil ditransfer:

```
{
  "requester-content-type": "application/octet-stream",
  "message-subject": "File xyzTest from MyCompany_0ID to partner YourCompany",
  "requester-file-name": "TestOutboundSyncMdn-9lmCr79hV.dat",
  "as2-from": "MyCompany_0ID",
  "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_0ID_MyCompany_0ID",
  "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/as2testcell10000/openAs2/
TestOutboundSyncMdn-9lmCr79hV.dat",
"as2-message-id": "fbe18db8-7361-42ff-8ab6-49ec1e435f34@c9c705f0baaaabaa",
"timestamp": "2022-07-11T06:30:10.791274Z"
}

```

Contoh file keluar yang tidak berhasil ditransfer:

```

{
  "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/as2IntegCell10002/openAs2/
openAs2TestOutboundWrongAs2Ids-necco-3VYn5n8wE.dat",
  "timestamp": "2022-07-11T21:17:24.802378Z"
}

```

Contoh file inbound yang berhasil ditransfer:

```

{
  "requester-content-type": "application/EDI-X12",
  "subject": "File openAs2TestInboundAsyncMdn-necco-5Ab6bTfC0.dat sent from MyCompany
to PartnerA",
  "client-ip": "10.0.109.105",
  "requester-file-name": "openAs2TestInboundAsyncMdn-necco-5Ab6bTfC0.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-5Ab6bTfC0.OPENAS2-11072022233606+0000-5dab0452-0ca1-4f9b-b622-
fba84effff3c@MyCompany_OID_PartnerA_OID.dat",
  "server-id": "s-5f7422b04c2447ef9",
  "timestamp": "2022-07-11T23:36:36.105030Z"
}
```

Contoh file masuk yang tidak berhasil ditransfer:

```
{
  "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"
}
```

Memantau penggunaan AS2

Anda dapat memantau aktivitas AS2 menggunakan Amazon CloudWatch dan AWS CloudTrail. Untuk melihat metrik server Transfer Family lainnya, lihat [CloudWatch Pencatatan Amazon untuk AWS Transfer Family](#)

Metrik AS2

Metrik	Deskripsi
InboundMessage	Jumlah total pesan AS2 yang berhasil diterima dari mitra dagang.
	Unit: Hitungan

Metrik	Deskripsi
	Periode: 5 menit
InboundFailedMessage	Jumlah total pesan AS2 yang tidak berhasil diterima dari mitra dagang. Artinya, mitra dagang mengirim pesan, tetapi server Transfer Family tidak berhasil memprosesnya. Unit: Hitungan Periode: 5 menit
OutboundMessage	Jumlah total pesan AS2 yang berhasil dikirim dari server Transfer Family ke mitra dagang. Unit: Hitungan Periode: 5 menit
OutboundFailedMessage	Jumlah total pesan AS2 yang tidak berhasil dikirim ke mitra dagang. Artinya, mereka dikirim dari server Transfer Family, tetapi tidak berhasil diterima oleh mitra dagang. Unit: Hitungan Periode: 5 menit

Kode Status AS2

Tabel berikut mencantumkan semua kode status yang dapat dicatat ke CloudWatch log saat Anda atau pasangan Anda mengirim pesan AS2. Langkah-langkah pemrosesan pesan yang berbeda berlaku untuk jenis pesan yang berbeda dan dimaksudkan untuk pemantauan saja. Status COMPLETED dan FAILED mewakili langkah terakhir dalam pemrosesan, dan terlihat dalam file JSON.

Kode	Deskripsi	Pemrosesan selesai?
PENGOLAHAN	Pesan sedang dalam proses dikonversi ke format akhirnya. Misalnya, langkah dekompresi dan dekripsi keduanya memiliki status ini.	Tidak
MDN_TRANSMIT	Pemrosesan pesan adalah mengirimkan respons MDN.	Tidak
MDN_RECEIVE	Pemrosesan pesan menerima respons MDN.	Tidak
DISELESAIKAN	Pemrosesan pesan telah selesai dengan sukses. Keadaan ini termasuk ketika MDN dikirim untuk pesan masuk atau untuk verifikasi MDN dari pesan keluar.	Ya
Failed	Pemrosesan pesan telah gagal. Untuk daftar kode kesalahan, lihat Kode kesalahan AS2 .	Ya

Kode kesalahan AS2

Tabel berikut mencantumkan dan menjelaskan kode kesalahan yang mungkin Anda terima dari transfer file AS2.

Kode kesalahan AS2

Kode	Kesalahan	Deskripsi dan resolusi
ACCESS_DENIED	Akses ditolak. Periksa apakah peran akses Anda	Terjadi saat menangani <code>StartFileTransfer</code> permintaan di mana salah satu

Kode	Kesalahan	Deskripsi dan resolusi
	memiliki izin yang diperlukan. - Jalur berkas tidak valid <i>send-file-path</i> <i>Gagal mendapatkan kredensial dengan ErrorCode: kode kesalahan</i>	dari <code>SendFilePaths</code> yang tidak valid atau cacat. Artinya, jalur tersebut tidak memiliki nama bucket Amazon S3, atau jalurnya menyertakan karakter yang tidak valid. Juga terjadi jika Transfer Family gagal mengambil peran akses atau peran logging. Pastikan jalur berisi nama bucket Amazon S3 dan nama kunci yang valid.
AGREEMENT_NOT_FOUND	Kesepakatan tidak ditemukan.	Entah perjanjian itu tidak ditemukan, atau perjanjian dikaitkan dengan profil yang tidak aktif. Perbarui perjanjian dalam server Transfer Family untuk menyertakan profil aktif.
CONNECTOR_NOT_FOUND	Konektor atau konfigurasi terkait tidak ditemukan.	Entah konektor tidak ditemukan, atau konektor dikaitkan dengan profil yang tidak aktif. Perbarui konektor untuk menyertakan profil aktif.

Kode	Kesalahan	Deskripsi dan resolusi
CREDENTIALS_RETRIEVAL_FAILED	1. Rahasia tidak ditemukan di Secrets Manager. 2. Tidak dapat mengakses Secrets Manager. 3. Gagal mendekripsi rahasia di Secrets Manager. 4. Tidak bisa mendapatkan nilai rahasia karena pelambatan.	Untuk otentikasi AS2 Basic, rahasianya harus diformat dengan benar. Resolusi berikut sesuai dengan kesalahan yang tercantum di kolom sebelumnya. 1. Pastikan bahwa ID rahasia sudah benar. 2. Pastikan bahwa peran akses memiliki izin yang sesuai untuk membaca rahasia. Peran akses harus menyediakan akses baca dan tulis ke direktori induk dari lokasi file yang digunakan dalam <code>StartFileTransfer</code> permintaan. Selain itu, pastikan bahwa peran tersebut menyediakan akses baca dan tulis ke direktori induk dari file yang ingin Anda kirim <code>StartFileTransfer</code>. 3. Jika kunci yang dikelola pelanggan digunakan untuk rahasia, pastikan bahwa peran akses memiliki izin untuk kunci AWS Key Management Service (AWS KMS).

Kode	Kesalahan	Deskripsi dan resolusi
		4. Untuk kuota yang berlaku, lihat Kuota untuk menangani rahasia .
DECOMPRESSION_FAILED	Gagal mendekompresi pesan.	Entah file yang dikirim rusak, atau algoritma kompresi tidak valid. Kirim ulang pesan dan verifikasi bahwa kompresi ZLIB digunakan, atau kirim ulang pesan tanpa kompresi diaktifkan.
DECRYPT_FAILED	<i>Gagal mendekripsi pesan Message-ID.</i> Pastikan bahwa mitra memiliki kunci enkripsi publik yang benar.	Dekripsi gagal. Konfirmasikan bahwa mitra mengirim muatan dengan menggunakan sertifikat yang valid dan enkripsi itu dilakukan dengan menggunakan algoritma enkripsi yang valid.
DECRYPT_FAILED_INVALID_SMIME_FORMAT	Tidak dapat mengurai mimePart yang diselimuti.	Payload MIME rusak atau dalam format SMIME yang tidak didukung. Pengirim harus memastikan bahwa format yang mereka gunakan didukung, dan kemudian mengirim ulang payload.

Kode	Kesalahan	Deskripsi dan resolusi
DECRYPT_FAILED_NO_DECRYPTION_KEY_FOUND	Tidak ditemukan kunci dekripsi yang cocok.	Profil mitra tidak memiliki sertifikat yang ditetapkan yang cocok dengan pesan, atau sertifikat yang cocok dengan pesan sekarang kedaluwarsa atau tidak lagi valid. Anda harus memperbarui profil mitra dan memastikan bahwa itu berisi sertifikat yang valid.
DECRYPT_FAILED_UNSUPPORTED_ENCRYPTION_ALG	<i>Dekripsi Muatan SMIME diminta menggunakan algoritma yang tidak didukung dengan ID: Encryption-ID.</i>	Pengirim jarak jauh telah mengirim muatan AS2 dengan algoritma enkripsi yang tidak didukung. Pengirim harus memilih algoritma enkripsi yang didukung oleh AWS Transfer Family.
DUPLICATE_MESSAGE	Langkah duplikat atau proses ganda.	Muatan memiliki langkah pemrosesan duplikat. Misalnya, ada dua langkah enkripsi. Kirim ulang pesan dengan satu langkah untuk penandatanganan, kompresi, dan enkripsi.

Kode	Kesalahan	Deskripsi dan resolusi
ENCRYPT_FAILED_NO_ENCRYPTION_KEY_FOUND	Tidak ada sertifikat enkripsi publik yang valid yang ditemukan di profil: <i>Local-profile-ID</i>	Transfer Family mencoba mengenkripsi pesan keluar, tetapi tidak ada sertifikat enkripsi yang ditemukan untuk profil lokal. Opsi resolusi: • Pastikan bahwa profil lokal memiliki sertifikat dan kunci pribadi untuk enkripsi terlampir. • Pastikan sertifikat enkripsi saat ini aktif.
ENCRYPTION_FAILED	Gagal mengenkripsi <i>nama</i> file.	File yang akan dikirim tidak tersedia untuk enkripsi. Verifikasi bahwa file tersebut berada di lokasi AS2 yang diharapkan dan yang AWS Transfer Family memiliki izin untuk membaca file.
FILE_SIZE_TOO_LARGE	Ukuran file terlalu besar.	Ini terjadi saat mengirim atau menerima file yang melebihi batas ukuran file.

Kode	Kesalahan	Deskripsi dan resolusi
HTTP_ERROR_RESPONSE_FROM_PARTNER	<i>Partner-URL mengembalikan status 400 untuk pesan dengan ID=Message-ID.</i>	Berkomunikasi dengan server AS2 mitra mengembalikan kode respons HTTP yang tidak terduga. Mitra mungkin dapat memberikan lebih banyak diagnostik dari log server AS2 mereka.
INSUFFICIENT_MESSAGE_SECURITY_UNENCRYPTED	Enkripsi diperlukan.	Mitra mengirim pesan yang tidak terenkripsi ke Transfer Family, yang tidak didukung. Pengirim harus menggunakan muatan terenkripsi.
INVALID_ENDPOINT_PROTOCOL	Hanya HTTP dan HTTPS yang didukung.	Anda harus menentukan HTTP atau HTTPS sebagai protokol dalam konfigurasi konektor AS2 Anda.

Kode	Kesalahan	Deskripsi dan resolusi
INVALID_REQUEST	1. Ada masalah dengan header pesan. 2. Tidak dapat mengurai JSON rahasia. Secret JSON tidak cocok dengan format yang diharapkan. 3. Rahasia harus berupa string JSON. 4. Nama pengguna tidak boleh mengandung titik dua. Nama pengguna tidak boleh mengandung karakter kontrol. Nama pengguna harus hanya berisi karakter ASCII. Kata sandi tidak boleh mengandung karakter kontrol. Kata sandi harus hanya berisi karakter ASCII.	Kesalahan ini memiliki beberapa penyebab. Resolusi berikut sesuai dengan kesalahan yang tercantum di kolom sebelumnya. 1. Periksa as2-from dan as2-to bidang. Pastikan ID pesan asli akurat untuk format MDN. Pastikan juga bahwa format ID pesan tidak hilang header AS2. 2. Pastikan bahwa nilai rahasia cocok dengan format yang didokumentasikan, seperti yang dijelaskan dalam Aktifkan otentikasi dasar untuk konektor AS2. 3. Pastikan bahwa rahasia disediakan sebagai string, dan bukan sebagai biner. 4. Lakukan koreksi yang diperlukan untuk nama pengguna atau kata sandi.

Kode	Kesalahan	Deskripsi dan resolusi
INVALID_URL_FORMAT	<i>Format URL tidak valid: URL</i>	Ini terjadi ketika Anda mengirim pesan keluar menggunakan konektor yang dikonfigurasi dengan URL yang salah bentuk. Pastikan konektor dikonfigurasi dengan URL HTTP atau HTTPS yang valid.
MDN_RESPONSE_INDICATES_AUTHENTICATION_FAILED	Tidak berlaku	Penerima tidak dapat mengautentikasi pengirim. Mitra dagang mengembalikan MDN ke Transfer Family dengan pengubah disposisi Error: authentication-failed.
MDN_RESPONSE_INDICATES_DECOMPRESSION_FAILED	Tidak berlaku	Ini terjadi ketika penerima tidak dapat mendekomposisi isi pesan. Mitra dagang mengembalikan MDN ke Transfer Family dengan pengubah disposisi Error: decompression-failed.
MDN_RESPONSE_INDICATES_DECRYPTION_FAILED	Tidak berlaku	Penerima tidak dapat mendekripsi isi pesan. Mitra dagang mengembalikan MDN ke Transfer Family dengan pengubah disposisi Error: authentication-failed.

Kode	Kesalahan	Deskripsi dan resolusi
MDN_RESPONSE_INDICATES_INSUFFICIENT_MESSAGE_SECURITY	Tidak berlaku	Penerima mengharapkan pesan ditandatangani atau dienkripsi, tetapi sebenarnya tidak. Mitra dagang mengembalikan MDN ke Transfer Family dengan pengubah disposisi Kesalahan :: insufficient-message-security Aktifkan penandatanganan dan/atau enkripsi pada konektor agar sesuai dengan harapan mitra dagang.
MDN_RESPONSE_INDICATES_INTEGRITY_CHECK_FAILED	Tidak berlaku	Penerima tidak dapat memverifikasi integritas konten. Mitra dagang mengembalikan MDN ke Transfer Family dengan pengubah disposisi Kesalahan :: integrity-check-failed
PATH_NOT_FOUND	Tidak dapat membuat jalur <i>berkas direktori</i> . Jalur induk tidak dapat ditemukan.	Transfer Family mencoba membuat direktori di bucket Amazon S3 pelanggan, tetapi bucket tidak ditemukan. Pastikan bahwa setiap jalur yang disebutkan dalam StartFileTransfer perintah berisi nama bucket yang ada.

Kode	Kesalahan	Deskripsi dan resolusi
SEND_FILE_NOT_FOUND	Jalur file jalur <i>file tidak ditemukan</i> .	Transfer Family tidak dapat menemukan file dalam operasi kirim file. Periksa apakah direktori dan jalur home yang dikonfigurasikan valid dan Transfer Family telah membaca izin untuk file tersebut.
SERVER_NOT_FOUND	Server yang terkait dengan pesan tidak dapat ditemukan.	Transfer Family tidak dapat menemukan server saat menerima pesan. Hal ini dapat terjadi jika server dihapus selama pemrosesan pesan masuk.
SERVER_NOT_ONLINE	Server <i>Server-ID tidak online</i> .	Server Transfer Family sedang offline. Mulai server sehingga dapat menerima dan memproses pesan.
SIGNING_FAILED	Gagal menandatangani berkas.	File yang akan dikirim tidak tersedia untuk ditandatangani, atau penandatanganan tidak dapat dilakukan. Verifikasi bahwa file tersebut berada di lokasi AS2 yang diharapkan dan yang AWS Transfer Family memiliki izin untuk membaca file.

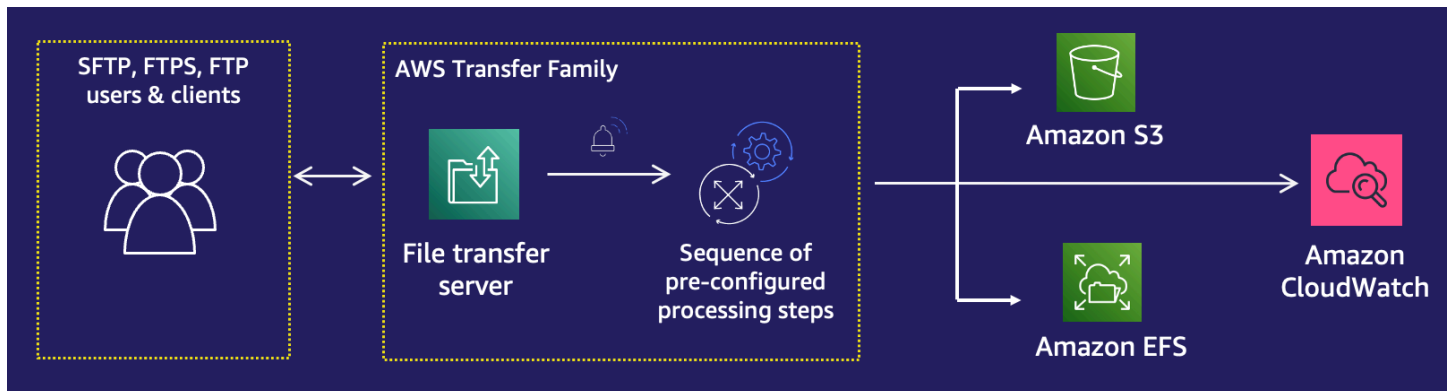
Kode	Kesalahan	Deskripsi dan resolusi
SIGNING_FAILED_NO_SIGNING_KEY_FOUND	Tidak ada sertifikat yang ditemukan untuk profil: <i>Local-profile-ID</i> .	Mencoba menandatangani pesan keluar, tetapi tidak ada sertifikat penandatanganan yang ditemukan untuk profil lokal. Opsi resolusi: • Pastikan bahwa profil lokal memiliki sertifikat dan kunci pribadi untuk penandatanganan terlampir. • Pastikan sertifikat penandatanganan saat ini aktif.
UNABLE_RESOLVE_HOST_TO_IP_ADDRESS	Tidak dapat menyelesaikan nama host ke alamat IP.	Transfer Family tidak dapat melakukan resolusi alamat DNS ke IP pada server DNS publik yang dikonfigurasi di konektor AS2. Perbarui konektor untuk menunjuk ke URL mitra yang valid.
UNABLE_TO_CONNECT_TO_REMOTE_HOST_OR_IP	Koneksi ke titik akhir habis waktu.	Transfer Family tidak dapat membuat koneksi socket ke server AS2 mitra yang dikonfigurasi. Periksa apakah server AS2 mitra tersedia di alamat IP yang dikonfigurasi.

Kode	Kesalahan	Deskripsi dan resolusi
UNABLE_TO_RESOLVE_HOSTNAME	Tidak dapat menyelesaikan nama host nama <i>host</i> .	Server Transfer Family tidak dapat menyelesaikan nama host mitra dengan menggunakan server DNS publik. Periksa apakah host yang dikonfigurasi terdaftar dan catatan DNS memiliki waktu untuk mempublikasikan.
VERIFICATION_FAILED	Verifikasi tanda tangan gagal untuk <i>ID pesan</i> AS2 atau kode MIC tidak cocok.	Periksa apakah sertifikat penandatanganan pengirim cocok dengan sertifikat penandatanganan untuk profil jarak jauh. Periksa juga apakah algoritma MIC kompatibel dengan AWS Transfer Family.

Kode	Kesalahan	Deskripsi dan resolusi
VERIFICATION_FAILED_NO_MATCHING_KEY_FOUND	- Tidak ada tanda tangan pesan pencocokan sertifikat publik yang dapat ditemukan di profil: <i>Partner-profile-ID</i> . <i>Tidak bisa mendapatkan sertifikat untuk profil yang tidak ada: Partner-profile-ID.</i> - Tidak ada sertifikat yang valid ditemukan di profil: <i>Partner-profile-ID</i> .	AWS Transfer Family sedang mencoba memverifikasi tanda tangan untuk pesan yang diterima, tetapi tidak ada sertifikat penandatanganan yang cocok ditemukan untuk profil mitra. Opsi resolusi: - Pastikan bahwa profil mitra memiliki sertifikat penandatanganan terlampir. - Pastikan sertifikat saat ini aktif. - Pastikan bahwa sertifikat adalah sertifikat penandatanganan yang benar untuk mitra.

AWS Transfer Family alur kerja terkelola

AWS Transfer Family mendukung alur kerja terkelola untuk pemrosesan file. Dengan alur kerja terkelola, Anda dapat memulai alur kerja setelah file ditransfer SFTP, FTPS, atau FTP. Dengan menggunakan fitur ini, Anda dapat dengan aman dan hemat biaya memenuhi persyaratan kepatuhan Anda untuk pertukaran file business-to-business (B2B) dengan mengoordinasikan semua langkah yang diperlukan untuk pemrosesan file. Selain itu, Anda mendapat manfaat dari end-to-end audit dan visibilitas.



Dengan mengatur tugas pemrosesan file, alur kerja terkelola membantu Anda memproses data sebelum dikonsumsi oleh aplikasi hilir. Tugas pemrosesan file tersebut mungkin termasuk:

- Memindahkan file ke folder khusus pengguna.
- Mendekripsi file sebagai bagian dari alur kerja.
- Menandai file.
- Melakukan pemrosesan kustom dengan membuat dan melampirkan AWS Lambda fungsi ke alur kerja.
- Mengirim pemberitahuan ketika file telah berhasil ditransfer. (Untuk posting blog yang merinci kasus penggunaan ini, lihat [Menyesuaikan pemberitahuan pengiriman file menggunakan alur kerja AWS Transfer Family terkelola](#).)

Untuk dengan cepat mereplikasi dan menstandarisasi tugas pemrosesan file pasca-unggah umum yang mencakup beberapa unit bisnis di organisasi Anda, Anda dapat menerapkan alur kerja dengan menggunakan infrastruktur sebagai kode (IaC). Anda dapat menentukan alur kerja terkelola yang akan dimulai pada file yang diunggah secara penuh. Anda juga dapat menentukan alur kerja terkelola yang berbeda untuk dimulai pada file yang hanya diunggah sebagian karena pemutusan sesi prematur. Penanganan pengecualian bawaan membantu Anda bereaksi dengan cepat terhadap hasil

pemrosesan file, sambil menawarkan Anda kontrol atas cara menangani kegagalan. Selain itu, setiap langkah alur kerja menghasilkan log terperinci, yang dapat Anda audit untuk melacak garis keturunan data.

Untuk memulai, lakukan tugas-tugas berikut:

1. Siapkan alur kerja Anda agar berisi tindakan pra-pemrosesan, seperti menyalin, menandai, dan langkah-langkah lain berdasarkan kebutuhan Anda. Lihat [Buat alur kerja](#) untuk detail.
2. Konfigurasi peran eksekusi, yang digunakan Transfer Family untuk menjalankan alur kerja. Lihat [IAMkebijakan untuk alur kerja](#) untuk detail.
3. Petakan alur kerja ke server, sehingga pada saat kedatangan file, tindakan yang ditentukan dalam alur kerja ini dievaluasi dan dimulai secara real time. Lihat [Konfigurasi dan jalankan alur kerja](#) untuk detail.

Informasi terkait

- Untuk memantau eksekusi alur kerja Anda, lihat. [Menggunakan CloudWatch metrik untuk Transfer Family](#)
- Untuk detail log eksekusi dan informasi pemecahan masalah, lihat. [Memecahkan masalah kesalahan terkait alur kerja menggunakan Amazon CloudWatch](#)
- Kami memiliki lokakarya yang dapat Anda hadir, di mana Anda dapat membangun solusi transfer file. Solusi ini memanfaatkan AWS Transfer Family FTPS titik akhir SFTP terkelola/dan Amazon Cognito dan DynamoDB untuk manajemen pengguna. Anda dapat melihat detail untuk lokakarya ini [di sini](#).
- Lihat [Alur Kerja AWS Transfer Family Terkelola](#) untuk pengenalan singkat tentang alur kerja Transfer Family.

Topik

- [Buat alur kerja](#)
- [Gunakan langkah-langkah yang telah ditentukan](#)
- [Gunakan langkah-langkah pemrosesan file khusus](#)
- [IAMkebijakan untuk alur kerja](#)
- [Penanganan pengecualian untuk alur kerja](#)
- [Pantau eksekusi alur kerja](#)
- [Buat alur kerja dari template](#)

- [Menghapus alur kerja dari server Transfer Family](#)
- [Pembatasan dan batasan alur kerja yang dikelola](#)

Untuk bantuan selengkapnya untuk memulai alur kerja terkelola, lihat sumber daya berikut:

- AWS Transfer Family video demo [alur kerja terkelola](#)
- [Membangun platform transfer file cloud-native menggunakan AWS Transfer Family alur kerja posting blog](#)

Buat alur kerja

Anda dapat membuat alur kerja terkelola menggunakan AWS Management Console, seperti yang dijelaskan dalam topik ini. Untuk membuat proses pembuatan alur kerja semudah mungkin, panel bantuan kontekstual tersedia untuk sebagian besar bagian di konsol.

Alur kerja memiliki dua jenis langkah:

- Langkah nominal — Langkah nominal adalah langkah pemrosesan file yang ingin Anda terapkan ke file yang masuk. Jika Anda memilih lebih dari satu langkah nominal, setiap langkah diproses dalam urutan linier.
- Langkah penanganan pengecualian - Penangan pengecualian adalah langkah pemrosesan file yang AWS Transfer Family dijalankan jika ada langkah nominal yang gagal atau mengakibatkan kesalahan validasi.

Buat alur kerja

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Di panel navigasi kiri, pilih Alur kerja.
3. Pada halaman Alur kerja, pilih Buat alur kerja.
4. Pada halaman Buat alur kerja, masukkan deskripsi. Deskripsi ini muncul di halaman Alur Kerja.
5. Di bagian Langkah nominal, pilih Tambah langkah. Tambahkan satu atau lebih langkah.
 - a. Pilih jenis langkah dari opsi yang tersedia. Untuk informasi selengkapnya tentang berbagai jenis langkah, lihat [the section called “Gunakan langkah-langkah yang telah ditentukan”](#).
 - b. Pilih Berikutnya, lalu konfigurasi parameter untuk langkah tersebut.

- c. Pilih Berikutnya, lalu tinjau detail untuk langkahnya.
- d. Pilih Buat langkah untuk menambahkan langkah dan melanjutkan.
- e. Lanjutkan menambahkan langkah-langkah sesuai kebutuhan. Jumlah maksimum langkah dalam alur kerja adalah 8.
- f. Setelah Anda menambahkan semua langkah nominal yang diperlukan, gulir ke bawah ke penanganan Exception - bagian opsional, dan pilih Tambah langkah.

 Note

Agar Anda diberitahu tentang kegagalan secara real time, kami sarankan Anda menyiapkan penanganan pengecualian dan langkah-langkah untuk mengeksekusi ketika alur kerja Anda gagal.

6. Untuk mengonfigurasi penanganan pengecualian, tambahkan langkah dengan cara yang sama seperti yang dijelaskan sebelumnya. Jika file menyebabkan langkah apa pun untuk melempar pengecualian, penanganan pengecualian Anda dipanggil satu per satu.
7. (Opsional) Gulir ke bawah ke bagian Tag, dan tambahkan tag untuk alur kerja Anda.
8. Tinjau konfigurasi, dan pilih Buat alur kerja.

 Important

Setelah membuat alur kerja, Anda tidak dapat mengeditnya, jadi pastikan untuk meninjau konfigurasi dengan cermat.

Konfigurasikan dan jalankan alur kerja

Sebelum Anda dapat menjalankan alur kerja, Anda harus mengaitkannya dengan server Transfer Family.

Untuk mengonfigurasi Transfer Family untuk menjalankan alur kerja pada file yang diunggah

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Di panel navigasi kiri, pilih Server.
 - Untuk menambahkan alur kerja ke server yang ada, pilih server yang ingin Anda gunakan untuk alur kerja Anda.

- Atau, buat server baru dan tambahkan alur kerja ke dalamnya. Untuk informasi selengkapnya, lihat [Mengkonfigurasi titik SFTP akhir,FTPS, atau FTP server](#).
3. Pada halaman detail untuk server, gulir ke bawah ke bagian Detail tambahan, lalu pilih Edit.

Note

Secara default, server tidak memiliki alur kerja terkait. Anda menggunakan bagian Detail tambahan untuk mengaitkan alur kerja dengan server yang dipilih.

4. Pada halaman Edit detail tambahan, di bagian Alur kerja terkelola, pilih alur kerja yang akan dijalankan di semua unggahan.

Note

Jika Anda belum memiliki alur kerja, pilih Buat Alur Kerja baru untuk membuatnya.

- a. Pilih ID alur kerja yang akan digunakan.
- b. Pilih peran eksekusi. Ini adalah peran yang diasumsikan Transfer Family saat menjalankan langkah-langkah alur kerja. Untuk informasi selengkapnya, lihat [IAMkebijakan untuk alur kerja](#). Pilih Simpan.

Managed workflows [Info](#)

Workflow for complete file uploads
Select the workflow that AWS Transfer Family should run on all files that are uploaded in full via this server

w-

Workflow for partial file uploads
Select the workflow that Transfer Family should run on all files that are only partially uploaded via this server

w-

Managed workflows execution role [Info](#)
Select the role that AWS Transfer Family should assume when executing a workflow

Note

Jika Anda tidak lagi ingin alur kerja dikaitkan dengan server, Anda dapat menghapus asosiasi tersebut. Untuk detailnya, lihat [Menghapus alur kerja dari server Transfer Family](#).

Untuk menjalankan alur kerja

Untuk menjalankan alur kerja, Anda mengunggah file ke server Transfer Family yang dikonfigurasi dengan alur kerja terkait.

Note

Setiap kali Anda menghapus alur kerja dari server dan menggantinya dengan yang baru, atau memperbarui konfigurasi server (yang memengaruhi peran eksekusi alur kerja), Anda harus menunggu sekitar 10 menit sebelum menjalankan alur kerja baru. Server Transfer Family menyimpan cache detail alur kerja, dan dibutuhkan waktu 10 menit bagi server untuk menyegarkan cache-nya.

Selain itu, Anda harus keluar dari SFTP sesi aktif apa pun, dan kemudian masuk kembali setelah masa tunggu 10 menit untuk melihat perubahannya.

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
>
```

Setelah file Anda diunggah, tindakan yang ditentukan dilakukan pada file Anda. Misalnya, jika alur kerja Anda berisi langkah penyalinan, file tersebut disalin ke lokasi yang Anda tentukan di langkah itu. Anda dapat menggunakan Amazon CloudWatch Logs untuk melacak langkah-langkah yang dijalankan dan status eksekusi mereka.

Lihat detail alur kerja

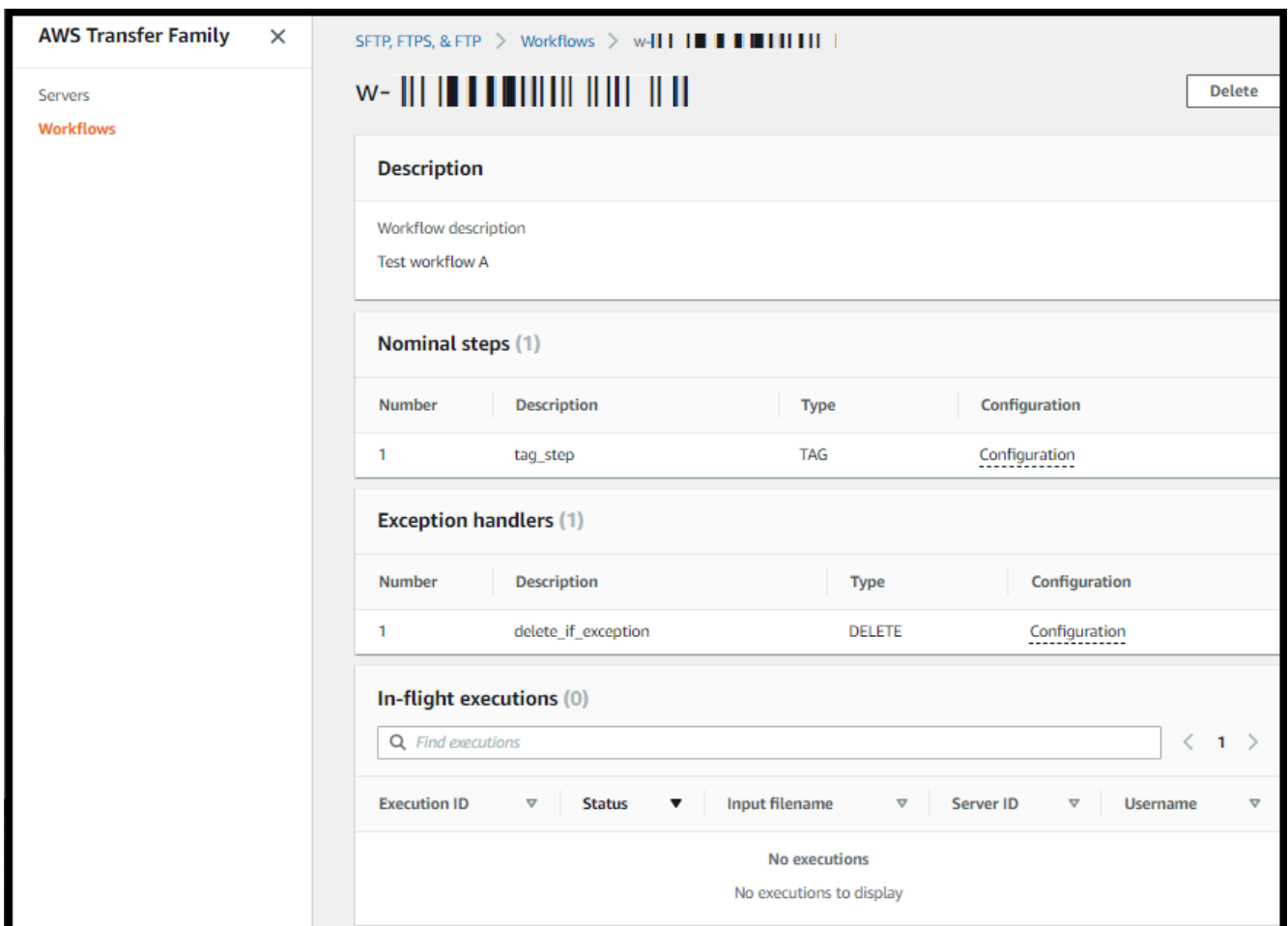
Anda dapat melihat detail tentang alur kerja yang dibuat sebelumnya atau eksekusi alur kerja. Untuk melihat detail ini, Anda dapat menggunakan konsol atau AWS Command Line Interface (AWS CLI).

Console

Lihat detail alur kerja

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Di panel navigasi kiri, pilih Alur kerja.
3. Pada halaman Alur Kerja, pilih alur kerja.

Halaman detail alur kerja terbuka.



CLI

Untuk melihat detail alur kerja, gunakan `describe-workflow` CLI perintah, seperti yang ditunjukkan pada contoh berikut. Ganti ID alur kerja `w-1234567890abcdef0` dengan nilai Anda sendiri. Untuk informasi selengkapnya, lihat [deskripsikan alur kerja di Referensi Perintah.AWS CLI](#)

```
# 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": {}
  }
}
```

Jika alur kerja Anda dibuat sebagai bagian dari AWS CloudFormation tumpukan, Anda dapat mengelola alur kerja menggunakan AWS CloudFormation konsol (<https://console.aws.amazon.com/cloudformation>).

Transfer Family > Workflows > workflow-32062074

Workflow-32062074 Delete

Info This workflow belongs to the AWS CloudFormation stack **WorkflowStack**. [Manage this stack](#) on the CloudFormation console.

Description

Workflow description

-

Nominal steps (1) [Info](#)

Number	Description	Type	Configuration
1	tagFileForArchive	TAG	Details

Exception handlers (0) [Info](#)

Number	Description	Type	Configuration

Gunakan langkah-langkah yang telah ditentukan

Saat membuat alur kerja, Anda dapat memilih untuk menambahkan salah satu langkah yang telah ditentukan berikut yang dibahas dalam topik ini. Anda juga dapat memilih untuk menambahkan langkah-langkah pemrosesan file kustom Anda sendiri. Untuk informasi selengkapnya, lihat [the section called “Gunakan langkah-langkah pemrosesan file khusus”](#).

Topik

- [Salin berkas](#)
- [Dekripsi file](#)
- [Berkas tag](#)
- [Hapus berkas](#)
- [Variabel bernama untuk alur kerja](#)
- [Contoh tag dan hapus alur kerja](#)

Salin berkas

Langkah salin file membuat salinan file yang diunggah di lokasi Amazon S3 baru. Saat ini, Anda dapat menggunakan langkah salin file hanya dengan Amazon S3.

Langkah copy file berikut menyalin file ke test folder di bucket file-test tujuan.

Jika langkah salin file bukan langkah pertama alur kerja Anda, Anda dapat menentukan lokasi File. Dengan menentukan lokasi file, Anda dapat menyalin file yang digunakan pada langkah sebelumnya atau file asli yang diunggah. Anda dapat menggunakan fitur ini untuk membuat beberapa salinan dari file asli sambil menjaga file sumber tetap utuh untuk arsip file dan penyimpanan catatan. Sebagai contoh, lihat [Contoh tag dan hapus alur kerja](#).

Configure copy parameters

Step name

File location

Select the file location to use as an input for this step

☒ Copy the file created from previous step to a new location

Input file is selected from the previous step's output

☐ Copy the original source file to a new location

Originally uploaded file

Destination bucket name

Destination key prefix

If you are copying files into a folder, specify / at the end of the prefix name. Use `${transfer:UserName}` or `${transfer:UploadDate}` to parametrize destination prefix by username or upload date respectively.

☐ Overwrite existing

Berikan ember dan detail kunci

Anda harus memberikan nama bucket dan kunci untuk tujuan langkah salin file. Kuncinya bisa berupa nama jalur atau nama file. Apakah kunci diperlakukan sebagai nama jalur atau nama file ditentukan oleh apakah Anda mengakhiri kunci dengan karakter garis miring (/) maju.

Jika karakter terakhir adalah /, file Anda disalin ke folder, dan namanya tidak berubah. Jika karakter terakhir adalah alfanumerik, file yang Anda unggah diubah namanya menjadi nilai kunci. Dalam hal

ini, jika file dengan nama itu sudah ada, perilaku tergantung pada pengaturan untuk bidang Timpa yang ada.

- Jika Timpa yang ada dipilih, file yang ada diganti dengan file yang sedang diproses.
- Jika Timpa yang ada tidak dipilih, tidak ada yang terjadi, dan pemrosesan alur kerja berhenti.

Tip

Jika penulisan bersamaan dijalankan pada jalur file yang sama, hal itu dapat mengakibatkan perilaku yang tidak terduga saat menimpa file.

Misalnya, jika nilai kunci Anda adalah `test/`, file yang Anda unggah akan disalin ke folder. `test` Jika nilai kunci Anda `test/today`, (dan Timpa yang ada dipilih) setiap file yang Anda unggah disalin ke file bernama `today` di `test` folder, dan setiap file berikutnya menimpa yang sebelumnya.

Note

Amazon S3 mendukung bucket dan objek, dan tidak memiliki hierarki. Namun, Anda dapat menggunakan awalan dan pembatas dalam nama kunci objek untuk menyiratkan hierarki dan mengatur data Anda dengan cara yang mirip dengan folder.

Gunakan variabel bernama dalam langkah copy file

Dalam langkah salin file, Anda dapat menggunakan variabel untuk menyalin file Anda secara dinamis ke folder khusus pengguna. Saat ini, Anda dapat menggunakan `${transfer:UserName}` atau `${transfer:UploadDate}` sebagai variabel untuk menyalin file ke lokasi tujuan untuk pengguna tertentu yang mengunggah file, atau berdasarkan tanggal saat ini.

Dalam contoh berikut, jika pengguna `richard-roe` mengunggah file, itu akan disalin ke folder `file-test2/richard-roe/processed/`. Jika pengguna `mary-major` mengunggah file, itu akan disalin ke folder. `file-test2/mary-major/processed/`

Configure parameters

Configure copy parameters

Step name

dynamic-copy

Destination bucket name

file-test2 ▼

Destination key prefix

If you are copying files into a folder, specify / at the end of the prefix name. Use `${transfer:UserName}` or `${transfer:UploadDate}` to parametrize destination prefix by username or upload date respectively.

`${transfer:UserName}/processed`

☐ Overwrite existing

Demikian pula, Anda dapat menggunakan `${transfer:UploadDate}` sebagai variabel untuk menyalin file ke lokasi tujuan yang dinamai untuk tanggal saat ini. Dalam contoh berikut, jika Anda menetapkan tujuan `${transfer:UploadDate}/processed` pada 1 Februari 2022, file yang diunggah akan disalin ke folder `file-test2/2022-02-01/processed/`.

Configure copy parameters

Step name

dynamic-copy-date

Destination bucket name

file-test2

Destination key prefix

If you are copying files into a folder, specify / at the end of the prefix name. Use `${transfer:UserName}` or `${transfer:UploadDate}` to parametrize destination prefix by username or upload date respectively.

`${transfer:UploadDate}/processed`

☐ Overwrite existing

Anda juga dapat menggunakan kedua variabel ini bersama-sama, menggabungkan fungsionalitasnya. Sebagai contoh:

- Anda dapat mengatur key prefix Destination ke **folder/\${transfer:UserName}/\${transfer:UploadDate}/**, yang akan membuat folder bersarang, misalnya. folder/marymajor/2023-01-05/
- Anda dapat mengatur key prefix Destination ke **folder/\${transfer:UserName}-\${transfer:UploadDate}/**, untuk menggabungkan dua variabel, misalnya. folder/marymajor-2023-01-05/

IAMizin untuk langkah salin

Agar langkah penyalinan berhasil, pastikan peran eksekusi untuk alur kerja Anda berisi izin berikut.

```
{
    "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

s3:ListBucketIzin hanya diperlukan jika Anda tidak memilih Timpa yang sudah ada. Izin ini memeriksa bucket Anda untuk melihat apakah file dengan nama yang sama sudah ada. Jika Anda telah memilih Timpa yang sudah ada, alur kerja tidak perlu memeriksa file, dan hanya bisa menuliskannya.

Jika file Amazon S3 memiliki tag, Anda perlu menambahkan satu atau dua izin ke kebijakan Anda. IAM

- Tambahkan s3:GetObjectTagging file Amazon S3 yang tidak berversi.
- Tambahkan s3:GetObjectVersionTagging untuk file Amazon S3 yang berversi.

Dekripsi file

Blog AWS penyimpanan memiliki posting yang menjelaskan cara mengenkripsi dan mendekripsi file, mengenkripsi dan mendekripsi file dengan [dan](#). PGP AWS Transfer Family

Gunakan PGP dekripsi dalam alur kerja Anda

Transfer Family memiliki dukungan bawaan untuk dekripsi Pretty Good Privacy (PGP). Anda dapat menggunakan PGP dekripsi pada file yang diunggah melalui SFTPFTPS, atau ke Amazon Simple

Storage Service (FTPAmazon S3) Simple Storage Service (Amazon) atau Amazon Elastic File System (Amazon). EFS

Untuk menggunakan PGP dekripsi, Anda harus membuat dan menyimpan kunci PGP pribadi yang akan digunakan untuk dekripsi file Anda. Pengguna Anda kemudian dapat mengenkripsi file dengan menggunakan kunci PGP enkripsi yang sesuai sebelum mengunggah file ke server Transfer Family Anda. Setelah Anda menerima file terenkripsi, Anda dapat mendekripsi file-file tersebut dalam alur kerja Anda. Untuk tutorial detail, lihat [Menyiapkan alur kerja terkelola untuk mendekripsi file](#).

Untuk menggunakan PGP dekripsi dalam alur kerja Anda

1. Identifikasi server Transfer Family untuk meng-host alur kerja Anda, atau buat yang baru. Anda harus memiliki ID server sebelum Anda dapat menyimpan PGP kunci Anda AWS Secrets Manager dengan nama rahasia yang benar.
2. Simpan PGP kunci Anda di AWS Secrets Manager bawah nama rahasia yang diperlukan. Untuk detailnya, lihat [Kelola PGP kunci](#). Alur kerja dapat secara otomatis menemukan PGP kunci yang benar untuk digunakan untuk dekripsi berdasarkan nama rahasia di Secrets Manager.

 Note

Ketika Anda menyimpan rahasia di Secrets Manager, Anda Akun AWS dikenakan biaya. Untuk informasi tentang harga, lihat [AWS Secrets Manager Harga](#).

3. Enkripsi file dengan menggunakan PGP key pair Anda. (Untuk daftar klien yang didukung, lihat [PGPKlien yang didukung](#).) Jika Anda menggunakan baris perintah, jalankan perintah berikut. Untuk menggunakan perintah ini, ganti *username@example.com* dengan alamat email yang Anda gunakan untuk membuat PGP key pair. Ganti *testfile.txt* dengan nama file yang ingin Anda enkripsi.

```
gpg -e -r username@example.com testfile.txt
```

4. Unggah file terenkripsi ke server Transfer Family Anda.
5. Konfigurasi langkah dekripsi dalam alur kerja Anda. Untuk informasi selengkapnya, lihat [Tambahkan langkah dekripsi](#).

Tambahkan langkah dekripsi

Langkah dekripsi mendekripsi file terenkripsi yang diunggah ke Amazon S3 atau Amazon sebagai bagian dari alur kerja Anda. EFS Untuk detail tentang mengonfigurasi dekripsi, lihat. [Gunakan PGP dekripsi dalam alur kerja Anda](#)

Saat Anda membuat langkah dekripsi untuk alur kerja, Anda harus menentukan tujuan untuk file yang didekripsi. Anda juga harus memilih apakah akan menimpa file yang ada jika file sudah ada di lokasi tujuan. Anda dapat memantau hasil alur kerja dekripsi dan mendapatkan log audit untuk setiap file secara real time menggunakan Amazon Logs. CloudWatch

Setelah Anda memilih jenis file Dekripsi untuk langkah Anda, halaman Konfigurasi parameter akan muncul. Isi nilai untuk bagian Konfigurasi parameter PGP dekripsi.

Opsi yang tersedia adalah sebagai berikut:

- Nama langkah - Masukkan nama deskriptif untuk langkah tersebut.
- Lokasi file — Dengan menentukan lokasi file, Anda dapat mendekripsi file yang digunakan pada langkah sebelumnya atau file asli yang diunggah.

Note

Parameter ini tidak tersedia jika langkah ini adalah langkah pertama dari alur kerja.

- Tujuan untuk file yang didekripsi — Pilih bucket Amazon S3 atau sistem file EFS Amazon sebagai tujuan untuk file yang didekripsi.
 - Jika memilih Amazon S3, Anda harus memberikan nama bucket tujuan dan awalan key tujuan. Untuk memparameterisasi awalan key tujuan dengan nama pengguna, **`${transfer:UserName}`** masukkan untuk Destination key prefix. Demikian pula, untuk parameterisasi awalan key tujuan dengan tanggal upload, **`${Transfer:UploadDate}`** masukkan untuk Destination key prefix.
 - Jika Anda memilih AmazonEFS, Anda harus menyediakan sistem dan jalur file tujuan.

Note

Opsi penyimpanan yang Anda pilih di sini harus sesuai dengan sistem penyimpanan yang digunakan oleh server Transfer Family yang terkait dengan alur kerja ini. Jika tidak, Anda akan menerima kesalahan saat mencoba menjalankan alur kerja ini.

- Timpa yang ada — Jika Anda mengunggah file, dan file dengan nama file yang sama sudah ada di tujuan, perilaku tergantung pada pengaturan untuk parameter ini:
 - Jika Timpa yang ada dipilih, file yang ada diganti dengan file yang sedang diproses.
 - Jika Timpa yang ada tidak dipilih, tidak ada yang terjadi, dan pemrosesan alur kerja berhenti.

 Tip

Jika penulisan bersamaan dijalankan pada jalur file yang sama, hal itu dapat mengakibatkan perilaku yang tidak terduga saat menimpa file.

Tangkapan layar berikut menunjukkan contoh opsi yang mungkin Anda pilih untuk langkah dekripsi file Anda.

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.

IAMizin untuk langkah dekripsi

Agar langkah dekripsi berhasil, pastikan peran eksekusi untuk alur kerja Anda berisi izin berikut.

Dekripsi file

337

```
{
    "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

s3:ListBucketIzin hanya diperlukan jika Anda tidak memilih Timpa yang sudah ada. Izin ini memeriksa bucket Anda untuk melihat apakah file dengan nama yang sama sudah ada. Jika Anda telah memilih Timpa yang sudah ada, alur kerja tidak perlu memeriksa file, dan hanya bisa menuliskannya.

Jika file Amazon S3 memiliki tag, Anda perlu menambahkan satu atau dua izin ke kebijakan Anda. IAM

- Tambahkan s3:GetObjectTagging file Amazon S3 yang tidak berversi.

- Tambahkan `s3:GetObjectVersionTagging` untuk file Amazon S3 yang berversi.

Berkas tag

Untuk menandai file yang masuk untuk pemrosesan hilir lebih lanjut, gunakan langkah tag. Masukkan nilai tag yang ingin Anda tetapkan ke file yang masuk. Saat ini, operasi tag hanya didukung jika Anda menggunakan Amazon S3 untuk penyimpanan server Transfer Family Anda.

Berikut contoh langkah tag menetapkan `scan_outcome` dan `clean` sebagai kunci tag dan nilai, masing-masing.

Configure tag parameters

Step name
tag scan

File location
Select the file location to use as an input for this step

☒ Tag the file created from previous step
Input file is selected from the previous step's output

☐ Tag the original source file
Originally uploaded file

Tags

Key	Value
scan_outcome	clean

Remove tag

Add tag

Agar langkah tag berhasil, pastikan peran eksekusi untuk alur kerja Anda berisi izin berikut.

```
{
  "Sid": "Tag",
  "Effect": "Allow",
  "Action": [
    "s3:PutObjectTagging",
    "s3:PutObjectVersionTagging"
  ]
}
```

```

    ],
    "Resource": [
        "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*"
    ]
}

```

Note

Jika alur kerja berisi langkah tag yang berjalan sebelum langkah salin atau dekripsi, Anda perlu menambahkan satu atau dua izin ke kebijakan Anda. IAM

- Tambahkan `s3:GetObjectTagging` file Amazon S3 yang tidak berversi.
- Tambahkan `s3:GetObjectVersionTagging` untuk file Amazon S3 yang berversi.

Hapus berkas

Untuk menghapus file yang diproses dari langkah alur kerja sebelumnya atau untuk menghapus file yang diunggah semula, gunakan langkah menghapus file.

The screenshot shows the 'Configure delete parameters' dialog. It has a 'Step name' field with the value 'delete original file'. Below it, under 'File location', there are two radio button options. The first option is 'Delete the file created from previous step' with the subtext 'Input file is selected from the previous step's output'. The second option is selected and is 'Delete the original source file' with the subtext 'Originally uploaded file'.

Agar langkah penghapusan berhasil, pastikan peran eksekusi untuk alur kerja Anda berisi izin berikut.

```

{
    "Sid": "Delete",
    "Effect": "Allow",
    "Action": [
        "s3:DeleteObjectVersion",
    ]
}

```

```
        "s3:DeleteObject"
      ],
      "Resource": "arn:aws:secretsmanager:region:account-ID:secret:aws/transfer/
* "
    }
  }
```

Variabel bernama untuk alur kerja

Untuk langkah menyalin dan mendekripsi, Anda dapat menggunakan variabel untuk melakukan tindakan secara dinamis. Saat ini, AWS Transfer Family mendukung variabel bernama berikut.

- Gunakan `${transfer:UserName}` untuk menyalin atau mendekripsi file ke tujuan berdasarkan pengguna yang mengunggah file.
- Gunakan `${transfer:UploadDate}` untuk menyalin atau mendekripsi file ke lokasi tujuan berdasarkan tanggal saat ini.

Contoh tag dan hapus alur kerja

Contoh berikut menggambarkan alur kerja yang menandai file masuk yang perlu diproses oleh aplikasi hilir, seperti platform analisis data. Setelah menandai file yang masuk, alur kerja kemudian menghapus file yang awalnya diunggah untuk menghemat biaya penyimpanan.

Console

Contoh tag dan alur kerja pindahkan

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Di panel navigasi kiri, pilih Alur kerja.
3. Pada halaman Alur kerja, pilih Buat alur kerja.
4. Pada halaman Buat alur kerja, masukkan deskripsi. Deskripsi ini muncul di halaman Alur Kerja.
5. Tambahkan langkah pertama (salin).
 - a. Di bagian Langkah nominal, pilih Tambah langkah.
 - b. Pilih Salin file, lalu pilih Berikutnya.
 - c. Masukkan nama langkah, lalu pilih bucket tujuan dan key prefix.

The screenshot displays the 'Configure parameters' step in the AWS Transfer Family console. On the left, a sidebar shows the progress: Step 1 'Choose step type' is completed, Step 2 'Configure parameters' is the current step, and Step 3 'Review and create' is next. The main area is titled 'Configure parameters' and contains a section 'Configure copy parameters'. This section includes three input fields: 'Step name' with the value 'copy-step-first-step', 'Destination bucket name' with a dropdown menu showing 'example-bucket', and 'Destination key prefix' with the value 'test/'. Below these fields is a checkbox labeled 'Overwrite existing' which is currently unchecked. A help text for the 'Destination key prefix' field explains that it should specify a folder path with a trailing slash and can be parametrized using variables like \${transfer:UserName} or \${transfer:UploadDate}.

- d. Pilih Berikutnya, lalu tinjau detail untuk langkahnya.
 - e. Pilih Buat langkah untuk menambahkan langkah dan melanjutkan.
6. Tambahkan langkah kedua (tag).
- a. Di bagian Langkah nominal, pilih Tambah langkah.
 - b. Pilih File Tag, lalu pilih Berikutnya.
 - c. Masukkan nama langkah.
 - d. Untuk lokasi File, pilih Tag file yang dibuat dari langkah sebelumnya.
 - e. Masukkan Kunci dan Nilai.

Configure tag parameters

Step name
tag scan

File location
Select the file location to use as an input for this step

☒ Tag the file created from previous step
Input file is selected from the previous step's output

☐ Tag the original source file
Originally uploaded file

Tags

Key	Value
scan_outcome	clean

Remove tag

Add tag

- f. Pilih Berikutnya, lalu tinjau detail untuk langkahnya.
 - g. Pilih Buat langkah untuk menambahkan langkah dan melanjutkan.
7. Tambahkan langkah ketiga (hapus).
- a. Di bagian Langkah nominal, pilih Tambah langkah.
 - b. Pilih Hapus file, lalu pilih Berikutnya.

Configure delete parameters

Step name
delete original file

File location
Select the file location to use as an input for this step

☐ Delete the file created from previous step
Input file is selected from the previous step's output

☒ Delete the original source file
Originally uploaded file

- c. Masukkan nama langkah.

- d. Untuk lokasi File, pilih Hapus file sumber asli.
 - e. Pilih Berikutnya, lalu tinjau detail untuk langkahnya.
 - f. Pilih Buat langkah untuk menambahkan langkah dan melanjutkan.
8. Tinjau konfigurasi alur kerja, lalu pilih Buat alur kerja.

CLI

Contoh tag dan alur kerja pindahkan

1. Simpan kode berikut ke dalam file; misalnya, `tagAndMoveWorkflow.json`. Ganti masing-masing *user input placeholder* dengan informasi Anda sendiri.

```
[
  {
    "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}"
      }
    }
  ]

```

Langkah pertama menyalin file yang diunggah ke lokasi Amazon S3 baru. Langkah kedua menambahkan tag (pasangan nilai kunci) ke file (`previous.file`) yang disalin ke lokasi baru. Dan, akhirnya, langkah ketiga menghapus file asli (`original.file`).

2. Buat alur kerja dari file yang disimpan. Ganti masing-masing *user input placeholder* dengan informasi Anda sendiri.

```

aws transfer create-workflow --description "short-description" --steps
file://path-to-file --region region-ID

```

Sebagai contoh:

```

aws transfer create-workflow --description "copy-tag-delete workflow" --steps
file://tagAndMoveWorkflow.json --region us-east-1

```

Note

Untuk detail selengkapnya tentang menggunakan file untuk memuat parameter, lihat [Cara memuat parameter dari file](#).

3. Perbarui server yang ada.

Note

Langkah ini mengasumsikan Anda sudah memiliki server Transfer Family dan Anda ingin mengaitkan alur kerja dengannya. Jika belum, lihat [Mengkonfigurasi titik SFTP akhir, FTPS, atau FTP server](#). Ganti masing-masing *user input placeholder* dengan informasi Anda sendiri.

```

aws transfer update-server --server-id server-ID --region region-ID

```

```
--workflow-details '{"OnUpload":[{"WorkflowId": "workflow-ID", "ExecutionRole": "execution-role-ARN"}]}'
```

Sebagai contoh:

```
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"}]}'
```

Gunakan langkah-langkah pemrosesan file khusus

Dengan menggunakan langkah pemrosesan file kustom, Anda dapat menggunakan logika pemrosesan file Anda Sendiri. AWS Lambda Setelah kedatangan file, server Transfer Family memanggil fungsi Lambda yang berisi logika pemrosesan file khusus, seperti mengenkripsi file, memindai malware, atau memeriksa jenis file yang salah. Dalam contoh berikut, AWS Lambda fungsi target digunakan untuk memproses file output dari langkah sebelumnya.

Configure custom parameters

Step name
custom file processing step

File location
Select the file location to use as an input for this step

☒ Apply custom processing to the file created from previous step
Input file is selected from the previous step's output

☐ Apply custom processing to the original source file
Originally uploaded file

Target
am:aws:lambda:us-east-2:1234567... ▼

Timeout (seconds)
60

Note

Untuk contoh fungsi Lambda, lihat. [Contoh fungsi Lambda untuk langkah alur kerja khusus](#). Misalnya peristiwa (termasuk lokasi untuk file yang diteruskan ke Lambda), lihat. [Contoh peristiwa dikirim ke AWS Lambda saat file upload](#)

Dengan langkah alur kerja khusus, Anda harus mengonfigurasi fungsi Lambda untuk memanggil [SendWorkflowStepState](#) API operasi. `SendWorkflowStepState` memberitahukan eksekusi alur kerja bahwa langkah telah selesai dengan status sukses atau kegagalan. Status `SendWorkflowStepState` API operasi memanggil langkah penanganan pengecualian atau langkah nominal dalam urutan linier, berdasarkan hasil fungsi Lambda.

Jika fungsi Lambda gagal atau habis waktu, langkahnya gagal, dan Anda lihat `StepErrored` di log Anda CloudWatch. Jika fungsi Lambda adalah bagian dari langkah nominal dan fungsi merespons `SendWorkflowStepState` dengan `Status="FAILURE"` atau waktu habis, aliran berlanjut dengan langkah-langkah penanganan pengecualian. Dalam hal ini, alur kerja tidak terus mengeksekusi langkah-langkah nominal yang tersisa (jika ada). Untuk detail selengkapnya, lihat [Penanganan pengecualian untuk alur kerja](#).

Ketika Anda memanggil `SendWorkflowStepState` API operasi, Anda harus mengirim parameter berikut:

```
{
  "ExecutionId": "string",
  "Status": "string",
  "Token": "string",
  "WorkflowId": "string"
}
```

Anda dapat mengekstrak `ExecutionId`, `Token`, dan `WorkflowId` dari peristiwa masukan yang diteruskan ketika fungsi Lambda dijalankan (contoh ditampilkan di bagian berikut). `Status` nilainya bisa berupa `SUCCESS` atau `FAILURE`.

Untuk dapat memanggil `SendWorkflowStepState` API operasi dari fungsi Lambda Anda, Anda harus menggunakan versi yang diterbitkan setelah [Alur Kerja Terkelola diperkenalkan](#). AWS SDK

Menggunakan beberapa fungsi Lambda secara berurutan

Bila Anda menggunakan beberapa langkah kustom satu demi satu, opsi lokasi File bekerja secara berbeda daripada jika Anda hanya menggunakan satu langkah kustom. Transfer Family tidak mendukung meneruskan file yang diproses Lambda kembali untuk digunakan sebagai input langkah berikutnya. Jadi, jika Anda memiliki beberapa langkah khusus yang semuanya dikonfigurasi untuk menggunakan `previous.file` opsi, semuanya menggunakan lokasi file yang sama (lokasi file input untuk langkah kustom pertama).

Note

`previous.file` Pengaturan juga bekerja secara berbeda jika Anda memiliki langkah yang telah ditentukan (tag, salin, dekripsi, atau hapus) setelah langkah khusus. Jika langkah yang telah ditentukan dikonfigurasi untuk menggunakan `previous.file` pengaturan, langkah yang telah ditentukan menggunakan file input yang sama yang digunakan oleh langkah kustom. File yang diproses dari langkah kustom tidak diteruskan ke langkah yang telah ditentukan.

Mengakses file setelah pemrosesan kustom

Jika Anda menggunakan Amazon S3 sebagai penyimpanan, dan jika alur kerja menyertakan langkah khusus yang melakukan tindakan pada file yang diunggah semula, langkah selanjutnya tidak dapat mengakses file yang diproses tersebut. Artinya, langkah apa pun setelah langkah khusus tidak dapat mereferensikan file yang diperbarui dari output langkah khusus.

Misalnya, Anda memiliki tiga langkah berikut dalam alur kerja Anda.

- Langkah 1 - Unggah file bernama `example-file.txt`.
- Langkah 2 — Memanggil fungsi Lambda yang `example-file.txt` berubah dalam beberapa cara.
- Langkah 3 — Mencoba untuk melakukan pemrosesan lebih lanjut pada versi terbaru dari `example-file.txt`.

Jika Anda mengonfigurasi `sourceFileLocation` untuk Langkah 3 `${original.file}`, Langkah 3 menggunakan lokasi file asli dari saat server mengunggah file ke penyimpanan di Langkah 1. Jika

Anda menggunakan `${previous.file}` untuk Langkah 3, Langkah 3 menggunakan kembali lokasi file yang Langkah 2 digunakan sebagai input.

Oleh karena itu, Langkah 3 menyebabkan kesalahan. Misalnya, jika langkah 3 mencoba menyalin yang diperbarui `example-file.txt`, Anda menerima kesalahan berikut:

```
{
  "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"
  },
}
```

Kesalahan ini terjadi karena langkah kustom memodifikasi tag entitas (ETag) `example-file.txt` agar tidak cocok dengan file asli.

Note

Perilaku ini tidak terjadi jika Anda menggunakan Amazon EFS karena Amazon EFS tidak menggunakan tag entitas untuk mengidentifikasi file.

Contoh peristiwa dikirim ke AWS Lambda saat file upload

Contoh berikut menunjukkan peristiwa yang dikirim ke AWS Lambda ketika file upload selesai. Salah satu contoh menggunakan server Transfer Family tempat domain dikonfigurasi dengan Amazon S3. Contoh lainnya menggunakan server Transfer Family di mana domain menggunakan AmazonEFS.

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": "MTg0N2Y3N2UtNWl5Ny00ZmZlLTk5YTgtZTU3YzViYjllNmZm",
    "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"
    }
}

```

Contoh fungsi Lambda untuk langkah alur kerja khusus

Fungsi Lambda berikut mengekstrak informasi mengenai status eksekusi, dan kemudian memanggil [SendWorkflowStepState](#) API operasi untuk mengembalikan status ke alur kerja untuk langkah tersebut —baik atau. SUCCESS FAILURE Sebelum fungsi Anda memanggil SendWorkflowStepState API

operasi, Anda dapat mengonfigurasi Lambda untuk mengambil tindakan berdasarkan logika alur kerja Anda.

```
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)
    }
```

IAMizin untuk langkah khusus

Agar langkah yang memanggil Lambda berhasil, pastikan peran eksekusi untuk alur kerja Anda berisi izin berikut.

```
{
    "Sid": "Custom",
    "Effect": "Allow",
    "Action": [
        "lambda:InvokeFunction"
    ],
    "Resource": [
        "arn:aws:lambda:region:account-id:function:function-name"
    ]
}
```

IAMkebijakan untuk alur kerja

Ketika Anda menambahkan alur kerja ke server, Anda harus memilih peran eksekusi. Server menggunakan peran ini ketika menjalankan alur kerja. Jika peran tidak memiliki izin yang tepat, AWS Transfer Family tidak dapat menjalankan alur kerja.

Bagian ini menjelaskan satu kemungkinan set AWS Identity and Access Management (IAM) izin yang dapat Anda gunakan untuk menjalankan alur kerja. Contoh lain dijelaskan nanti dalam topik ini.

Note

Jika file Amazon S3 memiliki tag, Anda perlu menambahkan satu atau dua izin ke kebijakan Anda. IAM

- Tambahkan `s3:GetObjectTagging` file Amazon S3 yang tidak berversi.
- Tambahkan `s3:GetObjectVersionTagging` untuk file Amazon S3 yang berversi.

Untuk membuat peran eksekusi untuk alur kerja Anda

1. Buat IAM peran baru, dan tambahkan kebijakan AWS terkelola `AWSTransferFullAccess` ke peran. Untuk informasi selengkapnya tentang membuat IAM peran baru, lihat [the section called "Buat IAM peran dan kebijakan"](#).
2. Buat kebijakan lain dengan izin berikut, dan lampirkan ke peran Anda. Ganti masing-masing *user input placeholder* dengan informasi Anda sendiri.

```
{
  "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. Simpan peran ini dan tentukan sebagai peran eksekusi saat Anda menambahkan alur kerja ke server.

Note

Ketika Anda membangun IAM peran, AWS merekomendasikan agar Anda membatasi akses ke sumber daya Anda sebanyak mungkin untuk alur kerja Anda.

Hubungan kepercayaan alur kerja

Peran eksekusi alur kerja juga membutuhkan hubungan kepercayaan dengan `transfer.amazonaws.com`. Untuk membangun hubungan kepercayaan AWS Transfer Family, lihat [Untuk membangun hubungan kepercayaan](#).

Saat Anda membangun hubungan kepercayaan Anda, Anda juga dapat mengambil langkah-langkah untuk menghindari masalah wakil yang membingungkan. Untuk deskripsi masalah ini, serta contoh cara menghindarinya, lihat [the section called "Pencegahan confused deputy lintas layanan"](#).

Contoh peran eksekusi: Dekripsi, salin, dan tag

Jika Anda memiliki alur kerja yang menyertakan langkah penandaan, penyalinan, dan dekripsi, Anda dapat menggunakan kebijakan berikut. IAM Ganti masing-masing *user input placeholder* dengan informasi Anda sendiri.

```
{
  "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/
*"
  }
]
}

```

Contoh peran eksekusi: Jalankan fungsi dan hapus

Dalam contoh ini, Anda memiliki alur kerja yang memanggil fungsi. AWS Lambda Jika alur kerja menghapus file yang diunggah dan memiliki langkah penanganan pengecualian untuk menindaklanjuti eksekusi alur kerja yang gagal pada langkah sebelumnya, gunakan kebijakan berikut. IAM Ganti masing-masing *user input placeholder* dengan informasi Anda sendiri.

```

{
  "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"
    ]
}
]
```

Penanganan pengecualian untuk alur kerja

Jika terjadi kesalahan selama eksekusi alur kerja, langkah-langkah penanganan pengecualian yang Anda tentukan akan dijalankan. Anda menentukan langkah penanganan kesalahan untuk alur kerja dengan cara yang sama seperti Anda menentukan langkah nominal untuk alur kerja. Misalnya, misalkan Anda telah mengonfigurasi pemrosesan kustom dalam langkah-langkah nominal untuk memvalidasi file yang masuk. Jika validasi file gagal, langkah penanganan pengecualian dapat mengirim email ke administrator.

Contoh alur kerja berikut berisi dua langkah:

- Satu langkah nominal yang memeriksa apakah file yang diunggah dalam format CSV
- Langkah penanganan pengecualian yang mengirim email jika file yang diunggah tidak dalam CSV format, dan langkah nominal gagal

Untuk memulai langkah penanganan pengecualian, AWS Lambda fungsi dalam langkah nominal harus merespons dengan `Status="FAILURE"`. Untuk informasi selengkapnya tentang penanganan kesalahan dalam alur kerja, lihat [the section called “Gunakan langkah-langkah pemrosesan file khusus”](#).

w-1234567890abcdef0 Delete			
Description			
Workflow description			
Check for CSV files			
Nominal steps (1) Info			
Number	Description	Type	Configuration
1	is-CSV	CUSTOM	Details
Exception handlers (1) Info			
Number	Description	Type	Configuration
1	send-email	CUSTOM	Details

Pantau eksekusi alur kerja

Amazon CloudWatch memonitor AWS sumber daya Anda dan aplikasi yang Anda jalankan AWS Cloud secara real time. Anda dapat menggunakan Amazon CloudWatch untuk mengumpulkan dan melacak metrik, yang merupakan variabel yang dapat Anda ukur untuk alur kerja Anda. Anda dapat melihat metrik alur kerja dan log gabungan menggunakan Amazon. CloudWatch

CloudWatch logging untuk alur kerja

CloudWatch menyediakan audit dan pencatatan terkonsolidasi untuk kemajuan dan hasil alur kerja.

Lihat CloudWatch log Amazon untuk alur kerja

1. Buka CloudWatch konsol Amazon di <https://console.aws.amazon.com/cloudwatch/>.
2. Di panel navigasi kiri, pilih Log, lalu pilih Grup log.
3. Pada halaman Grup log, pada bilah navigasi, pilih Wilayah yang benar untuk AWS Transfer Family server Anda.
4. Pilih grup log yang sesuai dengan server Anda.

Misalnya, jika ID server Anda s-1234567890abcdef0, grup log Anda adalah /aws/transfer/s-1234567890abcdef0.

5. Pada halaman detail grup log untuk server Anda, aliran log terbaru ditampilkan. Ada dua aliran log untuk pengguna yang Anda jelajahi:
- Satu untuk setiap sesi Secure Shell (SSH) File Transfer Protocol (SFTP).
 - Satu untuk alur kerja yang sedang dijalankan untuk server Anda. Format untuk aliran log untuk alur kerja adalah `username.workflowID.uniqueStreamSuffix`.

Misalnya, jika pengguna Andamary-major, Anda memiliki aliran log berikut:

```
mary-major-east.1234567890abcdef0
mary.w-abcdef01234567890.021345abcdef6789
```

 Note

Pengidentifikasi alfanumerik 16 digit yang tercantum dalam contoh ini adalah fiktif. Nilai yang Anda lihat di Amazon CloudWatch berbeda.

Halaman peristiwa Log untuk `mary-major-usa-east.1234567890abcdef0` menampilkan detail untuk setiap sesi pengguna, dan aliran `mary.w-abcdef01234567890.021345abcdef6789` log berisi detail untuk alur kerja.

Berikut ini adalah contoh aliran log untuk `mary.w-abcdef01234567890.021345abcdef6789`, berdasarkan alur kerja (`w-abcdef01234567890`) yang berisi langkah salin.

```
{
  "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 metrik untuk alur kerja

AWS Transfer Family menyediakan beberapa metrik untuk alur kerja. Anda dapat melihat metrik berapa banyak eksekusi alur kerja yang dimulai, diselesaikan dengan sukses, dan gagal pada menit sebelumnya. Semua CloudWatch metrik untuk Transfer Family dijelaskan dalam [Menggunakan CloudWatch metrik untuk Transfer Family](#).

Buat alur kerja dari template

Anda dapat menerapkan AWS CloudFormation tumpukan yang membuat alur kerja dan server dari template. Prosedur ini berisi contoh yang dapat Anda gunakan untuk menyebarkan alur kerja dengan cepat.

Untuk membuat AWS CloudFormation tumpukan yang membuat AWS Transfer Family alur kerja dan server

1. Buka AWS CloudFormation konsol di <https://console.aws.amazon.com/cloudformation>.
2. Simpan kode berikut ke file.

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. Ganti item berikut dengan nilai aktual Anda.

- Ganti *workflow-execution-role-arn* dengan ARN untuk peran eksekusi alur kerja yang sebenarnya. Sebagai contoh, `arn:aws:transfer:us-east-2:111122223333:workflow/w-1234567890abcdef0`.
- Ganti `arn:aws:lambda:region:account-id:function:function-name` dengan fungsi ARN for Lambda Anda. Misalnya, `arn:aws:lambda:us-east-2:123456789012:function:example-lambda-idp`.

4. Ikuti petunjuk untuk menerapkan AWS CloudFormation tumpukan dari template yang ada di [Memilih template tumpukan](#) di Panduan AWS CloudFormation Pengguna.

Setelah tumpukan dikerahkan, Anda dapat melihat detailnya di tab Output di CloudFormation konsol. Template membuat AWS Transfer Family SFTP server baru yang dikelola pengguna layanan pengguna, dan alur kerja baru, dan mengaitkan alur kerja dengan server baru.

Menghapus alur kerja dari server Transfer Family

Jika Anda telah mengaitkan alur kerja dengan server Transfer Family, dan sekarang Anda ingin menghapus asosiasi tersebut, Anda dapat melakukannya dengan menggunakan konsol atau secara terprogram.

Console

Untuk menghapus alur kerja dari server Transfer Family

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Di panel navigasi kiri, pilih Server.
3. Pilih pengenalan untuk server di kolom ID Server.
4. Pada halaman detail untuk server, gulir ke bawah ke bagian Detail tambahan, lalu pilih Edit.
5. Pada halaman Edit detail tambahan, di bagian Alur kerja terkelola, kosongkan informasi untuk semua pengaturan:
 - Pilih tanda hubung (-) dari daftar alur kerja untuk Alur Kerja untuk upload file lengkap.
 - Jika belum dihapus, pilih tanda hubung (-) dari daftar alur kerja untuk Alur Kerja untuk unggahan file sebagian.
 - Pilih tanda hubung (-) dari daftar peran untuk peran eksekusi alur kerja terkelola.

Jika Anda tidak melihat tanda hubung, gulir ke atas hingga Anda melihatnya, karena ini adalah nilai pertama di setiap menu.

Layar akan terlihat seperti berikut ini.

Managed workflows [Info](#)

Workflow for complete file uploads
Select the workflow that AWS Transfer Family should run on all files that are uploaded in full via this server

Select a workflow ▼ [Create a new Workflow](#)

Workflow for partial file uploads
Select the workflow that AWS Transfer Family should run on all files that are only partially uploaded via this server

Select a workflow ▼ [Create a new Workflow](#)

Managed workflows execution role [Info](#)
Select the role that AWS Transfer Family should assume when executing a workflow

- ▼

6. Gulir ke bawah dan pilih Simpan untuk menyimpan perubahan Anda.

CLI

Anda menggunakan panggilan `update-server` (atau `UpdateServer` untuk API), dan memberikan argumen kosong untuk `OnUpload` dan `OnPartialUpload` parameter.

Dari AWS CLI, jalankan perintah berikut:

```
aws transfer update-server --server-id your-server-id --workflow-details
'{"OnPartialUpload":[],"OnUpload":[]}'
```

Ganti *your-server-id* dengan ID untuk server Anda. Misalnya, jika ID server Anda `s-01234567890abcdef`, perintahnya adalah sebagai berikut:

```
aws transfer update-server --server-id s-01234567890abcdef --workflow-details
'{"OnPartialUpload":[],"OnUpload":[]}'
```

Pembatasan dan batasan alur kerja yang dikelola

Pembatasan

Pembatasan berikut saat ini berlaku untuk alur kerja pemrosesan pasca-unggah untuk AWS Transfer Family

- AWS Lambda Fungsi lintas akun dan lintas wilayah tidak didukung. Namun, Anda dapat menyalin seluruh akun, asalkan kebijakan AWS Identity and Access Management (IAM) Anda dikonfigurasi dengan benar.
- Untuk semua langkah alur kerja, setiap bucket Amazon S3 yang diakses oleh alur kerja harus berada di wilayah yang sama dengan alur kerja itu sendiri.
- Untuk langkah dekripsi, tujuan dekripsi harus cocok dengan sumber untuk Wilayah dan penyimpanan cadangan (misalnya, jika file yang akan didekripsi disimpan di Amazon S3, maka tujuan yang ditentukan juga harus di Amazon S3).
- Hanya langkah kustom asinkron yang didukung.
- Batas waktu langkah khusus adalah perkiraan. Artinya, mungkin perlu waktu sedikit lebih lama dari yang ditentukan. Selain itu, alur kerja tergantung pada fungsi Lambda. Oleh karena itu, jika fungsi tertunda selama eksekusi, alur kerja tidak menyadari penundaan.
- Jika Anda melebihi batas pembatasan, Transfer Family tidak menambahkan operasi alur kerja ke antrian.
- Alur kerja tidak dimulai untuk file yang memiliki ukuran 0. File dengan ukuran lebih besar dari 0 melakukan memulai alur kerja terkait.

Batasan

Selain itu, batasan fungsional berikut berlaku untuk alur kerja untuk Transfer Family:

- Jumlah alur kerja per Wilayah, per akun, dibatasi hingga 10.
- Batas waktu maksimum untuk langkah-langkah khusus adalah 30 menit.
- Jumlah maksimum langkah dalam alur kerja adalah 8.
- Jumlah maksimum tag per alur kerja adalah 50.
- Jumlah maksimum eksekusi bersamaan yang berisi langkah dekripsi adalah 250 per alur kerja.
- Anda dapat menyimpan maksimal 3 kunci PGP pribadi, per server Transfer Family, per pengguna.
- Ukuran maksimum untuk file yang didekripsi adalah 10 GB.
- Kami membatasi tingkat eksekusi baru menggunakan sistem [token bucket](#) dengan kapasitas burst 100 dan tingkat isi ulang 1.
- Setiap kali Anda menghapus alur kerja dari server dan menggantinya dengan yang baru, atau memperbarui konfigurasi server (yang memengaruhi peran eksekusi alur kerja), Anda harus menunggu sekitar 10 menit sebelum menjalankan alur kerja baru. Server Transfer Family

menyimpan cache detail alur kerja, dan dibutuhkan waktu 10 menit bagi server untuk menyegarkan cache-nya.

Selain itu, Anda harus keluar dari SFTP sesi aktif apa pun, dan kemudian masuk kembali setelah masa tunggu 10 menit untuk melihat perubahannya.

Mengelola server

Di bagian ini, Anda dapat menemukan informasi tentang cara melihat daftar server Anda, cara melihat detail server Anda, cara mengedit detail server Anda, dan cara mengubah kunci host untuk server yang SFTP diaktifkan.

Topik

- [Lihat daftar server](#)
- [Hapus server](#)
- [LihatSFTP,FTPS, dan detail FTP server](#)
- [Lihat detail AS2 server](#)
- [Edit detail server](#)
- [Kelola kunci host untuk server SFTP yang diaktifkan](#)
- [Memantau penggunaan di konsol](#)

Lihat daftar server

Di AWS Transfer Family konsol, Anda dapat menemukan daftar semua server Anda yang terletak di AWS Wilayah yang Anda pilih.

Untuk menemukan daftar server Anda yang ada di suatu AWS Wilayah

- Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.

Jika Anda memiliki satu atau beberapa server di AWS Wilayah saat ini, konsol terbuka untuk menampilkan daftar server Anda. Jika Anda tidak melihat daftar server, pastikan Anda berada di Wilayah yang benar. Anda juga dapat memilih Server dari panel navigasi.

Untuk informasi selengkapnya tentang melihat detail server Anda, lihat[LihatSFTP,FTPS, dan detail FTP server](#).

Hapus server

Prosedur ini menjelaskan cara menghapus server Transfer Family dengan menggunakan AWS Transfer Family konsol atau AWS CLI.

 Important

Anda ditagih, untuk setiap protokol yang diaktifkan untuk mengakses titik akhir Anda, hingga Anda menghapus server.

 Warning

Menghapus server mengakibatkan semua penggunaanya dihapus. Data dalam bucket yang diakses dengan menggunakan server tidak dihapus, dan tetap dapat diakses oleh AWS pengguna yang memiliki hak istimewa untuk bucket Amazon S3 tersebut.

Console

Untuk menghapus server dengan menggunakan konsol

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Di panel navigasi kiri, pilih Server.
3. Pilih kotak centang server yang ingin Anda hapus.
4. Untuk Tindakan, pilih Hapus.
5. Di kotak dialog konfirmasi yang muncul, masukkan kata **delete**, lalu pilih Hapus untuk mengonfirmasi bahwa Anda ingin menghapus server.

Server dihapus dari halaman Server dan Anda tidak lagi ditagih untuk itu.

AWS CLI

Untuk menghapus server dengan menggunakan CLI

1. (Opsional) Jalankan perintah berikut untuk melihat detail server yang ingin Anda hapus secara permanen.

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

`describe-server` Perintah ini mengembalikan semua detail untuk server Anda.

2. Jalankan perintah berikut untuk menghapus server.

```
aws transfer delete-server --server-id your-server-id
```

Jika berhasil, perintah menghapus server dan tidak mengembalikan informasi apa pun.

Lihat SFTP, FTPS, dan detail FTP server

Anda dapat menemukan daftar detail dan properti untuk AWS Transfer Family server individual. Properti server termasuk protokol, penyedia identitas, status, jenis titik akhir, nama host khusus, titik akhir, pengguna, peran logging, kunci host server, dan tag.

Untuk melihat detail server

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Di panel navigasi, pilih Server.
3. Pilih pengenalan di kolom ID Server untuk melihat halaman detail Server, yang ditampilkan berikut.

Anda dapat mengubah properti server di halaman ini dengan memilih Edit. Untuk informasi selengkapnya tentang mengedit detail server, lihat [Edit detail server](#). Halaman detail untuk AS2 server sedikit berbeda. Untuk AS2 server, lihat [Lihat detail AS2 server](#).

Protocols	Identity provider
Protocols over which clients can connect to your server's endpoint SFTP	Identity provider type Info Custom - AWS Lambda AWS Lambda function test-UserAuthenticationLambda ↗

Note

Kunci host server Deskripsi dan Nilai impor Tanggal baru per September 2022. Nilai-nilai ini diperkenalkan untuk mendukung fitur beberapa kunci host. Fitur ini memerlukan migrasi dari setiap kunci host tunggal yang digunakan sebelum pengenalan beberapa kunci host.

Nilai tanggal yang diimpor untuk kunci host server yang dimigrasi diatur ke tanggal modifikasi terakhir untuk server. Artinya, tanggal yang Anda lihat untuk kunci host yang

dimigrasi sesuai dengan tanggal terakhir Anda memodifikasi server dengan cara apa pun, sebelum migrasi kunci host server.

Satu-satunya kunci yang dimigrasikan adalah kunci host server tertua atau satu-satunya.

Setiap kunci tambahan memiliki tanggal sebenarnya sejak Anda mengimpornya.

Selain itu, kunci yang dimigrasi memiliki deskripsi yang membuatnya mudah untuk mengidentifikasinya sebagai telah dimigrasi.

Migrasi terjadi antara 2 September dan 13 September. Tanggal migrasi aktual dalam rentang ini tergantung pada Wilayah server Anda.

Additional details Edit

Log group /aws/transfer/s-	Domain Amazon S3	Login display banner View the display message
Logging role Info AWSTransferLoggingAccess	Workflow for complete uploads w-	SetStat option Ignore
Server host key Info SHA256:	Workflow for partial uploads -	TLS session resumption -
Security Policy Info TransferSecurityPolicy-2020-06	Managed workflows execution role transfer-workflows	Passive IP -

Lihat detail AS2 server

Anda dapat menemukan daftar detail dan properti untuk AWS Transfer Family server individual. Properti server termasuk protokol, status, dan banyak lagi. Untuk AS2 server, Anda juga dapat melihat alamat IP jalan AS2 keluar asinkronMDN.

Protocols Edit

Protocols over which clients can connect to your server's endpoint

- AS2

Identity provider Edit

AS2 Auth
Basic authentication is not supported for AS2. Access can be controlled through VPC security groups.

Setiap AS2 server diberi tiga alamat IP statis. Gunakan alamat IP ini untuk mengirim asinkron MDNs ke mitra dagang Anda. AS2

AS2 asynchronous MDN egress IP details

Below are the service managed static IP addresses used for sending your asynchronous MDNs to trading partners over AS2

-  
-  
-  

Bagian bawah halaman detail AS2 server berisi detail untuk alur kerja terlampir dan informasi pemantauan dan penandaan.

Workflows

Workflow for complete uploads
w-  0

Workflow for partial uploads
-

Managed workflows execution role


[Edit](#)

Monitoring

1h 3h 12h 1d 3d 1w 

UTC timezone 

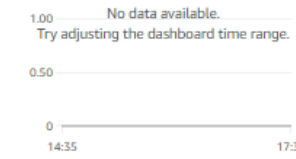
 

BytesIn

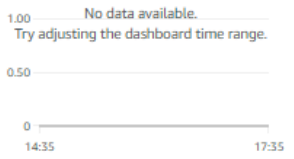
BytesOut

FilesIn

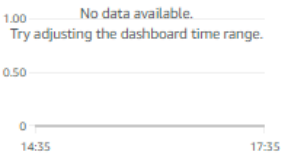
FilesOut



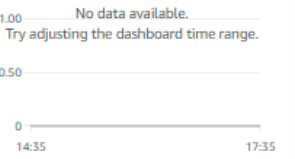
No data available.
Try adjusting the dashboard time range.



No data available.
Try adjusting the dashboard time range.



No data available.
Try adjusting the dashboard time range.



No data available.
Try adjusting the dashboard time range.

AS2 Monitoring

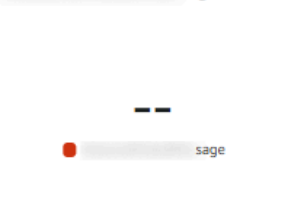
1h 3h 12h 1d 3d 1w 

UTC timezone 

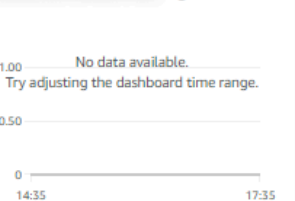
 

InboundMessage

InboundMessage



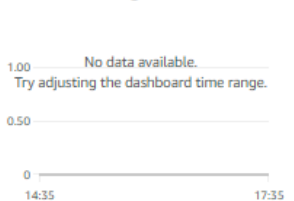
sage



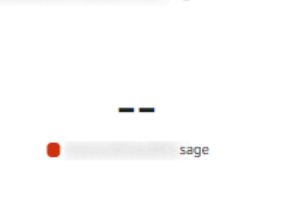
sage



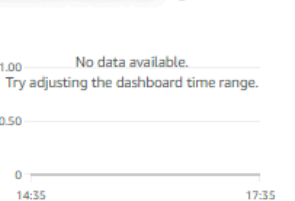
InboundMessage



No data available.
Try adjusting the dashboard time range.



sage



No data available.
Try adjusting the dashboard time range.

Edit detail server

Setelah Anda membuat AWS Transfer Family server, Anda dapat mengedit konfigurasi server.

Topik

- [Edit protokol transfer file](#)
- [Edit parameter penyedia identitas khusus](#)
- [Edit titik akhir server](#)
- [Edit konfigurasi pencatatan Anda](#)
- [Edit kebijakan keamanan](#)
- [Mengubah alur kerja terkelola untuk server Anda](#)
- [Ubah spanduk tampilan untuk server Anda](#)
- [Menempatkan server Anda secara online atau offline](#)

Untuk mengedit konfigurasi server

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Di panel navigasi kiri, pilih Server.
3. Pilih pengenal di kolom ID Server untuk melihat halaman detail Server, yang ditampilkan berikut.

Anda dapat mengubah properti server di halaman ini dengan memilih Edit:

- Untuk mengubah protokol, lihat. [Edit protokol transfer file](#)
- Untuk penyedia identitas, perhatikan bahwa Anda tidak dapat mengubah jenis penyedia identitas server setelah membuat server. Untuk mengubah penyedia identitas, hapus server dan buat yang baru dengan penyedia identitas yang Anda inginkan.

Note

Jika server Anda menggunakan penyedia identitas khusus, Anda dapat mengedit beberapa properti. Untuk detailnya, lihat [Edit parameter penyedia identitas khusus](#).

- Untuk mengubah jenis titik akhir atau nama host kustom, lihat. [Edit titik akhir server](#)
- Untuk menambahkan perjanjian, Anda harus terlebih dahulu menambahkan AS2 sebagai protokol ke server Anda. Untuk detailnya, lihat [Edit protokol transfer file](#).

- Untuk mengelola kunci host untuk server Anda, lihat [Kelola kunci host untuk server SFTP yang diaktifkan](#).
- Di bawah Detail tambahan, Anda dapat mengedit informasi berikut:
 - Untuk mengubah peran logging, lihat [Edit konfigurasi pencatatan Anda](#).
 - Untuk mengubah kebijakan keamanan, lihat [Edit kebijakan keamanan](#).
 - Untuk mengubah kunci host server, lihat [Kelola kunci host untuk server SFTP yang diaktifkan](#).
 - Untuk mengubah alur kerja terkelola untuk server Anda, lihat [Mengubah alur kerja terkelola untuk server Anda](#).
 - Untuk mengedit spanduk tampilan untuk server Anda, lihat [Ubah spanduk tampilan untuk server Anda](#).
- Di bawah Konfigurasi tambahan, Anda dapat mengedit informasi berikut:
 - SetStat opsi: aktifkan opsi ini untuk mengabaikan kesalahan yang dihasilkan saat klien mencoba menggunakan SETSTAT pada file yang Anda unggah ke bucket Amazon S3. Untuk detail tambahan, lihat SetStatOption dokumentasi dalam [ProtocolDetailstopik](#).
 - TLSDimulainya kembali sesi: menyediakan mekanisme untuk melanjutkan atau berbagi kunci rahasia yang dinegosiasikan antara kontrol dan koneksi data untuk suatu sesi. FTPS Untuk detail tambahan, lihat TlsSessionResumptionMode dokumentasi dalam [ProtocolDetailstopik](#).
 - IP pasif: menunjukkan mode pasif, untuk FTP dan FTPS protokol. Masukkan satu IPv4 alamat, seperti alamat IP publik firewall, router, atau penyeimbang beban. Untuk detail tambahan, lihat PassiveIp dokumentasi dalam [ProtocolDetailstopik](#).
- Untuk memulai atau menghentikan server Anda, lihat [Menempatkan server Anda secara online atau offline](#).
- Untuk menghapus server, lihat [Hapus server](#).
- Untuk mengedit properti pengguna, lihat [Mengelola kontrol akses](#).

Protocols Edit	Identity provider Edit
Protocols over which clients can connect to your server's endpoint SFTP	Identity provider type Info Custom - AWS Lambda AWS Lambda function test-UserAuthenticationLambda ↗

Note

Kunci host server Deskripsi dan Nilai impor Tanggal baru per September 2022. Nilai-nilai ini diperkenalkan untuk mendukung fitur beberapa kunci host. Fitur ini memerlukan migrasi dari setiap kunci host tunggal yang digunakan sebelum pengenalan beberapa kunci host.

Nilai tanggal yang diimpor untuk kunci host server yang dimigrasi diatur ke tanggal modifikasi terakhir untuk server. Artinya, tanggal yang Anda lihat untuk kunci host yang dimigrasi sesuai dengan tanggal terakhir Anda memodifikasi server dengan cara apa pun, sebelum migrasi kunci host server.

Satu-satunya kunci yang dimigrasikan adalah kunci host server tertua atau satu-satunya. Setiap kunci tambahan memiliki tanggal sebenarnya sejak Anda mengimpornya. Selain itu, kunci yang dimigrasi memiliki deskripsi yang membuatnya mudah untuk mengidentifikasinya sebagai telah dimigrasi.

Migrasi terjadi antara 2 September dan 13 September. Tanggal migrasi aktual dalam rentang ini tergantung pada Wilayah server Anda.

Additional details

Edit

Log group /aws/transfer/s-	Domain Amazon S3	Login display banner View the display message
Logging role Info AWSTransferLoggingAccess	Workflow for complete uploads w-	SetStat option Ignore
Server host key Info SHA256:	Workflow for partial uploads -	TLS session resumption -
Security Policy Info TransferSecurityPolicy-2020-06	Managed workflows execution role transfer-workflows	Passive IP -

Edit protokol transfer file

Di AWS Transfer Family konsol, Anda dapat mengedit protokol transfer file. Protokol transfer file menghubungkan klien ke titik akhir server Anda.

Untuk mengedit protokol

1. Pada halaman Detail Server, pilih Edit di samping Protokol.
2. Pada halaman Edit protokol, pilih atau kosongkan kotak centang protokol atau kotak centang untuk menambah atau menghapus protokol transfer file berikut:

- Secure Shell (SSH) Protokol Transfer File (SFTP) — transfer file melalui SSH

Untuk informasi lebih lanjut tentang SFTP, lihat [Buat server SFTP yang diaktifkan](#).

- File Transfer Protocol Secure (FTPS) — transfer file dengan TLS enkripsi

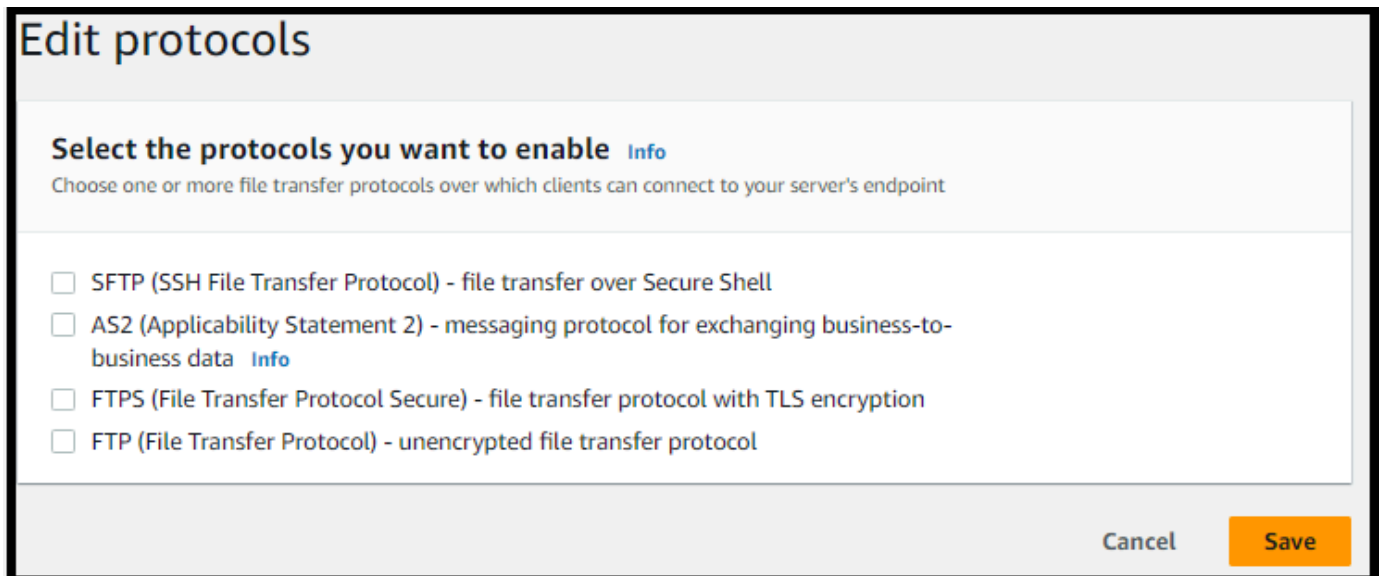
Untuk informasi lebih lanjut tentang FTPS, lihat [Buat server FTPS yang diaktifkan](#).

- Protokol Transfer File (FTP) — transfer file tidak terenkripsi

Untuk informasi lebih lanjut tentang FTPS, lihat [Buat server FTP yang diaktifkan](#).

Note

Jika Anda memiliki server yang ada diaktifkan hanya untuk SFTP, dan Anda ingin menambahkan FTPS dan FTP, Anda harus memastikan bahwa Anda memiliki penyedia identitas yang tepat dan pengaturan tipe titik akhir yang kompatibel dengan FTPS dan FTP.



Edit protocols

Select the protocols you want to enable [Info](#)

Choose one or more file transfer protocols over which clients can connect to your server's endpoint

- ☐ SFTP (SSH File Transfer Protocol) - file transfer over Secure Shell
- ☐ AS2 (Applicability Statement 2) - messaging protocol for exchanging business-to-business data [Info](#)
- ☐ FTPS (File Transfer Protocol Secure) - file transfer protocol with TLS encryption
- ☐ FTP (File Transfer Protocol) - unencrypted file transfer protocol

Cancel Save

Jika Anda memilih FTPS, Anda harus memilih sertifikat yang disimpan di AWS Certificate Manager (ACM) yang akan digunakan untuk mengidentifikasi server Anda ketika klien terhubung dengannya FTPS.

Untuk meminta sertifikat publik baru, lihat [Meminta sertifikat publik](#) di Panduan AWS Certificate Manager Pengguna.

Untuk mengimpor sertifikat yang ada ACM, lihat [Mengimpor sertifikat ke ACM](#) dalam Panduan AWS Certificate Manager Pengguna.

Untuk meminta sertifikat pribadi untuk digunakan FTPS melalui alamat IP pribadi, lihat [Meminta sertifikat pribadi](#) di Panduan AWS Certificate Manager Pengguna.

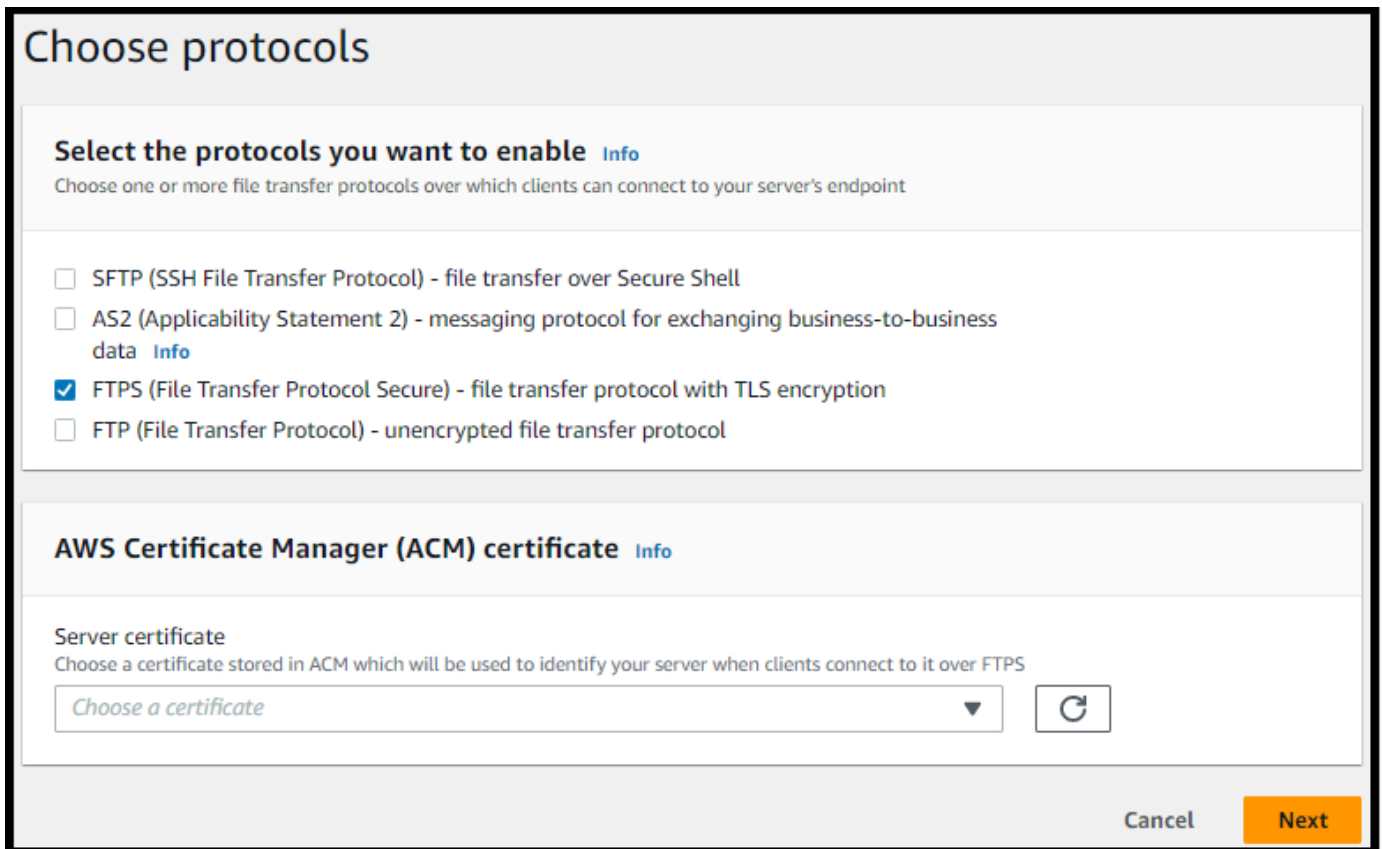
Sertifikat dengan algoritme kriptografi dan ukuran kunci berikut didukung:

- 2048-bit RSA (_2048) RSA

- 4096-bit RSA (_4096) RSA
- Elliptic Prime Curve 256 bit (EC_prime256v1)
- Elliptic Prime Curve 384 bit (EC_secp384r1)
- Elliptic Prime Curve 521 bit (EC_secp521r1)

 Note

Sertifikat harus berupa SSL TLS sertifikat/X.509 versi 3 yang valid dengan FQDN atau alamat IP yang ditentukan dan informasi tentang penerbit.



The screenshot shows a 'Choose protocols' dialog box. It has a title bar 'Choose protocols' and a subtitle 'Select the protocols you want to enable' with an 'Info' link. Below the subtitle is a description: 'Choose one or more file transfer protocols over which clients can connect to your server's endpoint'. There are four checkboxes: 'SFTP (SSH File Transfer Protocol) - file transfer over Secure Shell', 'AS2 (Applicability Statement 2) - messaging protocol for exchanging business-to-business data' (with an 'Info' link), 'FTPS (File Transfer Protocol Secure) - file transfer protocol with TLS encryption' (which is checked), and 'FTP (File Transfer Protocol) - unencrypted file transfer protocol'. Below this is a section for 'AWS Certificate Manager (ACM) certificate' with an 'Info' link. Under this section is a 'Server certificate' label and a description: 'Choose a certificate stored in ACM which will be used to identify your server when clients connect to it over FTPS'. There is a dropdown menu with the text 'Choose a certificate' and a refresh button. At the bottom right are 'Cancel' and 'Next' buttons.

3. Pilih Simpan. Anda dikembalikan ke halaman detail Server.

Edit parameter penyedia identitas khusus

Di AWS Transfer Family konsol, untuk penyedia identitas khusus, Anda dapat mengubah beberapa pengaturan, tergantung pada apakah Anda menggunakan fungsi Lambda atau Gateway. API Dalam

kedua kasus, jika server Anda menggunakan SFTP protokol, Anda dapat mengedit metode otentikasi Anda.

- Jika Anda menggunakan Lambda sebagai penyedia identitas, Anda dapat mengubah fungsi Lambda yang mendasarinya.

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

AWS Lambda function

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

- Jika Anda menggunakan API Gateway sebagai penyedia identitas, Anda dapat memperbarui Gateway URL atau peran pemanggilan, atau keduanya.

Transfer Family > Servers > s-[redacted] > 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

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

Edit titik akhir server

Di AWS Transfer Family konsol, Anda dapat memodifikasi jenis titik akhir server dan nama host khusus.

Untuk mengedit detail titik akhir server

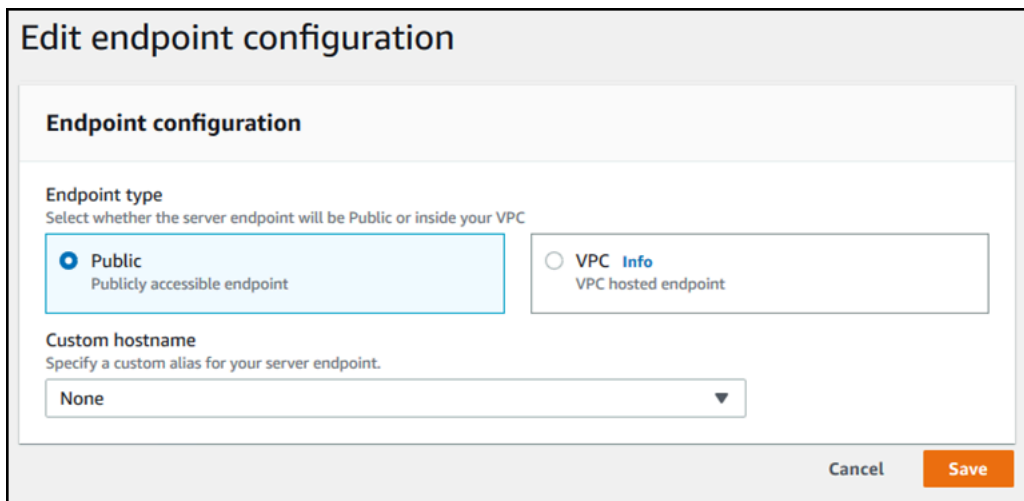
1. Pada halaman Detail Server, pilih Edit di samping Detail titik akhir.
2. Pada halaman konfigurasi Edit titik akhir, untuk tipe Endpoint, pilih salah satu dari berikut ini:
 - Publik — Opsi ini membuat server Anda dapat diakses melalui internet.
 - VPC— Opsi ini membuat server Anda dapat diakses di cloud pribadi virtual Anda (VPC). Untuk informasi tentang VPC, lihat [Buat server di cloud pribadi virtual](#).
3. Untuk nama host Kustom, pilih salah satu dari berikut ini:
 - Tidak ada - Jika Anda tidak ingin menggunakan domain khusus, pilih Tidak Ada.

Anda mendapatkan nama host server yang disediakan oleh AWS Transfer Family. Nama host server mengambil formulir `serverId.server.transfer.regionId.amazonaws.com`.

- DNS Alias Amazon Route 53 - Untuk menggunakan DNS alias yang dibuat secara otomatis untuk Anda di Route 53, pilih opsi ini.
- Lainnya DNS — Untuk menggunakan nama host yang sudah Anda miliki di DNS layanan eksternal pilih Lainnya DNS.

Memilih Amazon Route 53 DNS alias atau Lainnya DNS menentukan metode resolusi nama untuk dikaitkan dengan titik akhir server Anda.

Misalnya, domain kustom Anda mungkin `ftp.inbox.example.com`. Nama host khusus menggunakan DNS nama yang Anda berikan dan dapat diselesaikan oleh DNS layanan. Anda dapat menggunakan Route 53 sebagai DNS resolver Anda, atau menggunakan penyedia DNS layanan Anda sendiri. Untuk mempelajari cara AWS Transfer Family menggunakan Route 53 untuk merutekan lalu lintas dari domain kustom Anda ke titik akhir server, lihat [Bekerja dengan nama host khusus](#).



4. Pilih Simpan. Anda dikembalikan ke halaman detail Server.

Edit konfigurasi pencatatan Anda

Di AWS Transfer Family konsol, Anda dapat mengubah konfigurasi logging Anda.

Note

Jika Transfer Family membuat IAM peran CloudWatch logging untuk Anda saat membuat server, IAM peran tersebut akan dipanggil `AWSTransferLoggingAccess`. Anda dapat menggunakannya untuk semua server Transfer Family Anda.

Untuk mengedit konfigurasi logging Anda

1. Pada halaman Detail Server, pilih Edit di samping Detail tambahan.
2. Berdasarkan konfigurasi Anda, pilih antara peran logging, JSON logging terstruktur, atau keduanya. Untuk informasi selengkapnya, lihat [Memperbarui logging untuk server](#).

Edit kebijakan keamanan

Prosedur ini menjelaskan cara mengubah kebijakan keamanan server Transfer Family dengan menggunakan AWS Transfer Family konsol atau AWS CLI.

Note

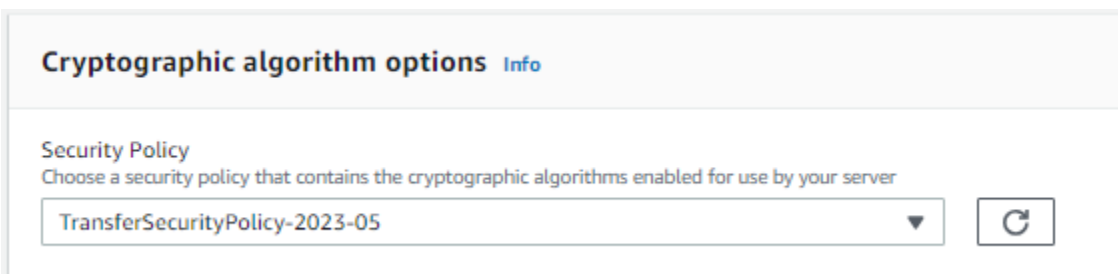
Jika titik akhir Anda FIPS diaktifkan, Anda tidak dapat mengubah kebijakan FIPS keamanan menjadi kebijakan FIPS non-keamanan.

Console

Untuk mengedit kebijakan keamanan dengan menggunakan konsol

1. Pada halaman Detail Server, pilih Edit di samping Detail tambahan.
2. Di bagian Opsi Algoritma Kriptografi, pilih kebijakan keamanan yang berisi algoritme kriptografi yang diaktifkan untuk digunakan oleh server Anda.

Untuk informasi selengkapnya tentang kebijakan keamanan, lihat [Kebijakan keamanan untuk AWS Transfer Family](#).



Cryptographic algorithm options Info

Security Policy
Choose a security policy that contains the cryptographic algorithms enabled for use by your server

TransferSecurityPolicy-2023-05

3. Pilih Simpan.

Anda dikembalikan ke halaman Detail Server di mana Anda dapat melihat kebijakan keamanan yang diperbarui.

AWS CLI

Untuk mengedit kebijakan keamanan dengan menggunakan CLI

1. Jalankan perintah berikut untuk melihat kebijakan keamanan saat ini yang dilampirkan ke server Anda.

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

`describe-server` Perintah ini mengembalikan semua detail untuk server Anda, termasuk baris berikut:

```
"SecurityPolicyName": "TransferSecurityPolicy-2018-11"
```

Dalam hal ini, kebijakan keamanan untuk server adalah `TransferSecurityPolicy-2018-11`.

2. Pastikan untuk memberikan nama yang tepat dari kebijakan keamanan ke perintah. Misalnya, jalankan perintah berikut untuk memperbarui server ke `TransferSecurityPolicy-2023-05`.

```
aws transfer update-server --server-id your-server-id --security-policy-name  
"TransferSecurityPolicy-2023-05"
```

 Note

Nama-nama kebijakan keamanan yang tersedia tercantum di [Kebijakan keamanan untuk AWS Transfer Family](#).

Jika berhasil, perintah mengembalikan kode berikut, dan memperbarui kebijakan keamanan server Anda.

```
{  
  "ServerId": "your-server-id"  
}
```

Mengubah alur kerja terkelola untuk server Anda

Di AWS Transfer Family konsol, Anda dapat mengubah alur kerja terkelola yang terkait dengan server.

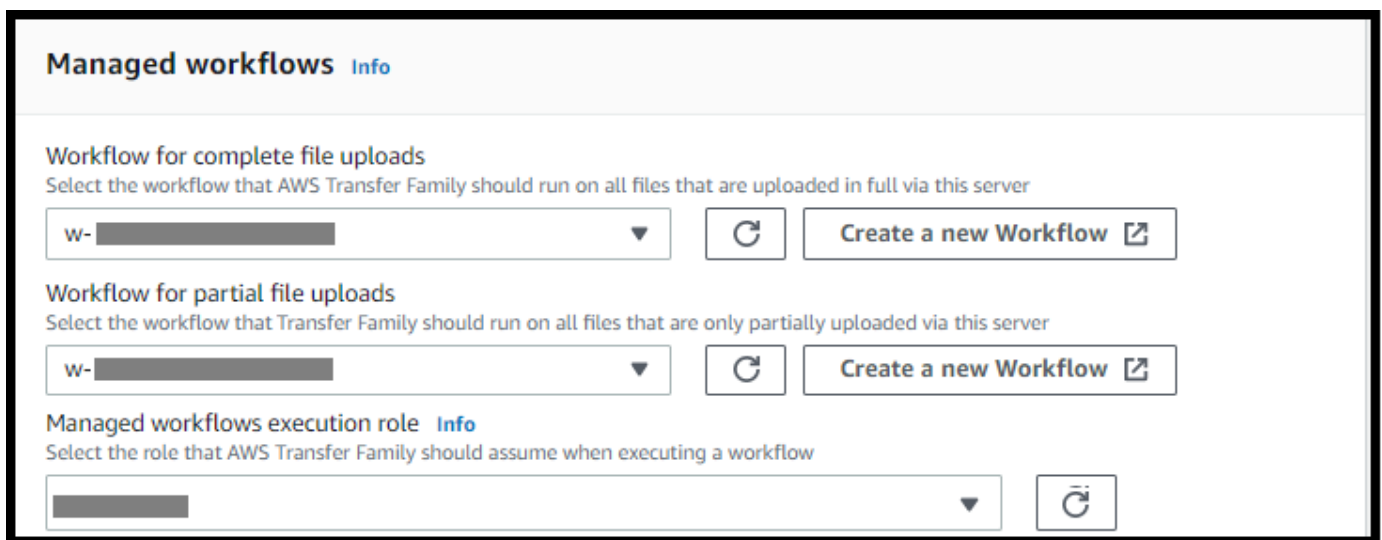
Untuk mengubah alur kerja terkelola

1. Pada halaman Detail Server, pilih Edit di samping Detail tambahan.
2. Pada halaman Edit detail tambahan, di bagian Alur kerja terkelola, pilih alur kerja yang akan dijalankan di semua unggahan.

Note

Jika Anda belum memiliki alur kerja, pilih Buat alur kerja baru untuk membuatnya.

- a. Pilih ID alur kerja yang akan digunakan.
- b. Pilih peran eksekusi. Ini adalah peran yang diasumsikan Transfer Family saat menjalankan langkah-langkah alur kerja. Untuk informasi selengkapnya, lihat [IAMkebijakan untuk alur kerja](#). Pilih Simpan.



3. Pilih Simpan. Anda dikembalikan ke halaman detail Server.

Ubah spanduk tampilan untuk server Anda

Di AWS Transfer Family konsol, Anda dapat mengubah spanduk tampilan yang terkait dengan server.

Untuk mengubah spanduk tampilan

1. Pada halaman Detail Server, pilih Edit di samping Detail tambahan.
2. Pada halaman Edit detail tambahan, di bagian Tampilan spanduk, masukkan teks untuk spanduk tampilan yang tersedia.
3. Pilih Simpan. Anda dikembalikan ke halaman detail Server.

Menempatkan server Anda secara online atau offline

Di AWS Transfer Family konsol, Anda dapat membawa server Anda online atau membawanya offline.

Untuk membawa server Anda secara online

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Di panel navigasi, pilih Server.
3. Pilih kotak centang server yang sedang offline.
4. Untuk Tindakan, pilih Mulai.

Diperlukan beberapa menit bagi server untuk beralih dari offline ke online.

Note

Ketika Anda menghentikan server untuk membuatnya offline, saat ini Anda masih dikenakan biaya layanan untuk server itu. Untuk menghilangkan biaya berbasis server tambahan, hapus server itu.

Untuk membuat server Anda offline

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Di panel navigasi, pilih Server.
3. Pilih kotak centang server yang sedang online.
4. Untuk Tindakan, pilih Berhenti.

Saat server memulai atau mematikan, server tidak tersedia untuk operasi file. Konsol tidak menampilkan status mulai dan berhenti.

Jika Anda menemukan kondisi kesalahan `START_FAILED` atau `STOP_FAILED`, hubungi AWS Dukungan untuk membantu menyelesaikan masalah Anda.

Kelola kunci host untuk server SFTP yang diaktifkan

Important

Jika Anda tidak berencana untuk memigrasikan pengguna yang ada dari server SFTP berkemampuan -yang ada ke server baru yang SFTP diaktifkan, abaikan bagian ini. Mengubah kunci host server secara tidak sengaja dapat mengganggu. Bergantung pada bagaimana SFTP klien Anda dikonfigurasi, itu dapat segera gagal, dengan pesan bahwa tidak ada kunci host tepercaya, atau menyajikan petunjuk yang mengancam. Jika ada skrip untuk mengotomatisasi koneksi, kemungkinan besar akan gagal juga.

Secara default, AWS Transfer Family berikan kunci host untuk server SFTP -enabled Anda. Anda dapat mengganti kunci host default dengan kunci host dari server lain. Lakukan hanya jika Anda berencana untuk memindahkan pengguna yang ada dari server SFTP -enabled yang ada ke server baru yang SFTP diaktifkan.

Untuk mencegah pengguna diminta untuk memverifikasi keaslian server SFTP -enabled Anda lagi, impor kunci host untuk server lokal Anda ke server yang diaktifkan. SFTP Melakukan hal ini juga mencegah pengguna Anda mendapatkan peringatan tentang potensi man-in-the-middle serangan.

Anda juga dapat memutar kunci host secara berkala, sebagai langkah keamanan tambahan.

Note

Meskipun konsol Transfer Family memungkinkan Anda untuk menentukan dan menambahkan kunci host server untuk semua server, kunci ini hanya berguna untuk server yang menggunakan SFTP protokol.

Topik

- [Tambahkan kunci host server tambahan](#)
- [Hapus kunci host server](#)
- [Putar kunci host server](#)
- [Informasi kunci host server tambahan](#)

Tambahkan kunci host server tambahan

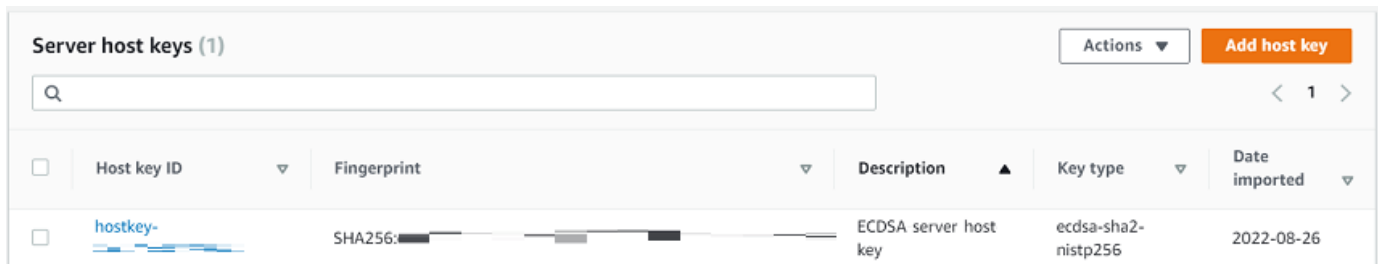
Di AWS Transfer Family konsol, Anda dapat menambahkan kunci host server tambahan. Menambahkan kunci host tambahan dari format yang berbeda dapat berguna untuk mengidentifikasi server ketika klien terhubung dengannya, serta meningkatkan profil keamanan Anda. Misalnya, jika kunci asli Anda adalah RSA kunci, Anda dapat menambahkan ECDSA kunci tambahan.

Note

SFTP klien terhubung menggunakan kunci publik pertama yang dimilikinya yang dapat mencocokkan salah satu kunci server aktif.

Untuk menambahkan kunci host server tambahan

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Di panel navigasi kiri, pilih Server, lalu pilih server yang menggunakan SFTP protokol.
3. Pada halaman detail server, gulir ke bawah ke bagian Kunci host Server.



Server host keys (1)						Actions	Add host key
	Host key ID	Fingerprint	Description	Key type	Date imported		
☐	hostkey-	SHA256: [fingerprint]	ECDSA server host key	ecdsa-sha2-nistp256	2022-08-26		

4. Pilih Tambahkan kunci host.

Halaman kunci Add server host ditampilkan.

5. Di bagian Server Host Key, masukkan RSAECDSA,, atau kunci ED25519 pribadi yang digunakan untuk mengidentifikasi server Anda ketika klien terhubung ke SFTP server yang diaktifkan.

Note

Saat Anda membuat kunci host server, pastikan untuk menentukan -N "" (tidak ada frasa sandi). Lihat [Membuat SSH kunci di macOS, Linux, atau Unix](#) untuk detail tentang cara menghasilkan pasangan kunci.

Server Host Key [Info](#)

Private key - *optional*

Upload an RSA, ECDSA, or ED25519 private key that will be used to identify your SFTP server when clients connect to it. Additional keys can be added once the server is created.

Enter an optional RSA, ECDSA, or ED25519 key

Description - *optional*

Add a description to differentiate between multiple private keys

Enter optional description

 You can ignore this section unless you are migrating users from an existing SFTP server.

- (Optional) Tambahkan deskripsi untuk membedakan antara beberapa kunci host server. Anda juga dapat menambahkan tag untuk kunci Anda.
- Pilih Tambah kunci. Anda dikembalikan ke halaman detail Server.

Untuk menambahkan kunci host dengan menggunakan AWS Command Line Interface (AWS CLI), gunakan [the section called “ImportHostKey”](#) API operasi dan berikan kunci host baru. Jika Anda membuat SFTP server baru yang diaktifkan, Anda memberikan kunci host Anda sebagai parameter dalam [the section called “CreateServer”](#) API operasi. Anda juga dapat menggunakan AWS CLI untuk memperbarui deskripsi untuk kunci host yang ada.

Contoh import-host-key AWS CLI perintah berikut mengimpor kunci host untuk server SFTP - enabled tertentu.

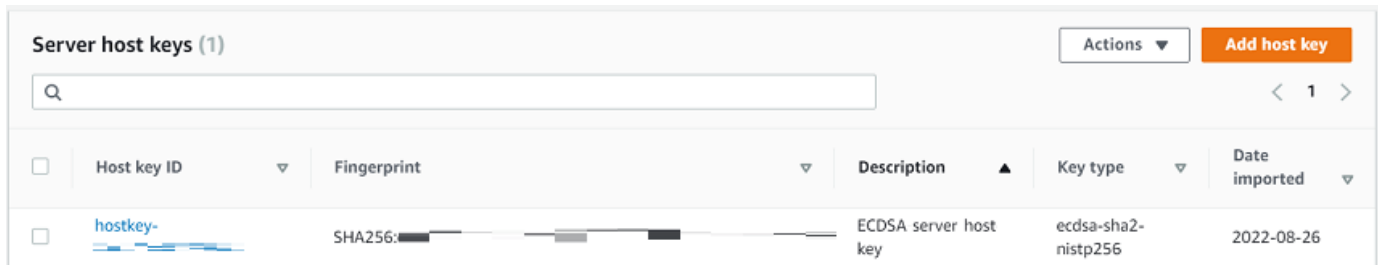
```
aws transfer import-host-key --description key-description --server-id your-server-id
--host-key-body file://my-host-key
```

Hapus kunci host server

Di AWS Transfer Family konsol, Anda dapat menghapus kunci host server.

Untuk menghapus kunci host server

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Di panel navigasi kiri, pilih Server, lalu pilih server yang menggunakan SFTP protokol.
3. Pada halaman detail server, gulir ke bawah ke bagian Kunci host Server.



Server host keys (1)					Actions	Add host key
	Host key ID	Fingerprint	Description	Key type	Date imported	
☐	hostkey-	SHA256: [fingerprint]	ECDSA server host key	ecdsa-sha2-nistp256	2022-08-26	

4. Di bagian Server Host Keys, pilih kunci, dan kemudian di bawah Tindakan, pilih Hapus.
5. Di kotak dialog konfirmasi yang muncul, masukkan kata **delete**, lalu pilih Hapus untuk mengonfirmasi bahwa Anda ingin menghapus kunci host.

Kunci host dihapus dari halaman Server.

Untuk menghapus kunci host dengan menggunakan AWS CLI, gunakan [the section called “DeleteHostKey”](#) API operasi dan berikan ID server dan ID kunci host.

Contoh `delete-host-key` AWS CLI perintah berikut menghapus kunci host untuk server SFTP - enabled yang ditentukan.

```
aws transfer delete-host-key --server-id your-server-id --host-key-id your-host-key-id
```

Putar kunci host server

Secara berkala, Anda dapat memutar kunci host server Anda.

Note

Transfer Family menggunakan kunci yang ditambahkan pertama untuk setiap algoritme sebagai kunci host aktif. Anda dapat mengaitkan hingga 10 kunci host per SFTP server, tetapi hanya satu kunci per algoritma yang aktif pada waktu tertentu.

Misalnya, asumsikan bahwa Anda telah menambahkan set kunci host server berikut ke server Anda.

Kunci host server

Jenis kunci host	Tanggal ditambahkan ke server	Aktif?
RSA	1 April 2020	Tidak
ECDSA	Februari 1, 2020	Tidak
ED25519	1 Desember, 2019	Tidak
RSA	1 Oktober 2019	Ya
ECDSA	1 Juni 2019	Ya
ED25519	Maret 1, 2019	Ya

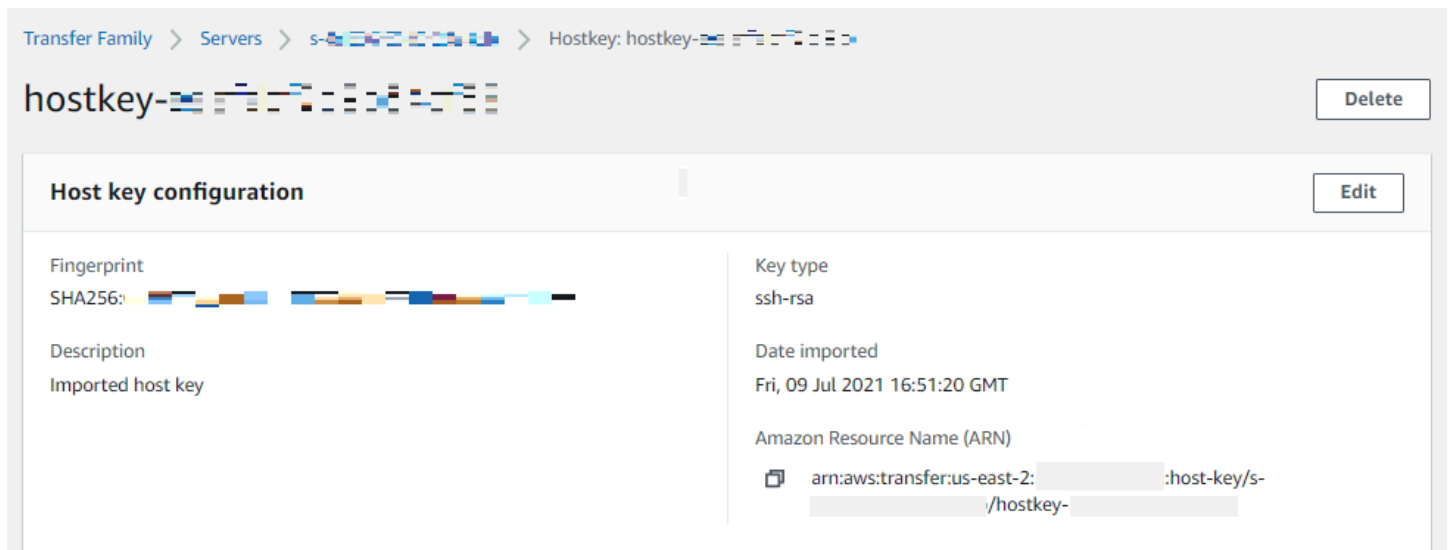
Kunci tertua untuk setiap algoritma aktif. Jika Anda menghapus RSA kunci yang Anda tambahkan pada 1 Oktober 2019, maka RSA kunci yang Anda tambahkan pada 1 April 2020 menjadi aktif.

Untuk memutar kunci host server

1. Tambahkan kunci host server baru. Prosedur ini dijelaskan dalam [Tambahkan kunci host server tambahan](#).
2. Hapus satu atau beberapa kunci host dari jenis yang sama yang telah Anda tambahkan sebelumnya. Prosedur ini dijelaskan dalam [Hapus kunci host server](#).
3. Pastikan bahwa kunci paling awal yang tersisa dari jenis yang sama adalah kunci yang ingin Anda aktifkan.

Informasi kunci host server tambahan

Anda dapat memilih kunci host untuk menampilkan detail kunci tersebut.



Anda dapat menghapus kunci host, atau mengedit deskripsinya dari menu Tindakan di layar Detail Server. Pilih tombol host, lalu pilih tindakan yang sesuai dari menu.



Memantau penggunaan di konsol

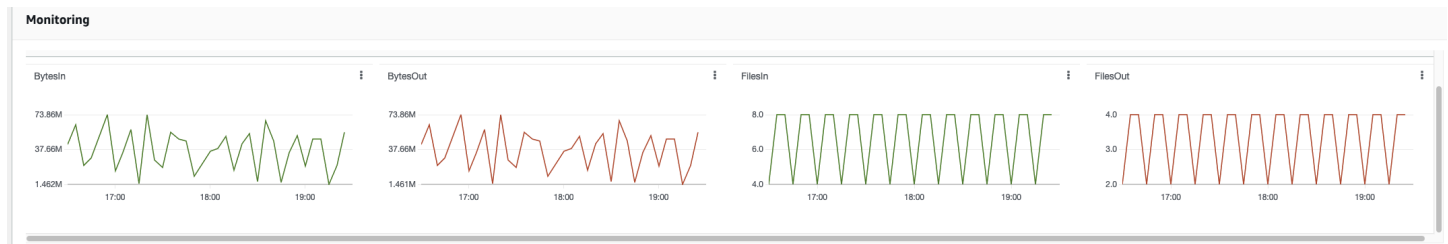
Anda bisa mendapatkan informasi tentang metrik server Anda di halaman detail Servernya. Ini memberi Anda satu tempat untuk memantau beban kerja transfer file Anda. Anda dapat melacak berapa banyak file yang telah Anda tukarkan dengan mitra Anda dan melacak penggunaannya dengan cermat menggunakan dasbor terpusat. Untuk detailnya, lihat [LihatSFTP,FTPS, dan detail FTP server](#). Tabel berikut menjelaskan metrik yang tersedia untuk Transfer Family.

Namespace	Metrik	Deskripsi
AWS/Transfer	BytesIn	Jumlah total byte yang ditransfer ke server.
		Unit: Jumlah

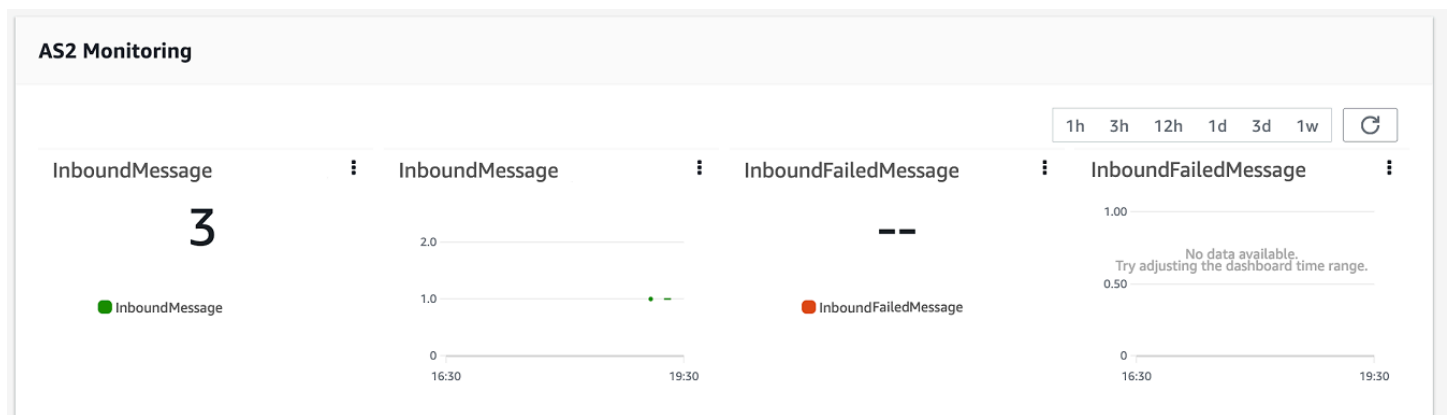
Namespace	Metrik	Deskripsi
		Periode: 5 menit
	BytesOut	Jumlah total byte yang ditransfer keluar dari server. Unit: Jumlah Periode: 5 menit
	FilesIn	Jumlah total file yang ditransfer ke server. Untuk server yang menggunakan AS2 protokol, metrik ini mewakili jumlah pesan yang diterima. Unit: Jumlah Periode: 5 menit
	FilesOut	Jumlah total file yang ditransfer keluar dari server. Unit: Jumlah Periode: 5 menit
	InboundMessage	Jumlah total AS2 pesan yang berhasil diterima dari mitra dagang. Unit: Jumlah Periode: 5 menit
	InboundFailedMessage	Jumlah total AS2 pesan yang tidak berhasil diterima dari mitra dagang. Artinya, mitra dagang mengirim pesan, tetapi server Transfer Family tidak berhasil memprosesnya. Unit: Jumlah Periode: 5 menit

Namespace	Metrik	Deskripsi
	OnPartialUploadExecutionsStarted	Jumlah total eksekusi on-partial-upload alur kerja dimulai di server. Unit: Jumlah Periode: 1 menit
	OnPartialUploadExecutionsSuccess	Jumlah total eksekusi on-partial-upload alur kerja yang berhasil di server. Unit: Jumlah Periode: 1 menit
	OnPartialUploadExecutionsFailed	Jumlah total eksekusi on-partial-upload alur kerja yang gagal di server. Unit: Jumlah Periode: 1 menit
	OnUploadExecutionsStarted	Jumlah total eksekusi alur kerja dimulai di server. Unit: Jumlah Periode: 1 menit
	OnUploadExecutionsSuccess	Jumlah total eksekusi alur kerja yang berhasil di server. Unit: Jumlah Periode: 1 menit
	OnUploadExecutionsFailed	Jumlah total eksekusi alur kerja yang gagal di server. Unit: Jumlah Periode: 1 menit

Bagian Pemantauan berisi empat grafik individual. Grafik ini menunjukkan byte masuk, byte keluar, file masuk, dan file keluar.



Untuk server yang memiliki AS2 protokol diaktifkan, ada bagian AS2Monitoring di bawah informasi Monitoring. Bagian ini berisi detail untuk jumlah pesan masuk, baik yang berhasil maupun yang gagal.



Untuk membuka grafik yang dipilih di jendelanya sendiri, pilih ikon perluas



Anda juga dapat mengklik ikon elipsis vertikal grafik



untuk membuka menu tarik-turun dengan item berikut:

- **Memperbesar** — Membuka grafik yang dipilih di jendelanya sendiri.
- **Refresh** — Muat ulang grafik dengan data terbaru.
- **Lihat dalam metrik** — Membuka detail metrik yang sesuai di Amazon. CloudWatch
- **Lihat log** - Membuka grup log yang sesuai di CloudWatch.

Mengelola kontrol akses

Anda dapat mengontrol akses pengguna ke AWS Transfer Family sumber daya dengan menggunakan kebijakan AWS Identity and Access Management (IAM). IAMKebijakan adalah pernyataan, biasanya dalam JSON format, yang memungkinkan tingkat akses tertentu ke sumber daya. Anda menggunakan IAM kebijakan untuk menentukan operasi file apa yang ingin Anda izinkan untuk dilakukan dan tidak dilakukan oleh pengguna. Anda juga dapat menggunakan IAM kebijakan untuk menentukan bucket atau bucket Amazon S3 yang ingin Anda akses kepada pengguna. Untuk menentukan kebijakan ini bagi pengguna, Anda membuat IAM peran AWS Transfer Family yang memiliki hubungan IAM kebijakan dan kepercayaan yang terkait dengannya.

Setiap pengguna diberi IAM peran. Jenis IAM peran yang AWS Transfer Family digunakan disebut peran layanan. Saat pengguna masuk ke server Anda, AWS Transfer Family asumsikan IAM peran yang dipetakan ke pengguna. Untuk mempelajari cara membuat IAM peran yang menyediakan akses pengguna ke bucket Amazon S3, lihat [Membuat peran untuk mendelegasikan izin ke AWS layanan](#) di Panduan Pengguna. IAM

Anda dapat memberikan akses hanya tulis ke objek Amazon S3 dengan menggunakan izin tertentu dalam kebijakan. IAM Untuk detailnya, lihat [Memberikan kemampuan untuk hanya menulis dan membuat daftar file](#).

Blog AWS Penyimpanan berisi posting yang merinci cara mengatur akses hak istimewa paling sedikit. Untuk detailnya, lihat [Menerapkan akses hak istimewa terkecil dalam AWS Transfer Family alur kerja](#).

Note

Jika bucket Amazon S3 dienkripsi menggunakan AWS Key Management Service (AWS KMS), Anda harus menentukan izin tambahan dalam kebijakan Anda. Untuk detailnya, lihat [Enkripsi data di Amazon S3](#). Selain itu, Anda dapat melihat informasi selengkapnya tentang [kebijakan sesi](#) di Panduan IAM Pengguna.

Topik

- [Mengizinkan akses baca dan tulis ke bucket Amazon S3](#)
- [Membuat kebijakan sesi untuk bucket Amazon S3](#)

- [Mencegah pengguna berjalan mkdir di bucket S3](#)

Mengizinkan akses baca dan tulis ke bucket Amazon S3

Bagian ini menjelaskan cara membuat IAM kebijakan yang memungkinkan akses baca dan tulis ke bucket Amazon S3 tertentu. Menetapkan IAM peran yang memiliki IAM kebijakan ini kepada pengguna Anda memberi pengguna akses baca/tulis ke bucket Amazon S3 yang ditentukan.

Kebijakan berikut menyediakan akses baca, tulis, dan penandaan terprogram ke bucket Amazon S3. PutObjectACLPernyataan GetObjectACL dan pernyataan hanya diperlukan jika Anda perlu mengaktifkan Akses Lintas Akun. Artinya, server Transfer Family Anda perlu mengakses bucket di akun yang berbeda.

```
{
  "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/*"]
    }
  ]
}
```


ListBucketTindakan tersebut membutuhkan izin untuk ember itu sendiri.

DELETETindakanPUT,GET, dan memerlukan izin objek. Karena ini adalah sumber daya yang berbeda, mereka ditentukan menggunakan Nama Sumber Daya Amazon yang berbeda (ARNs).

Untuk lebih membatasi akses pengguna hanya ke home awalan bucket Amazon S3 yang ditentukan, lihat. [Membuat kebijakan sesi untuk bucket Amazon S3](#)

Membuat kebijakan sesi untuk bucket Amazon S3

Kebijakan sesi adalah kebijakan AWS Identity and Access Management (IAM) yang membatasi pengguna pada bagian tertentu dari bucket Amazon S3. Ia melakukannya dengan mengevaluasi akses secara real time.

Note

Kebijakan sesi hanya digunakan dengan Amazon S3. Untuk AmazonEFS, Anda menggunakan izin POSIX file untuk membatasi akses.

Anda dapat menggunakan kebijakan sesi saat Anda perlu memberikan akses yang sama ke sekelompok pengguna ke bagian tertentu dari bucket Amazon S3 Anda. Misalnya, sekelompok pengguna mungkin hanya memerlukan akses ke home direktori. Kelompok pengguna itu berbagi IAM peran yang sama.

Note

Panjang maksimum kebijakan sesi adalah 2048 karakter. Untuk detail selengkapnya, lihat [parameter Permintaan kebijakan](#) untuk CreateUser tindakan dalam APIreferensi.

Untuk membuat kebijakan sesi, gunakan variabel kebijakan berikut dalam IAM kebijakan Anda:

- `${transfer:HomeBucket}`
- `${transfer:HomeDirectory}`
- `${transfer:HomeFolder}`
- `${transfer:UserName}`

⚠ Important

Anda tidak dapat menggunakan variabel sebelumnya dalam Kebijakan Terkelola. Anda juga tidak dapat menggunakannya sebagai variabel kebijakan dalam definisi IAM peran. Anda membuat variabel-variabel ini dalam IAM kebijakan dan menyediakannya secara langsung saat menyiapkan pengguna Anda. Selain itu, Anda tidak dapat menggunakan `${aws:Username}` variabel dalam kebijakan sesi ini. Variabel ini mengacu pada nama IAM pengguna dan bukan nama pengguna yang diperlukan oleh AWS Transfer Family.

Kode berikut menunjukkan contoh kebijakan sesi.

```
{
  "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

Contoh kebijakan sebelumnya mengasumsikan bahwa pengguna memiliki direktori home mereka diatur untuk menyertakan garis miring, untuk menandakan bahwa itu adalah direktori. Jika, di sisi lain, Anda menetapkan pengguna `HomeDirectory` tanpa garis miring, maka Anda harus memasukkannya sebagai bagian dari kebijakan Anda.

Dalam contoh kebijakan sebelumnya, perhatikan penggunaan parameter `transfer:HomeFolder`, `transfer:HomeBucket`, dan `transfer:HomeDirectory` kebijakan. Parameter ini diatur untuk `HomeDirectory` yang dikonfigurasi untuk pengguna, seperti yang dijelaskan dalam [HomeDirectory](#) dan [Menerapkan metode API Gateway Anda](#). Parameter ini memiliki definisi berikut:

- `transfer:HomeBucketParameter` diganti dengan komponen pertama `HomeDirectory`.
- `transfer:HomeFolderParameter` diganti dengan bagian `HomeDirectory` parameter yang tersisa.
- `transfer:HomeDirectoryParameter` memiliki garis miring depan (/) dihapus sehingga dapat digunakan sebagai bagian dari S3 Amazon Resource Name (ARN) dalam sebuah Resource pernyataan.

Note

Jika Anda menggunakan direktori logik—yaitu, pengguna adalah LOGICAL —parameter kebijakan ini (`HomeBucket`, `HomeDirectory`, dan `HomeFolder`) tidak didukung.

`homeDirectoryType`

Misalnya, asumsikan bahwa HomeDirectory parameter yang dikonfigurasi untuk pengguna Transfer Family adalah `/home/bob/amazon/stuff/`.

- `transfer:HomeBucket` diatur ke `/home`.
- `transfer:HomeFolder` diatur ke `/bob/amazon/stuff/`.
- `transfer:HomeDirectory` menjadi `/home/bob/amazon/stuff/`.

Yang pertama "Sid" memungkinkan pengguna untuk membuat daftar semua direktori mulai dari `/home/bob/amazon/stuff/`.

Yang kedua "Sid" membatasi pengguna put dan get akses ke jalur yang sama, `/home/bob/amazon/stuff/`.

Dengan kebijakan sebelumnya, saat pengguna masuk, mereka hanya dapat mengakses objek di direktori home mereka. Pada waktu koneksi, AWS Transfer Family ganti variabel-variabel ini dengan nilai yang sesuai untuk pengguna. Melakukan hal ini memudahkan penerapan dokumen kebijakan yang sama ke beberapa pengguna. Pendekatan ini mengurangi overhead manajemen IAM peran dan kebijakan untuk mengelola akses pengguna ke bucket Amazon S3 Anda.

Anda juga dapat menggunakan kebijakan sesi untuk menyesuaikan akses untuk setiap pengguna berdasarkan kebutuhan bisnis Anda. Untuk informasi [selengkapnya, lihat Izin untuk AssumeRole AssumeRoleWithSAML, dan AssumeRoleWithWebIdentity](#) di Panduan IAM Pengguna.

Note

AWS Transfer Family menyimpan kebijakan JSON, bukan Amazon Resource Name (ARN) kebijakan. Jadi, ketika Anda mengubah kebijakan di IAM konsol, Anda harus kembali ke AWS Transfer Family konsol dan memperbarui pengguna Anda dengan konten kebijakan terbaru. Anda dapat memperbarui pengguna pada tab Info Kebijakan di bagian Konfigurasi pengguna. Jika Anda menggunakan AWS CLI, Anda dapat menggunakan perintah berikut untuk memperbarui kebijakan.

```
aws transfer update-user --server-id server --user-name user --policy \  
    "$(aws iam get-policy-version --policy-arn policy --version-id version --  
    output json)"
```

Mencegah pengguna berjalan **mkdir** di bucket S3

Anda dapat membatasi kemampuan pengguna untuk membuat direktori di bucket Amazon S3. Untuk melakukannya, Anda membuat IAM kebijakan yang mengizinkan `s3:PutObject` tindakan tetapi juga menyangkalnya ketika kunci diakhiri dengan `/` (garis miring ke depan). Kebijakan contoh berikut memungkinkan pengguna untuk mengunggah file ke bucket Amazon S3 tetapi menolak `mkdir` perintah di bucket Amazon S3.

```
{
  "Sid": "DenyMkdir",
  "Action": [
    "s3:PutObject"
  ],
  "Effect": "Deny",
  "Resource": [
    "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*/",
    "arn:aws:s3:::DOC-EXAMPLE-BUCKET/*/*"
  ]
}
```

Note

Baris sumber daya kedua membuat pengguna tidak mungkin membuat sub-folder dengan menjalankan perintah seperti `my-file DOC-EXAMPLE-BUCKET/new-folder/my-file`.

Penebangan untuk AWS Transfer Family

AWS Transfer Family terintegrasi dengan keduanya AWS CloudTrail dan Amazon CloudWatch. CloudTrail dan CloudWatch melayani tujuan yang berbeda tetapi saling melengkapi:

- CloudTrail adalah AWS layanan yang membuat catatan tindakan yang diambil dalam diri Anda Akun AWS. Ini terus memantau dan merekam API panggilan untuk aktivitas seperti login konsol, AWS Command Line Interface perintah, dan SDK/API panggilan. Ini memungkinkan Anda untuk menyimpan log siapa yang mengambil tindakan apa, kapan, dan dari mana. CloudTrail membantu audit, manajemen akses, dan kepatuhan terhadap peraturan dengan memberikan riwayat semua aktivitas di AWS lingkungan Anda. Untuk detailnya, lihat [Panduan AWS CloudTrail Pengguna](#).
- CloudWatch adalah layanan pemantauan untuk AWS sumber daya dan aplikasi. Ini mengumpulkan metrik dan log untuk memberikan visibilitas ke pemanfaatan sumber daya, kinerja aplikasi, dan kesehatan sistem secara keseluruhan. CloudWatch membantu tugas operasional seperti pemecahan masalah, pengaturan alarm, dan penskalaan otomatis. Untuk detailnya, lihat [Panduan CloudWatch Pengguna Amazon](#).

Topik

- [AWS CloudTrail penebangan untuk AWS Transfer Family](#)
- [CloudWatch Pencatatan Amazon untuk AWS Transfer Family](#)

AWS CloudTrail penebangan untuk AWS Transfer Family

AWS Transfer Family terintegrasi dengan AWS CloudTrail, layanan yang menyediakan catatan tindakan yang diambil oleh pengguna, peran, atau AWS layanan di AWS Transfer Family. CloudTrail menangkap semua panggilan API untuk AWS Transfer Family sebagai peristiwa. Panggilan yang direkam mencakup panggilan dari AWS Transfer Family konsol dan panggilan kode ke operasi API AWS Transfer Family ini.

Trail adalah konfigurasi yang memungkinkan pengiriman peristiwa sebagai file log ke bucket Amazon S3 yang Anda tentukan. CloudTrail file log berisi satu atau lebih entri log. Peristiwa mewakili permintaan tunggal dari sumber manapun dan mencakup informasi tentang tindakan yang diminta, tanggal dan waktu tindakan, parameter permintaan, dan sebagainya. CloudTrail file log bukanlah jejak tumpukan yang diurutkan dari panggilan API publik, jadi file tersebut tidak muncul dalam urutan tertentu.

Untuk catatan berkelanjutan tentang peristiwa di akun AWS Anda, termasuk peristiwa untuk AWS Transfer Family, buat jejak. Jejak memungkinkan CloudTrail untuk mengirimkan file log ke bucket Amazon S3. Secara default, saat Anda membuat jejak di konsol tersebut, jejak diterapkan ke semua Wilayah AWS. Jejak mencatat peristiwa dari semua Wilayah di partisi AWS dan mengirimkan file log ke bucket Amazon S3 yang Anda tentukan. Selain itu, Anda dapat mengonfigurasi AWS layanan lain untuk menganalisis lebih lanjut dan menindaklanjuti data peristiwa yang dikumpulkan dalam CloudTrail log. Untuk informasi selengkapnya, lihat berikut:

- [Gambaran umum untuk membuat jejak](#)
- [CloudTrail layanan dan integrasi yang didukung](#)
- [Mengonfigurasi notifikasi Amazon SNS untuk CloudTrail](#)
- [Menerima file CloudTrail log dari beberapa wilayah](#) dan [Menerima file CloudTrail log dari beberapa akun](#)

Semua AWS Transfer Family tindakan dicatat oleh CloudTrail dan didokumentasikan dalam [ActionsAPI reference](#). Misalnya, panggilan `keCreateServer`, `ListUsers` dan `StopServer` tindakan menghasilkan entri dalam file CloudTrail log.

Setiap peristiwa atau entri log berisi informasi tentang siapa yang membuat permintaan tersebut. Informasi identitas membantu Anda menentukan hal berikut:

- Baik permintaan tersebut dibuat dengan kredensial pengguna root atau AWS Identity and Access Management.
- Baik permintaan tersebut dibuat dengan kredensial keamanan sementara untuk peran atau pengguna gabungan.
- Apakah permintaan dibuat oleh layanan AWS lain.

Untuk informasi selengkapnya, lihat [Elemen userIdentity CloudTrail](#).

Jika Anda membuat jejak, Anda dapat mengaktifkan pengiriman CloudTrail acara secara berkelanjutan ke bucket Amazon S3, termasuk acara untuk AWS Transfer Family. Jika Anda tidak mengonfigurasi jejak, Anda masih dapat melihat peristiwa terbaru di CloudTrail konsol dalam Riwayat acara.

Dengan menggunakan informasi yang dikumpulkan oleh CloudTrail, Anda dapat menentukan permintaan yang dibuat AWS Transfer Family, alamat IP dari mana permintaan dibuat, siapa yang membuat permintaan, kapan dibuat, dan detail tambahan.

Untuk mempelajari selengkapnya CloudTrail, lihat [Panduan AWS CloudTrail Pengguna](#).

Topik

- [Aktifkan AWS CloudTrail pencatatan](#)
- [Contoh entri log untuk membuat server](#)

Aktifkan AWS CloudTrail pencatatan

Anda dapat memantau panggilan AWS Transfer Family API menggunakan AWS CloudTrail. Dengan memantau panggilan API, Anda bisa mendapatkan informasi keamanan dan operasional yang berguna. Jika Anda [mengaktifkan pencatatan level objek Amazon S3](#), RoleSessionName terdapat dalam bidang Peminta sebagai [AWS:Role Unique Identifier]/username.sessionid@server-id Untuk informasi selengkapnya tentang pengidentifikasi unik peran AWS Identity and Access Management (IAM), lihat [Pengidentifikasi unik di Panduan Pengguna](#). AWS Identity and Access Management

Important

Panjang maksimum RoleSessionName adalah 64 karakter. Jika RoleSessionName lebih panjang, server-id akan terpotong.

Contoh entri log untuk membuat server

Contoh berikut menunjukkan entri CloudTrail log (dalam format JSON) yang menunjukkan tindakan. CreateServer

```
{
  "eventVersion": "1.09",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "AAAA4FFF5HHHHH6NNWWW: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": "AAAA4FFF5HHHHH6NNWWW",
      "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"  
}
```

CloudWatch Pencatatan Amazon untuk AWS Transfer Family

Amazon CloudWatch memantau AWS Transfer Family sumber daya Anda dan aplikasi yang Anda jalankan AWS secara real time. Anda dapat menggunakan CloudWatch untuk mengumpulkan dan melacak metrik, yang merupakan variabel yang dapat Anda ukur untuk sumber daya dan aplikasi Anda.

CloudWatch Halaman beranda secara otomatis menampilkan metrik tentang Transfer Family dan setiap AWS layanan lain yang Anda gunakan. Anda dapat membuat dasbor kustom tambahan untuk menampilkan metrik tentang aplikasi khusus Anda, dan menampilkan koleksi metrik-metrik kustom yang Anda pilih.

Anda dapat membuat alarm yang mengawasi metrik dan mengirimkan notifikasi atau secara otomatis melakukan perubahan pada sumber daya yang sedang dipantau ketika ada pelanggaran ambang batas. Misalnya, Anda dapat memantau file yang ditransfer ke server Transfer Family dan menggunakan data tersebut untuk menentukan apakah Anda perlu menggunakan server tambahan untuk menangani peningkatan beban. Anda juga dapat menggunakan data ini untuk menghentikan atau menghapus instance yang kurang digunakan untuk menghemat uang.

Topik

- [Membuat, memperbarui, dan melihat logging untuk server](#)
- [Mengelola logging untuk alur kerja](#)
- [Konfigurasi peran CloudWatch logging](#)
- [Melihat aliran log Transfer Family](#)
- [Membuat CloudWatch alarm Amazon](#)
- [Mencatat API panggilan Amazon S3 ke log akses S3](#)
- [Contoh untuk membatasi masalah wakil yang membingungkan](#)
- [CloudWatch struktur log untuk Transfer Family](#)
- [Contoh entri CloudWatch log](#)

- [Menggunakan CloudWatch metrik untuk Transfer Family](#)
- [Menggunakan Notifikasi Pengguna AWS dengan AWS Transfer Family](#)

Membuat, memperbarui, dan melihat logging untuk server

Untuk semua AWS Transfer Family server, Anda dapat memilih di antara dua opsi untuk logging: `LoggingRole` (digunakan untuk mencatat alur kerja yang dilampirkan ke server) atau `StructuredLogDestinations`. Manfaat menggunakan `StructuredLogDestinations` meliputi:

- Menerima log dalam JSON format terstruktur.
- Kueri log Anda dengan Amazon CloudWatch Logs Insights, yang secara otomatis menemukan bidang yang JSON diformat.
- Bagikan grup log di seluruh AWS Transfer Family sumber daya memungkinkan Anda menggabungkan aliran log dari beberapa server ke dalam satu grup log, sehingga memudahkan pengelolaan konfigurasi pemantauan dan pengaturan penyimpanan log.
- Buat metrik dan visualisasi agregat yang dapat ditambahkan ke dasbor. CloudWatch
- Lacak data penggunaan dan kinerja dengan menggunakan grup log untuk membuat metrik log, visualisasi, dan dasbor terkonsolidasi.

Opsi untuk `LoggingRole` atau `StructuredLogDestinations` dikonfigurasi dan dikendalikan secara terpisah. Untuk setiap server, Anda dapat mengatur satu atau kedua metode logging, atau mengonfigurasi server Anda agar tidak memiliki logging apa pun (meskipun ini tidak disarankan).

Jika Anda membuat server baru menggunakan konsol Transfer Family, logging diaktifkan secara default. Setelah Anda membuat server, Anda dapat menggunakan `UpdateServer` API panggilan untuk mengubah konfigurasi logging Anda. Untuk detailnya, lihat [StructuredLogDestinations](#).

Saat ini, untuk alur kerja, jika ingin logging diaktifkan, Anda harus menentukan peran logging:

- Jika Anda mengaitkan alur kerja dengan server, menggunakan `UpdateServer` API panggilan `CreateServer` atau, sistem tidak secara otomatis membuat peran logging. Jika Anda ingin mencatat peristiwa alur kerja Anda, Anda perlu melampirkan peran logging secara eksplisit ke server.
- Jika Anda membuat server menggunakan konsol Transfer Family dan melampirkan alur kerja, log akan dikirim ke grup log yang berisi ID server dalam nama. Formatnya `/aws/transfer/server-`

id, misalnya, `/aws/transfer/s-1111aaaa2222bbbb3`. Log server dapat dikirim ke grup log yang sama atau yang lain.

Pertimbangan log untuk membuat dan mengedit server di konsol

- Server baru yang dibuat melalui konsol hanya mendukung JSON logging terstruktur, kecuali alur kerja dilampirkan ke server.
- Tidak ada logging bukanlah opsi untuk server baru yang Anda buat di konsol.
- Server yang ada dapat mengaktifkan JSON logging terstruktur melalui konsol kapan saja.
- Mengaktifkan JSON logging terstruktur melalui konsol menonaktifkan metode logging yang ada, agar tidak menagih pelanggan dua kali lipat. Pengecualiannya adalah jika alur kerja dilampirkan ke server.
- Jika Anda mengaktifkan JSON logging terstruktur, Anda tidak dapat menonaktifkannya nanti melalui konsol.
- Jika Anda mengaktifkan JSON logging terstruktur, Anda dapat mengubah tujuan grup log melalui konsol kapan saja.
- Jika Anda mengaktifkan JSON logging terstruktur, Anda tidak dapat mengedit peran logging melalui konsol jika Anda telah mengaktifkan kedua jenis logging melalui API. Pengecualiannya adalah jika server Anda memiliki alur kerja yang terpasang. Namun, peran logging terus muncul di Detail tambahan.

Pertimbangan logging untuk membuat dan mengedit server menggunakan atau API SDK

- Jika Anda membuat server baru melalui API, Anda dapat mengonfigurasi salah satu atau kedua jenis logging, atau memilih tidak ada logging.
- Untuk server yang ada, aktifkan dan nonaktifkan JSON logging terstruktur kapan saja.
- Anda dapat mengubah grup log melalui API kapan saja.
- Anda dapat mengubah peran logging melalui setiap API saat.

Untuk mengaktifkan logging terstruktur, Anda harus masuk ke akun dengan izin berikut

- `logs:CreateLogDelivery`
- `logs>DeleteLogDelivery`
- `logs:DescribeLogGroups`

- logs:DescribeResourcePolicies
- logs:GetLogDelivery
- logs:ListLogDeliveries
- logs:PutResourcePolicy
- logs:UpdateLogDelivery

Contoh kebijakan tersedia di bagian ini [Konfigurasi peran CloudWatch logging](#).

Topik

- [Membuat logging untuk server](#)
- [Memperbarui logging untuk server](#)
- [Melihat konfigurasi server](#)

Membuat logging untuk server

Saat Anda membuat server baru, pada halaman Konfigurasi detail tambahan, Anda dapat menentukan grup log yang ada, atau membuat yang baru.

Transfer Family > Servers > Create server

Step 1
Choose protocols

Step 2
Choose an identity provider

Step 3
Choose an endpoint

Step 4
Choose a domain

Step 5
Configure additional details

Step 6
Review and create

Configure additional details

Logging Info

Log group Info
Choose the CloudWatch log group where your events will be delivered in a structured JSON format

☒ Create a new log group ☐ Choose an existing log group

Choose an existing log group

Logging role Info
Choose the IAM role that will be used to deliver events to your CloudWatch logs

☐ Create a new role ☐ Choose an existing role

Logging role is only required when selecting a workflow in the Managed workflows section below.

Jika Anda memilih grup log yang ada, Anda harus memilih salah satu yang terkait dengan Akun AWS.

Transfer Family > Servers > Create server

Step 1
Choose protocols

Step 2
Choose an identity provider

Step 3
Choose an endpoint

Step 4
Choose a domain

Step 5
Configure additional details

Step 6
Review and create

Configure additional details

Logging Info

Log group Info
Choose the CloudWatch log group where your events will be delivered in a structured JSON format

☐ Create a new log group ☒ Choose an existing log group

Logging role Info
Choose the IAM role that will be used to deliver events to your CloudWatch logs

☐ Create a new role ☐ Choose an existing role

Logging role is only required when selecting a workflow in the Managed workflows section below.

Jika Anda memilih Buat grup log, CloudWatch konsol (<https://console.aws.amazon.com/cloudwatch/>) terbuka ke halaman Buat grup log. Untuk detailnya, lihat [Membuat grup log di CloudWatch Log](#).

Memperbarui logging untuk server

Detail untuk pencatatan bergantung pada skenario pembaruan Anda.

Note

Ketika Anda memilih masuk ke JSON logging terstruktur, mungkin ada penundaan, dalam kasus yang jarang terjadi, di mana Transfer Family berhenti masuk dalam format lama, tetapi membutuhkan waktu untuk mulai masuk dalam JSON format baru. Hal ini dapat mengakibatkan peristiwa yang tidak masuk log. Tidak akan ada gangguan layanan, tetapi Anda harus berhati-hati mentransfer file selama satu jam pertama setelah mengubah metode logging Anda, karena log dapat dihapus.

Jika Anda mengedit server yang ada, opsi Anda bergantung pada status server.

- Server sudah mengaktifkan peran logging, tetapi tidak mengaktifkan JSON Pencatatan terstruktur.

Edit additional details

Logging [Info](#)

Log group [Info](#)

Choose an existing log group from the dropdown or create a new log group in Amazon CloudWatch

☒ Enable structured JSON logging

/aws/transfer/scooter ▼



Create log group [↗](#)

Enabling the structured JSON log format will override your existing logging configuration. Potential changes include new log format and log group.

Logging Role [Info](#)

Select an existing role from your account

AWSTransferLoggingAccess ▼



Workflows events will be delivered to a log group labelled with the server ID.

- Server tidak memiliki logging yang diaktifkan.

Edit additional details

Logging [Info](#)

Log group [Info](#)

Choose an existing log group from the dropdown or create a new log group in Amazon CloudWatch

☐ Enable structured JSON logging

Choose an existing log group ▼



Create log group [↗](#)

Logging Role [Info](#)

Select an existing role from your account

Choose a role ▼



Logging role is only required when selecting a workflow in the Managed workflows section below.

- Server sudah mengaktifkan JSON Pencatatan terstruktur, tetapi tidak memiliki peran logging yang ditentukan.

Edit additional details

Logging [Info](#)

Log group [Info](#)

Choose an existing log group from the dropdown or create a new log group in Amazon CloudWatch

☒ Enable structured JSON logging

/aws/transfer/  ▼



Create log group [↗](#)

Logging Role [Info](#)

Select an existing role from your account

Choose a role ▼



Logging role is only required when selecting a workflow in the Managed workflows section below.

- Server sudah mengaktifkan JSON Pencatatan terstruktur, dan juga memiliki peran logging yang ditentukan.

Edit additional details

Logging [Info](#)

Log group [Info](#)
Choose an existing log group from the dropdown or create a new log group in Amazon CloudWatch

☒ Enable structured JSON logging

[Create log group](#)

Logging Role [Info](#)
Select an existing role from your account

[Refresh](#)

Workflows events will be delivered to a log group labelled with the server ID.

Melihat konfigurasi server

Detail untuk halaman konfigurasi server bergantung pada skenario Anda:

Bergantung pada skenario Anda, halaman konfigurasi server mungkin terlihat seperti salah satu contoh berikut:

- Tidak ada logging yang diaktifkan.

Additional details			Edit
Log group -	Domain Amazon S3	Login display banner View the display message	
Logging role Info -	Workflow for complete uploads -	SetStat option Ignore	
Server host key Info SHA256: [redacted]	Workflow for partial uploads -	TLS session resumption -	
Security Policy Info TransferSecurityPolicy-2018-11	Managed workflows execution role -	Passive IP -	

- JSONPencatatan terstruktur diaktifkan.

Additional details			Edit
Log group /aws/transfer/s[redacted]	Domain Amazon S3	Login display banner View the display message	
Logging role Info -	Workflow for complete uploads -	SetStat option Ignore	
Server host key Info SHA256: [redacted]	Workflow for partial uploads -	TLS session resumption -	
Security Policy Info TransferSecurityPolicy-2020-06	Managed workflows execution role -	Passive IP -	

- Peran logging diaktifkan, tetapi JSON logging terstruktur tidak diaktifkan.

Additional details			Edit
Log group -	Domain Amazon S3	Login display banner View the display message	
Logging role Info AWSTransferLoggingAccess	Workflow for complete uploads w-[redacted]	SetStat option Ignore	
Server host key Info SHA256:lx39/[redacted]	Workflow for partial uploads -	TLS session resumption -	
Security Policy Info TransferSecurityPolicy-2018-11	Managed workflows execution role [redacted] execution-role	Passive IP -	

- Kedua jenis logging (peran logging dan JSON logging terstruktur) diaktifkan.

Additional details

Edit

Log group /aws/transfer/s-	Domain Amazon S3	Login display banner View the display message
Logging role Info AWSTransferLoggingAccess	Workflow for complete uploads w-	SetStat option Ignore
Server host key Info SHA256:	Workflow for partial uploads -	TLS session resumption -
Security Policy Info TransferSecurityPolicy-2020-06	Managed workflows execution role transfer-workflows	Passive IP -

Mengelola logging untuk alur kerja

CloudWatch menyediakan audit dan pencatatan terkonsolidasi untuk kemajuan dan hasil alur kerja. Selain itu, AWS Transfer Family menyediakan beberapa metrik untuk alur kerja. Anda dapat melihat metrik berapa banyak eksekusi alur kerja yang dimulai, diselesaikan dengan sukses, dan gagal pada menit sebelumnya. Semua CloudWatch metrik untuk Transfer Family dijelaskan dalam [Menggunakan CloudWatch metrik untuk Transfer Family](#).

Lihat CloudWatch log Amazon untuk alur kerja

1. Buka CloudWatch konsol Amazon di <https://console.aws.amazon.com/cloudwatch/>.
2. Di panel navigasi kiri, pilih Log, lalu pilih Grup log.
3. Pada halaman Grup log, pada bilah navigasi, pilih Wilayah yang benar untuk AWS Transfer Family server Anda.
4. Pilih grup log yang sesuai dengan server Anda.

Misalnya, jika ID server Anda `s-1234567890abcdef0`, grup log Anda adalah `/aws/transfer/s-1234567890abcdef0`.

5. Pada halaman detail grup log untuk server Anda, aliran log terbaru ditampilkan. Ada dua aliran log untuk pengguna yang Anda jelajahi:

- Satu untuk setiap sesi Secure Shell (SSH) File Transfer Protocol (SFTP).

- Satu untuk alur kerja yang sedang dijalankan untuk server Anda. Format untuk aliran log untuk alur kerja adalah `username.workflowID.uniqueStreamSuffix`.

Misalnya, jika pengguna Andamary-major, Anda memiliki aliran log berikut:

```
mary-major-east.1234567890abcdef0
mary.w-abcdef01234567890.021345abcdef6789
```

Note

Pengidentifikasi alfanumerik 16 digit yang tercantum dalam contoh ini adalah fiktif. Nilai yang Anda lihat di Amazon CloudWatch berbeda.

Halaman peristiwa Log untuk `mary-major-usa-east.1234567890abcdef0` menampilkan detail untuk setiap sesi pengguna, dan aliran `mary.w-abcdef01234567890.021345abcdef6789` log berisi detail untuk alur kerja.

Berikut ini adalah contoh aliran log untuk `mary.w-abcdef01234567890.021345abcdef6789`, berdasarkan alur kerja (`w-abcdef01234567890`) yang berisi langkah salin.

```
{
  "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"
}
```

Konfigurasi peran CloudWatch logging

Untuk menetapkan akses, Anda membuat IAM kebijakan berbasis sumber daya dan IAM peran yang menyediakan informasi akses tersebut.

Untuk mengaktifkan CloudWatch pencatatan Amazon, Anda mulai dengan membuat IAM kebijakan yang memungkinkan CloudWatch pencatatan log. Anda kemudian membuat IAM peran dan melampirkan kebijakan padanya. Anda dapat melakukan ini ketika Anda [membuat server](#) atau dengan [mengedit server yang ada](#). Untuk informasi selengkapnya CloudWatch, lihat [Apa itu Amazon CloudWatch?](#) dan [Apa itu CloudWatch log Amazon?](#) di Panduan CloudWatch Pengguna Amazon.

Gunakan contoh IAM kebijakan berikut untuk mengizinkan CloudWatch pencatatan.

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:Akun AWS:log-group:/aws/transfer/*"
  }
]
}

```

Dalam contoh kebijakan sebelumnya, untuk **Resource**, ganti *region-id* and *Akun AWS* dengan nilai-nilai Anda. Sebagai contoh, **"Resource": "arn:aws::logs:us-east-1:11112223333:log-group:/aws/transfer/*"**.

Anda kemudian membuat peran dan melampirkan kebijakan CloudWatch Log yang Anda buat.

Untuk membuat IAM peran dan melampirkan kebijakan

1. Di panel navigasi, pilih Peran, lalu pilih Buat peran.

Pada halaman Buat peran, pastikan bahwa AWS layanan dipilih.

2. Pilih Transfer dari daftar layanan, lalu pilih Berikutnya: Izin. Ini membangun hubungan kepercayaan antara AWS Transfer Family dan IAM peran. Selain itu, tambahkan `aws:SourceAccount` dan `aws:SourceArn` kondisikan kunci untuk melindungi diri Anda dari masalah wakil yang membingungkan. Lihat dokumentasi berikut untuk lebih jelasnya:

- Prosedur untuk membangun hubungan kepercayaan dengan AWS Transfer Family: [Untuk membangun hubungan kepercayaan](#)
- Deskripsi untuk masalah wakil yang bingung: [masalah wakil yang bingung](#)

3. Di bagian Lampirkan kebijakan izin, cari dan pilih kebijakan CloudWatch Log yang baru saja Anda buat, lalu pilih Berikutnya: Tag.
4. (Opsional) Masukkan kunci dan nilai untuk tag, dan pilih Berikutnya: Tinjau.
5. Pada halaman Tinjauan, masukkan nama dan deskripsi untuk peran baru Anda, lalu pilih Buat peran.
6. Untuk melihat log, pilih ID Server untuk membuka halaman konfigurasi server, dan pilih Lihat log. Anda diarahkan ke CloudWatch konsol tempat Anda dapat melihat aliran log Anda.

Pada CloudWatch halaman server Anda, Anda dapat melihat catatan otentikasi pengguna (keberhasilan dan kegagalan), unggahan data (PUToperasi), dan unduhan data (GEToperasi).

Melihat aliran log Transfer Family

Untuk melihat log server Transfer Family

1. Arahkan ke halaman detail untuk server.
2. Pilih Lihat log. Ini membuka Amazon CloudWatch.
3. Grup log untuk server yang Anda pilih akan ditampilkan.

CloudWatch ×

CloudWatch > Log groups > /aws/transfer/s-

/aws/transfer/s- Actions View in Logs Insights Search log group

Log group details

ARN arn:aws:logs:us-east-2:5:log-group:/aws/transfer/s-:*	Metric filters 0 Subscription filters 0 Contributor Insights rules -	Data protection - new Inactive Sensitive data found - new - KMS key ID -
Creation time 2 years ago		
Retention Never expire		
Stored bytes 39.39 MB		

Log streams Metric filters Subscription filters Contributor Insights Tags Data protection - new

Log streams (10) [Refresh] [Delete] [Create log stream] [Search all log streams]

Q Filter log streams or try prefix search ☐ Exact match ☐ Show expired [Info](#) < 1 > [Settings]

☐ Log stream	Last event
☐ ERRORS	2023-
☐ scooterstack4-	2023-
☐ scooterstack4-	2023-
☐ scooterstack4-	2023-

4. Anda dapat memilih aliran log untuk menampilkan detail dan entri individual untuk aliran.

- Jika ada daftar untuk ERRORS, Anda dapat memilihnya untuk melihat detail kesalahan terbaru untuk server.

CloudWatch > Log groups > /aws/transfer/s- > ERRORS

Log events

You can use the filter bar below to search for and match terms, phrases, or values in your log events. [Learn more about filter patterns](#)

↻

Actions ▾

Create metric filter

Q

Filter events

Clear

1m

30m

1h

12h

Custom

Display ▾

▶	Timestamp	Message	
There are older events to load. Load more.			
▶	2023-03-23T16:08:29.281-04:00	ERRORS AUTH_FAILURE Method=password User=ubuntu Message= SourceIP=	
▶	2023-03-23T16:08:30.979-04:00	ERRORS AUTH_FAILURE Method=password User=ubuntu Message= SourceIP=	
▶	2023-03-23T16:08:32.647-04:00	ERRORS AUTH_FAILURE Method=password User=ubuntu Message= SourceIP=	
▶	2023-03-23T16:08:34.306-04:00	ERRORS AUTH_FAILURE Method=password User=ubuntu Message= SourceIP=	
▶	2023-03-23T16:08:36.010-04:00	ERRORS AUTH_FAILURE Method=password User=ubuntu Message= SourceIP=	
▶	2023-03-23T16:08:37.659-04:00	ERRORS AUTH_FAILURE Method=password User=ubuntu Message= SourceIP=	
▶	2023-03-23T16:12:33.307-04:00	ERRORS AUTH_FAILURE Method=password User=scooterstack4 Message="Missing POSIX profile" Source...	
▼	2023-03-23T16:12:34.943-04:00	ERRORS AUTH_FAILURE Method=password User=scooterstack4 Message="Missing POSIX profile" Source...	
		ERRORS AUTH_FAILURE Method=password User=scooterstack4 Message="Missing POSIX profile" SourceIP=	Copy
▼	2023-03-23T16:12:56.857-04:00	ERRORS AUTH_FAILURE Method=password User=debian Message= SourceIP=	
		ERRORS AUTH_FAILURE Method=password User=debian Message= SourceIP=	Copy
▼	2023-03-23T16:12:58.430-04:00	ERRORS AUTH_FAILURE Method=password User=debian Message= SourceIP=	
		ERRORS AUTH_FAILURE Method=password User=debian Message= SourceIP=	Copy
▶	2023-03-23T16:13:00.106-04:00	ERRORS AUTH_FAILURE Method=password User=debian Message= SourceIP=	

- Pilih entri lain untuk melihat contoh aliran log.

CloudWatch > Log groups > /aws/transfer/s- > scooterstack4.

Log events

You can use the filter bar below to search for and match terms, phrases, or values in your log events. [Learn more about filter patterns](#)

↻

Actions ▾

Create metric filter

Q

Filter events

Clear

1m

30m

1h

12h

Custom

Display ▾

▶	Timestamp	Message	
No older events at this moment. Retry			
▼	2023-03-23T16:19:43.747-04:00	scooterstack4. CONNECTED SourceIP= User=scooterstack4 HomeDir=/fs- scooterstack4. CONNECTED SourceIP= User=scooterstack4 HomeDir=/fs- Client=SSH-2.0- OpenSSH_7.4 Role=arn:aws:iam:: :role/ Kex=	Copy
▼	2023-03-23T16:19:47.030-04:00	scooterstack4. DISCONNECTED scooterstack4. DISCONNECTED	Copy
No newer events at this moment. Auto retry paused. Resume			

Melihat aliran log Transfer Family

424

- Jika server Anda memiliki alur kerja terkelola yang terkait dengannya, Anda dapat melihat log untuk menjalankan alur kerja.

Note

Format untuk aliran log untuk alur kerja adalah `username.workflowId.uniqueStreamSuffix`. Misalnya, `decrypt-user.w-a1111222233334444.aaaa1111bbbb2222` bisa menjadi nama aliran log untuk pengguna dan alur kerja. **decrypt-user w-a1111222233334444**

CloudWatch > Log groups > /aws/transfer/s- > decrypt-user.w-

Log events

You can use the filter bar below to search for and match terms, phrases, or values in your log events. [Learn more about filter patterns](#)

Actions Create metric filter

Filter events Clear 1m 30m 1h 12h Custom Display

Timestamp	Message
	There are older events to load. Load more .
2023-03-21T13:37:57.795-04:00	<code>{"type": "StepStarted", "details": {"input": {"fileLocation": {"backingStore": "S3", "bucket": "", "key": "decrypt-</code>
2023-03-21T14:12:02.850-04:00	<code>{"type": "StepStarted", "details": {"input": {"fileLocation": {"backingStore": "S3", "bucket": "", "key": "decrypt- <pre>{ "type": "StepStarted", "details": { "input": { "fileLocation": { "backingStore": "S3", "bucket": "", "key": "decrypt-user/test.json.gpg", "versionId": "", "etag": "" } }, "stepType": "DECRYPT", "stepName": "decrypt-step" }, "workflowId": "w-", "executionId": "", "transferDetails": { "serverId": "s-", "username": "decrypt-user", "sessionId": "" } }</pre></code>
2023-03-21T14:12:03.464-04:00	<code>{"type": "StepCompleted", "details": {"output": {}}, "stepType": "DECRYPT", "stepName": "decrypt-step", "workflowId": "w-</code>

Note

Untuk entri log yang diperluas, Anda dapat menyalin entri ke clipboard dengan memilih Salin. Untuk detail selengkapnya tentang CloudWatch log, lihat [Melihat data log](#).

Membuat CloudWatch alarm Amazon

Contoh berikut menunjukkan cara membuat CloudWatch alarm Amazon menggunakan AWS Transfer Family metrik,FilesIn.

CDK

```
new cloudwatch.Metric({
  namespace: "AWS/Transfer",
  metricName: "FilesIn",
  dimensionsMap: { ServerId: "s-000000000000000000" },
  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-000000000000000000
  Statistic: Average
  Period: 60
  Threshold: 1000
  EvaluationPeriods: 10
  DatapointsToAlarm: 5
  ComparisonOperator: GreaterThanOrEqualToThreshold
```

Mencatat API panggilan Amazon S3 ke log akses S3

Jika Anda [menggunakan log akses Amazon S3 untuk mengidentifikasi permintaan S3](#) yang dibuat atas nama pengguna transfer file Anda, RoleSessionName digunakan untuk menampilkan IAM

peran mana yang diasumsikan untuk melayani transfer file. Ini juga menampilkan informasi tambahan seperti nama pengguna, id sesi, dan server-id yang digunakan untuk transfer. Formatnya adalah [AWS:Role Unique Identifier]/username.sessionid@server-id dan terkandung dalam bidang Pemohon. Misalnya, berikut ini adalah konten untuk bidang Peminta sampel dari log akses S3 untuk file yang disalin ke bucket S3.

```
arn:aws:sts::AWS-Account-ID:assumed-role/IamRoleName/  
username.sessionid@server-id
```

Di bidang Pemohon di atas, ini menunjukkan IAM Peran yang disebut `IamRoleName`. Untuk informasi selengkapnya tentang pengidentifikasi unik IAM peran, lihat [Pengidentifikasi unik](#) di AWS Identity and Access Management Panduan Pengguna.

Contoh untuk membatasi masalah wakil yang membingungkan

Masalah deputy yang bingung adalah masalah keamanan di mana entitas yang tidak memiliki izin untuk melakukan tindakan dapat memaksa entitas yang lebih istimewa untuk melakukan tindakan. Pada tahun AWS, peniruan lintas layanan dapat mengakibatkan masalah wakil yang membingungkan. Untuk detail selengkapnya, lihat [Pencegahan confused deputy lintas layanan](#).

Note

Dalam contoh berikut, ganti masing-masing *user input placeholder* dengan informasi Anda sendiri.

Dalam contoh ini, Anda dapat menghapus ARN detail alur kerja jika server Anda tidak memiliki alur kerja yang melekat padanya.

Contoh kebijakan logging/pemanggilan berikut memungkinkan setiap server (dan alur kerja) di akun untuk mengambil peran.

```
{  
  "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/*"
        ]
      }
    }
  }
]
}

```

Contoh kebijakan logging/pemanggilan berikut memungkinkan server tertentu (dan alur kerja) untuk mengambil peran.

```

{
  "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 struktur log untuk Transfer Family

Topik ini menjelaskan bidang yang diisi dalam log Transfer Family: baik untuk entri log terstruktur JSON maupun entri log lama.

Topik

- [Log terstruktur JSON untuk Transfer Family](#)
- [Log lama untuk Transfer Family](#)

Log terstruktur JSON untuk Transfer Family

Tabel berikut berisi rincian untuk bidang entri log untuk tindakan Transfer Family SFTP/FTP/FTPS, dalam format log terstruktur JSON yang baru.

Bidang	Deskripsi	Entri contoh
activity-type	The action by the user	BUKA TUTUP PARTIAL_CLOSE TERPUTUS TERHUBUNG
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 Algoritma kriptografi)	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 to the endpoint if their home directory type is PATH: if they	/user-home-bucket/test

Bidang	Deskripsi	Entri contoh
	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 Algoritma kriptografi)	diffie-hellman-group14-sha256
message	Provides more information related to the error	<string>
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-timur laut-1:12346789012: server/s-1234567890akeu2js2
role	The IAM role of the user	arn:aws:iam: :0293883675: peran/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

Bidang	Deskripsi	Entri contoh
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

Log lama untuk Transfer Family

Tabel berikut berisi rincian untuk entri log untuk berbagai tindakan Transfer Family.

Note

Entri ini tidak dalam format log terstruktur JSON yang baru.

Tabel berikut berisi rincian untuk entri log untuk berbagai tindakan Transfer Family, dalam format log terstruktur JSON yang baru.

Tindakan	Log yang sesuai dalam CloudWatch Log Amazon
Kegagalan otentikasi	KESALAHAN AUTH_FAILURE METHOD=PublicKey User="RSA SHA256:lfz3r2nmly4rak+b7rb1rsvuibae+a+hxg0c7l1jiZ0" sourceIP=3.8.172.211
SALIN/TANDA/HAPUS/DEKRIPSI alur kerja	<pre>{ "type": "StepStarted", "details": { "input": { "FileLocation": { "backingStore": "EFS", "fileSystemId": "fs-12345678", "path": "/lhr/regex.py" }, "stepType": "TAG", "stepName": "successful_tag_step" }, "workflowid": "workFlowid", "successful_tag_step": { "workflowid": "workFlowid", "stepName": "successful_tag_step" }, "workflowid": "workFlowid", "successful_tag_step": { "workflowid": "workFlowid", "stepName": "successful_tag_step" } } }</pre>

Tindakan	Log yang sesuai dalam CloudWatch Log Amazon
	<pre>{ "workFlowid": "workFlowid", "stepName": "w-1111aaaa22bbbb3", "ExecutionID": "81234abcd-1234-efgh-5678-ijklmnopqr90", "transferDetails": { "serverID": "s-1234abcd5678efghi", "nama pengguna": "lhr", "sessionID": "123456767890abcdef0" } }</pre>
Alur kerja langkah kustom	<pre>{ "type": "CustomStepInvoked", "details": { "output": { "token": "MZM4MjG5YWUTYT EzMy 00 Yjlz LWI3OG MtYz U4OGI2 ZjQyMz E5" }, "StepType": "CUSTOM", "stepName": "efs-s3_copy_2", "workFlowid": "w-9283e49d33297c3f7", "executionID": "1234abcd-1234-efgh-5678-ijklmnopqr90", "transferDetails": { "serverID": "s-zzzzzz11aaaa22223", "username": "lhr", "sessionID": "1234567890abcdef0" } } }</pre>
Menghapus	lhr.33a8fb495ffb383b HAPUS PATH=/ember/pengguna/123.jpg
Unduh	lhr.33a8fb495ffb383b JALUR TERBUKA =/ember/pengguna/mode 123.jpg =Baca lhr.33a8fb495ffb383b JALUR TUTUP =/ember/pengguna/123.jpg = 3618546 BytesOut
Login/Logout	user.914984e553bcddb6 SUMBER TERHUBUNG = 1.22.111.222 pengguna = LHR = KLIEN LOGIS = SSH-2.0-openssh_7.4 peran = arn:aws:iam::123456789012: peran/sftp-s3-akses HomeDir user.914984e553bcddb6 TERPUTUS

Tindakan	Log yang sesuai dalam CloudWatch Log Amazon
Mengganti nama	lhr.33a8fb495ffb383b GANTI NAMA PATH=/bucket/user/lambo.png =/ember/pengguna/ferrari.png NewPath
Contoh log kesalahan alur kerja	{"type": "StepErrored ", "details": {"ErrorType": "BAD_REQUEST", "ErrorMessage": "Tidak dapat menandai berkas Efs", "stepType": "stepName": "successful_tag_step"}, "workFlowid": "w-1234abcd5678efghi", "ExecutionID": "81234abcd5678efghi", "ExecutionID": "81234abcd5678efghi": "ExecutionID": "81234abcd5678efghi": "ExecutionID": "81234abcd5678efghi": "ExecutionID": "81234abcd534abcd-1234-efgh-5678-ijklmnopqr90", "transferDetails": {"serverID": "s-1234abcd5678efghi", "nama pengguna": "lhr", "sessionID": "1234567890abcdef0"}}}
Symlink	lhr.eb49cf7b8651e6d5 CREATE_SYMLINK =/fs-12345678/lhr/pqr.jpg =abc.jpg LinkPath TargetPath
Unggahan	lhr.33a8fb495ffb383b JALUR TERBUKA =/bucket/pengguna/mode 123.jpg =Buat memotong tulisi lhr.33a8fb495ffb383b JALUR TUTUP =/ember/pengguna/123.jpg = 3618546 BytesIn

Tindakan	Log yang sesuai dalam CloudWatch Log Amazon
Alur Kerja	<pre>{ "type": "ExecutionStarted", "details": { "input": { "backingStore": "EFS", "fileSystemId": "fs-12345678", "path": "/lhr/regex.py", "initialFileLocation": {} }, "workflowId": "w-1111aaa222bbb3", "executionId": "1234abcd-1234efgh-5678-ijklmnopqr90", "transferDetails": { "serverId": "s-zzzz1111aaaa22223", "namaPengguna": "lhr", "sessionId": "1234567890abcdef0" } } }</pre> <pre>{ "type": "StepStarted", "details": { "input": { "FileLocation": { "backingStore": "EFS", "fileSystemId": "fs-12345678", "path": "/lhr/regex.py" }, "stepType": "CUSTOM", "stepName": "efs-s3_copy_2" }, "workflowId": "w-9283e49d33297c3f7", "executionId": "1234abcd-1234efgh-5678-ijklmnopqr90", "transferDetails": { "serverId": "s-18ca49dce5d842e0b", "username": "lhr", "sessionId": "1234567890abcdef0" } } }</pre>

Contoh entri CloudWatch log

Topik ini menyajikan contoh entri log.

Topik

- [Contoh entri log sesi transfer](#)
- [Contoh entri log untuk konektor SFTP](#)
- [Contoh entri log untuk kegagalan algoritma pertukaran Kunci](#)

Contoh entri log sesi transfer

Dalam contoh ini, SFTP pengguna terhubung ke server Transfer Family, mengunggah file, lalu memutuskan sambungan dari sesi.

Entri log berikut mencerminkan SFTP pengguna yang terhubung ke server Transfer Family.

```
{
  "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"
}
```

Entri log berikut mencerminkan SFTP pengguna yang mengunggah file ke bucket Amazon S3 mereka.

```
{
  "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"
}
```

Entri log berikut mencerminkan SFTP pengguna memutuskan sambungan dari sesi mereka SFTP. Pertama, klien menutup koneksi ke bucket, dan kemudian klien memutuskan sesi. SFTP

```
{
  "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"
  }

```

Contoh entri log untuk konektor SFTP

Bagian ini berisi contoh log untuk transfer yang berhasil dan tidak berhasil. Log dihasilkan ke grup log bernama `aws/transfer/connector-id`, di mana *connector-id* adalah pengenal untuk SFTP konektor Anda.

Note

Entri log untuk SFTP konektor hanya dihasilkan saat Anda menjalankan `StartFileTransfer` perintah.

Entri log ini untuk transfer yang berhasil diselesaikan.

```

{
  "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"
}

```

Entri log ini untuk transfer yang waktunya habis, dan dengan demikian tidak berhasil diselesaikan.

```
{
  "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"
}
```

Deskripsi untuk beberapa bidang kunci dalam contoh log sebelumnya.

- `timestamp` mewakili saat log ditambahkan ke CloudWatch. `start-time` dan `end-time` sesuai dengan kapan konektor benar-benar memulai dan menyelesaikan transfer.
- `transfer-id` adalah pengidentifikasi unik yang ditetapkan untuk setiap `start-file-transfer` permintaan. Jika pengguna melewati beberapa jalur file dalam satu `start-file-transfer` API panggilan, semua file berbagi yang sama `transfer-id`.
- `file-transfer-id` adalah nilai unik yang dihasilkan untuk setiap file yang ditransfer. Perhatikan bahwa bagian awal `file-transfer-id` adalah sama dengan `transfer-id`.

Contoh entri log untuk kegagalan algoritma pertukaran Kunci

Bagian ini berisi contoh log di mana algoritma pertukaran Kunci (KEX) gagal. Ini adalah contoh dari aliran `ERRORS` log untuk log terstruktur.

Entri log ini adalah contoh di mana ada kesalahan jenis kunci host.

```
{
  "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"
}
```

Entri log ini adalah contoh di mana ada KEX ketidakcocokan.

```
{
  "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"
}
```

Menggunakan CloudWatch metrik untuk Transfer Family

Note

Anda juga bisa mendapatkan metrik untuk Transfer Family dari dalam konsol Transfer Family itu sendiri. Untuk detailnya, lihat [Memantau penggunaan di konsol](#)

Anda bisa mendapatkan informasi tentang server Anda menggunakan CloudWatch metrik. Metrik mewakili kumpulan titik data yang diurutkan waktu yang dipublikasikan ke CloudWatch. [Saat menggunakan metrik, Anda harus menentukan namespace Transfer Family, nama metrik, dan dimensi.](#) Untuk informasi selengkapnya tentang metrik, lihat [Metrik](#) di CloudWatch Panduan Pengguna Amazon.

Tabel berikut menjelaskan CloudWatch metrik untuk Transfer Family.

Namespace	Metrik	Deskripsi
AWS/Transfer	BytesIn	Jumlah total byte yang ditransfer ke server. Unit: Jumlah Periode: 5 menit

Namespace	Metrik	Deskripsi
	BytesOut	Jumlah total byte yang ditransfer keluar dari server. Unit: Jumlah Periode: 5 menit
	FilesIn	Jumlah total file yang ditransfer ke server. Untuk server yang menggunakan AS2 protokol, metrik ini mewakili jumlah pesan yang diterima. Unit: Jumlah Periode: 5 menit
	FilesOut	Jumlah total file yang ditransfer keluar dari server. Unit: Jumlah Periode: 5 menit
	InboundMessage	Jumlah total AS2 pesan yang berhasil diterima dari mitra dagang. Unit: Jumlah Periode: 5 menit
	InboundFailedMessage	Jumlah total AS2 pesan yang tidak berhasil diterima dari mitra dagang. Artinya, mitra dagang mengirim pesan, tetapi server Transfer Family tidak berhasil memprosesnya. Unit: Jumlah Periode: 5 menit

Namespace	Metrik	Deskripsi
	OnPartialUploadExecutionsStarted	Jumlah total eksekusi on-partial-upload alur kerja dimulai di server. Unit: Jumlah Periode: 1 menit
	OnPartialUploadExecutionsSuccess	Jumlah total eksekusi on-partial-upload alur kerja yang berhasil di server. Unit: Jumlah Periode: 1 menit
	OnPartialUploadExecutionsFailed	Jumlah total eksekusi on-partial-upload alur kerja yang gagal di server. Unit: Jumlah Periode: 1 menit
	OnUploadExecutionsStarted	Jumlah total eksekusi alur kerja dimulai di server. Unit: Jumlah Periode: 1 menit
	OnUploadExecutionsSuccess	Jumlah total eksekusi alur kerja yang berhasil di server. Unit: Jumlah Periode: 1 menit
	OnUploadExecutionsFailed	Jumlah total eksekusi alur kerja yang gagal di server. Unit: Jumlah Periode: 1 menit

Dimensi Transfer Family

Dimensi adalah pasangan nama/nilai yang merupakan bagian dari identitas metrik. Untuk informasi selengkapnya tentang dimensi, lihat [Dimensi](#) di Panduan CloudWatch Pengguna Amazon.

Tabel berikut menjelaskan CloudWatch dimensi Transfer Family.

Dimensi	Deskripsi
ServerId	ID unik dari server.

Menggunakan Notifikasi Pengguna AWS dengan AWS Transfer Family

Untuk mendapatkan pemberitahuan tentang AWS Transfer Family acara, Anda dapat menggunakan [Notifikasi Pengguna AWS](#) untuk mengatur berbagai saluran pengiriman. Jika acara cocok dengan aturan yang Anda tentukan, Anda menerima pemberitahuan.

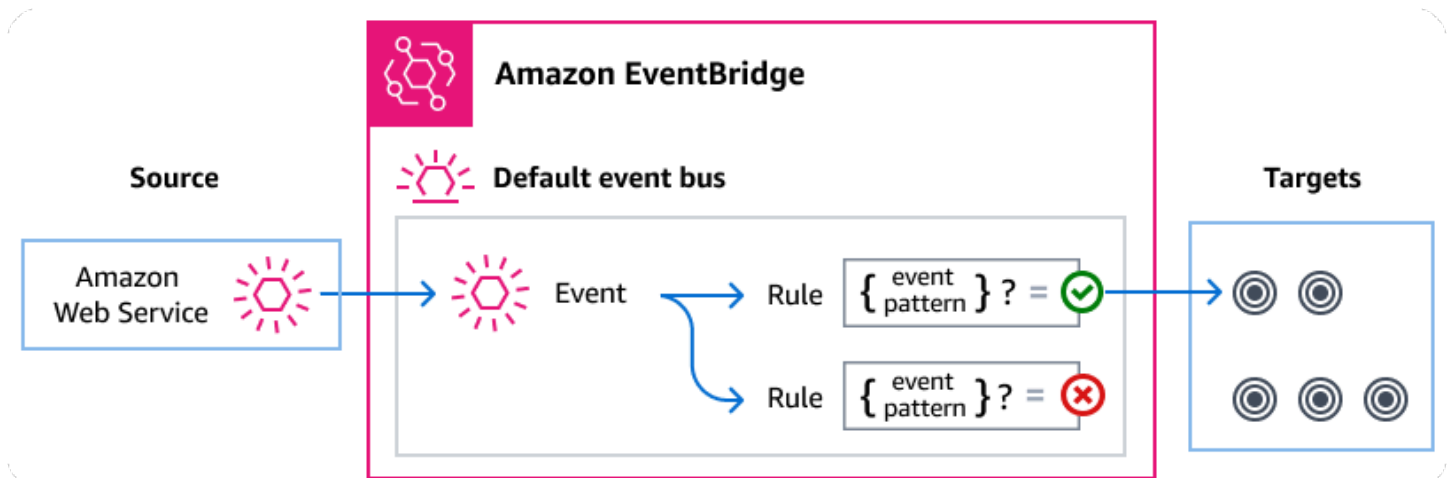
Anda dapat menerima notifikasi untuk peristiwa melalui beberapa saluran, termasuk email, notifikasi obrolan [Amazon Q Developer dalam aplikasi obrolan](#), atau notifikasi push [AWS Console Mobile Application](#). Anda juga dapat melihat notifikasi di [Pusat Pemberitahuan Konsol](#). Notifikasi Pengguna mendukung agregasi, yang dapat mengurangi jumlah pemberitahuan yang Anda terima selama acara tertentu.

Untuk informasi selengkapnya, lihat [Menyesuaikan pemberitahuan pengiriman file menggunakan posting blog alur kerja AWS Transfer Family terkelola](#), dan [Apa itu? Notifikasi Pengguna AWS](#) dalam Notifikasi Pengguna AWS User Guide.

Mengelola Transfer Family acara menggunakan Amazon EventBridge

Amazon EventBridge adalah layanan tanpa server yang menggunakan peristiwa untuk menghubungkan komponen aplikasi bersama-sama, yang dapat memudahkan Anda membangun aplikasi berbasis peristiwa yang dapat diskalakan. Arsitektur berbasis peristiwa adalah gaya membangun sistem perangkat lunak yang digabungkan secara longgar yang bekerja sama dengan memancarkan dan menanggapi peristiwa. Peristiwa mewakili perubahan dalam sumber daya atau lingkungan.

Seperti banyak AWS layanan, Transfer Family menghasilkan dan mengirim acara ke bus acara EventBridge default. Perhatikan bahwa bus acara default secara otomatis disediakan di setiap AWS akun. Bus acara adalah router yang menerima acara dan mengirimkannya ke nol atau lebih tujuan, atau target. Anda menentukan aturan untuk bus acara yang mengevaluasi peristiwa saat mereka tiba. Setiap aturan memeriksa apakah suatu peristiwa cocok dengan pola acara aturan. Jika acara cocok, bus acara mengirimkan acara ke satu atau lebih target yang ditentukan.



Topik

- [Transfer Family acara](#)
- [Mengirim Transfer Family acara dengan menggunakan EventBridge aturan](#)
- [Amazon EventBridge izin](#)
- [EventBridge Sumber daya tambahan](#)
- [Transfer Family referensi detail acara](#)

Transfer Family acara

Transfer Family secara otomatis mengirimkan acara ke bus EventBridge acara default. Anda dapat membuat aturan pada bus acara di mana setiap aturan mencakup pola acara dan satu atau lebih target. Peristiwa yang cocok dengan pola acara aturan dikirimkan ke target yang ditentukan [berdasarkan upaya terbaik](#), namun, beberapa peristiwa mungkin dikirimkan secara tidak berurutan.

Peristiwa berikut dihasilkan oleh Transfer Family. Untuk informasi selengkapnya, lihat [EventBridge peristiwa](#) di Panduan Amazon EventBridge Pengguna.

SFTP,FTPS, dan peristiwa FTP server

Jenis detail acara	Deskripsi
FTPUnduhan File Server Selesai	File telah berhasil diunduh untuk FTP protokol.
FTPUnduhan Server File Gagal	Upaya untuk mengunduh file gagal untuk FTP protokol.
FTPUnggahan Server File Selesai	Sebuah file telah berhasil diunggah untuk FTP protokol.
FTPUnggah Server File Gagal	Upaya untuk mengunggah file gagal untuk FTP protokol.
FTPSUnduhan File Server Selesai	File telah berhasil diunduh untuk FTPS protokol.
FTPSUnduhan Server File Gagal	Upaya untuk mengunduh file gagal untuk FTPS protokol.
FTPSUnggahan Server File Selesai	Sebuah file telah berhasil diunggah untuk FTPS protokol.
FTPSUnggah Server File Gagal	Upaya untuk mengunggah file gagal untuk FTPS protokol.
SFTPUnduhan File Server Selesai	File telah berhasil diunduh untuk SFTP protokol.

Jenis detail acara	Deskripsi
<u>SFTPUnduhan File Server Gagal</u>	Upaya untuk mengunduh file gagal untuk SFTP protokol.
<u>SFTPUnggahan File Server Selesai</u>	Sebuah file telah berhasil diunggah untuk SFTP protokol.
<u>SFTPUnggahan File Server Gagal</u>	Upaya untuk mengunggah file gagal untuk SFTP protokol.

SFTPacara konektor

Jenis detail acara	Deskripsi
<u>SFTPKirim File Konektor Selesai</u>	Transfer file dari konektor ke SFTP server jarak jauh telah berhasil diselesaikan.
<u>SFTPKirim File Konektor Gagal</u>	Transfer file dari konektor ke SFTP server jarak jauh gagal.
<u>SFTPPengambilan File Konektor Selesai</u>	Transfer file dari SFTP server jarak jauh ke konektor telah berhasil diselesaikan.
<u>SFTPPengambilan File Konektor Gagal</u>	Transfer file dari SFTP server jarak jauh ke konektor gagal.

Acara A2S

Jenis detail acara	Deskripsi
<u>AS2Payload Terima Selesai</u>	Muatan untuk AS2 pesan telah diterima.
<u>AS2Penerimaan Muatan Gagal</u>	Muatan untuk AS2 pesan belum diterima.
<u>AS2Muatan Kirim Selesai</u>	Payload untuk AS2 pesan telah berhasil dikirim.

Jenis detail acara	Deskripsi
AS2Kirim Muatan Gagal	Muatan untuk AS2 pesan gagal dikirim.
AS2MDNTerima Selesai	Pemberitahuan disposisi pesan untuk AS2 pesan telah diterima.
AS2MDNMenerima Gagal	Pemberitahuan disposisi AS2 pesan untuk pesan belum diterima.
AS2MDNKirim Selesai	Pemberitahuan disposisi pesan untuk AS2 pesan telah berhasil dikirim.
AS2MDNKirim Gagal	Pemberitahuan disposisi AS2 pesan untuk pesan gagal dikirim.

Mengirim Transfer Family acara dengan menggunakan EventBridge aturan

Jika Anda ingin bus acara EventBridge default mengirim Transfer Family acara ke target, Anda harus membuat aturan yang berisi pola peristiwa yang cocok dengan data dalam Transfer Family acara yang Anda inginkan.

Anda dapat membuat aturan dengan mengikuti langkah-langkah umum ini:

1. Buat pola acara untuk aturan yang menentukan berikut:
 - Transfer Family adalah sumber peristiwa yang dievaluasi oleh aturan.
 - (Opsional) Setiap data acara lain untuk mencocokkannya.

Untuk informasi selengkapnya, lihat [????](#).

2. (Opsional) Buat transformator input yang menyesuaikan data dari peristiwa sebelum EventBridge mengirim informasi ke target aturan.

Untuk informasi selengkapnya, lihat [Transformasi input](#) di Panduan EventBridge Pengguna.

3. Tentukan target yang EventBridge ingin Anda sampaikan acara yang cocok dengan pola acara.

Target dapat berupa AWS layanan lain, perangkat lunak sebagai layanan (SaaS) aplikasi, API tujuan, atau titik akhir kustom lainnya. Untuk informasi lebih lanjut, lihat [Target](#) di Panduan Pengguna EventBridge .

Untuk petunjuk komprehensif tentang cara membuat aturan bus acara, lihat [Membuat aturan yang bereaksi terhadap peristiwa](#) di Panduan EventBridge Pengguna.

Membuat pola acara untuk Transfer Family acara

Saat Transfer Family mengirimkan acara ke bus acara default, EventBridge gunakan pola acara yang ditentukan untuk setiap aturan untuk menentukan apakah acara harus dikirim ke target aturan. Pola peristiwa cocok dengan data dalam Transfer Family peristiwa yang diinginkan. Setiap pola peristiwa adalah JSON objek yang berisi berikut:

- `sourceAtribut` yang mengidentifikasi layanan yang mengirim acara. Untuk Transfer Family acara, sumbernya adalah `aws.transfer`.
- (Opsional) `detail-type` Atribut yang berisi array jenis acara yang cocok.
- (Opsional) `detail` Atribut yang berisi data peristiwa lain yang cocok.

Misalnya, pola acara berikut cocok dengan semua peristiwa dari Transfer Family:

```
{
  "source": ["aws.transfer"]
}
```

Contoh pola peristiwa berikut cocok dengan semua peristiwa SFTP konektor:

```
{
  "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"]
}
```

Contoh pola acara berikut cocok dengan semua peristiwa gagal Transfer Family:

```
{
  "source": ["aws.transfer"],
  "detail-type": [{"wildcard", "*Failed"}]
}
```

Contoh pola acara berikut cocok dengan SFTP unduhan yang berhasil untuk pengguna *username*:


```
{
  "source": ["aws.transfer"],
  "detail-type": ["SFTP Server File Download Completed"],
  "detail": {
    "username": [username]
  }
}
```

Untuk informasi selengkapnya tentang penulisan pola acara, lihat [Pola acara](#) di Panduan EventBridge Pengguna.

Menguji pola acara untuk Transfer Family acara di EventBridge

Anda dapat menggunakan EventBridge Sandbox untuk mendefinisikan dan menguji pola peristiwa dengan cepat, tanpa harus menyelesaikan proses pembuatan atau pengeditan aturan yang lebih luas. Menggunakan Sandbox, Anda dapat menentukan pola peristiwa dan menggunakan contoh peristiwa untuk mengonfirmasi bahwa pola tersebut cocok dengan peristiwa yang diinginkan. EventBridge memberi Anda pilihan untuk membuat aturan baru dengan menggunakan pola acara itu langsung dari kotak pasir.

Untuk informasi selengkapnya, lihat [Menguji pola peristiwa menggunakan EventBridge Kotak Pasir](#) di Panduan EventBridge Pengguna.

Amazon EventBridge izin

Transfer Family tidak memerlukan izin tambahan untuk mengirimkan acara ke Amazon EventBridge.

Target yang Anda tentukan mungkin memerlukan izin atau konfigurasi tertentu. Untuk detail selengkapnya tentang penggunaan layanan khusus untuk target, lihat [Amazon EventBridge target](#) di Panduan Amazon EventBridge Pengguna.

EventBridge Sumber daya tambahan

Lihat topik-topik berikut di [Panduan Amazon EventBridge Pengguna](#) untuk informasi lebih lanjut tentang cara menggunakan EventBridge untuk memproses dan mengelola acara.

- Untuk informasi rinci tentang cara kerja bus acara, lihat [bus Amazon EventBridge acara](#).
- Untuk informasi tentang struktur acara, lihat [Acara](#).

- Untuk informasi tentang membuat pola peristiwa untuk EventBridge digunakan saat mencocokkan peristiwa dengan aturan, lihat [Pola acara](#).
- Untuk informasi tentang membuat aturan untuk menentukan EventBridge proses peristiwa, lihat [Aturan](#).
- Untuk informasi tentang cara menentukan layanan atau tujuan lain yang EventBridge mengirimkan peristiwa yang cocok, lihat [Target](#).

Transfer Family referensi detail acara

Semua peristiwa dari AWS layanan memiliki seperangkat bidang umum yang berisi metadata tentang acara tersebut. Metadata ini dapat mencakup AWS layanan yang merupakan sumber acara, waktu acara dibuat, akun dan Wilayah tempat acara berlangsung, dan lainnya. Untuk definisi bidang umum ini, lihat [Referensi struktur acara](#) di Panduan Amazon EventBridge Pengguna.

Selain itu, setiap acara memiliki detail bidang yang berisi data khusus untuk peristiwa tertentu. Referensi berikut mendefinisikan bidang detail untuk berbagai Transfer Family acara.

Saat Anda menggunakan EventBridge untuk memilih dan mengelola Transfer Family acara, pertimbangkan hal berikut:

- `sourceBidang` untuk semua acara dari Transfer Family diatur ke `aws.transfer`.
- `detail-typeBidang` menentukan jenis acara.

Misalnya, FTP File Server Download Completed.

- `detailBidang` berisi data yang spesifik untuk peristiwa tertentu.

Untuk informasi tentang membuat pola peristiwa yang memungkinkan aturan untuk mencocokkan Transfer Family peristiwa, lihat [Pola acara](#) di Panduan Amazon EventBridge Pengguna.

Untuk informasi selengkapnya tentang peristiwa dan cara EventBridge memprosesnya, lihat [Amazon EventBridge peristiwa](#) di Panduan Amazon EventBridge Pengguna.

Topik

- [SFTP,FTPS, dan peristiwa FTP server](#)
- [SFTPaacara konektor](#)
- [AS2acara](#)

SFTP,FTPS, dan peristiwa FTP server

Berikut ini adalah bidang detail untuk SFTP,FTPS, dan peristiwa FTP server:

- FTPUnduhan File Server Selesai
- FTPUnduhan Server File Gagal
- FTPUnggahan Server File Selesai
- FTPUnggah Server File Gagal
- FTPSUnduhan File Server Selesai
- FTPSUnduhan Server File Gagal
- FTPSUnggahan Server File Selesai
- FTPSUnggah Server File Gagal
- SFTPUnduhan File Server Selesai
- SFTPUnduhan File Server Gagal
- SFTPUnggahan File Server Selesai
- SFTPUnggahan File Server Gagal

detail-typeBidang source dan disertakan di bawah ini karena mengandung nilai khusus untuk Transfer Family acara. Untuk definisi bidang metadata lain yang disertakan dalam semua peristiwa, lihat [Referensi struktur acara](#) di Amazon EventBridge Panduan Pengguna.

```
{
  . . . ,
  "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

Mengidentifikasi jenis acara.

Untuk acara ini, nilainya adalah salah satu dari SFTP, FTPS, atau nama acara FTP server yang tercantum sebelumnya.

source

Mengidentifikasi layanan yang menghasilkan peristiwa. Untuk acara Transfer Family, nilai ini adalah `aws.transfer`.

detail

JSONObjek yang berisi informasi tentang acara tersebut. Layanan yang menghasilkan acara menentukan konten bidang ini.

Untuk acara ini, data meliputi yang berikut:

failure-code

Kategori mengapa transfer gagal. Nilai: `PARTIAL_UPLOAD` | `PARTIAL_DOWNLOAD` | `UNKNOWN_ERROR`

status-code

Apakah transfer berhasil. Nilai: `COMPLETED` | `FAILED`.

protocol

Protokol yang digunakan untuk transfer. Nilai: `SFTP` | `FTPS` | `FTP`

bytes

Jumlah byte yang ditransfer.

client-ip

Alamat IP untuk klien yang terlibat dalam transfer

failure-message

Untuk transfer yang gagal, detail mengapa transfer gagal.

end-timestamp

Untuk transfer yang berhasil, stempel waktu ketika file selesai diproses.

etag

Tag entitas (hanya digunakan untuk file Amazon S3).

file-path

Jalur ke file yang ditransfer.

server-id

ID unik untuk server Transfer Family.

username

Pengguna yang melakukan transfer.

session-id

Pengidentifikasi unik untuk sesi transfer.

start-timestamp

Untuk transfer yang berhasil, stempel waktu saat pemrosesan file dimulai.

Example SFTPUnduhan File Server Contoh peristiwa yang gagal

Contoh berikut menunjukkan peristiwa di mana unduhan gagal di SFTP server (Amazon EFS adalah penyimpanan yang digunakan).

```
{
  "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 FTPFile Server Upload Contoh acara selesai

Contoh berikut menunjukkan peristiwa di mana unggahan berhasil diselesaikan di FTP server (Amazon S3 adalah penyimpanan yang digunakan).

```

{
  "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"
  }
}

```

SFTPacara konektor

Berikut ini adalah bidang detail untuk acara SFTP konektor:

- SFTPKirim File Konektor Selesai
- SFTPKirim File Konektor Gagal
- SFTPPengambilan File Konektor Selesai
- SFTPPengambilan File Konektor Gagal

detail-typeBidang source dan disertakan di bawah ini karena mengandung nilai khusus untuk Transfer Family acara. Untuk definisi bidang metadata lain yang disertakan dalam semua peristiwa, lihat [Referensi struktur acara](#) di Amazon EventBridge Panduan Pengguna.

```

{
  . . . ,
  "detail-type": "string",
  "source": "aws.transfer",
  . . . ,
  "detail": {
    "operation" : "string",
    "connector-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",
    "bytes" : "number",
    "local-file-location" : {

```

```
    "domain" : "string",
    "bucket" : "string",
    "key" : "string"
  },
}
```

detail-type

Mengidentifikasi jenis acara.

Untuk acara ini, nilainya adalah salah satu nama acara SFTP konektor yang tercantum sebelumnya.

source

Mengidentifikasi layanan yang menghasilkan peristiwa. Untuk Transfer Family acara, nilai ini adalah `aws.transfer`.

detail

JSONObjek yang berisi informasi tentang acara tersebut. Layanan yang menghasilkan acara menentukan konten bidang ini.

Untuk acara ini, data meliputi yang berikut:

operation

Apakah `StartFileTransfer` permintaan mengirim atau mengambil file. Nilai: `SEND` | `RETRIEVE`.

connector-id

Pengidentifikasi unik untuk SFTP konektor yang digunakan.

transfer-id

Pengenal unik untuk acara transfer (`StartFileTransfer` permintaan).

file-transfer-id

Pengidentifikasi unik untuk file yang ditransfer.

url

URL Dari pasangan AS2 atau SFTP titik akhir.

`file-path`

Lokasi dan file yang sedang dikirim atau diambil.

`status-code`

Apakah transfer berhasil. Nilai: `FAILED` | `COMPLETED`.

`failure-code`

Untuk transfer yang gagal, kode alasan mengapa transfer gagal.

`failure-message`

Untuk transfer yang gagal, detail mengapa transfer gagal.

`start-timestamp`

Untuk transfer yang berhasil, stempel waktu saat pemrosesan file dimulai.

`end-timestamp`

Untuk transfer yang berhasil, stempel waktu saat pemrosesan file selesai.

`local-directory-path`

Untuk `RETRIEVE` permintaan, lokasi untuk menempatkan file yang diambil.

`remote-directory-path`

Untuk `SEND` permintaan, direktori file di mana untuk menempatkan file di SFTP server mitra. Ini adalah nilai untuk `RemoteDirectoryPath` yang diteruskan pengguna ke `StartFileTransfer` permintaan. Anda dapat menentukan direktori default di SFTP server mitra. Jika demikian, bidang ini kosong.

`bytes`

Jumlah byte yang ditransfer. Nilainya adalah 0 untuk transfer yang gagal.

`local-file-location`

Parameter ini berisi rincian lokasi file AWS penyimpanan.

`domain`

Penyimpanan yang digunakan. Saat ini, satu-satunya nilai adalah `S3`.

bucket

Wadah untuk objek di Amazon S3.

key

Nama yang ditetapkan untuk objek di Amazon S3.

Example SFTPKonektor File Kirim contoh peristiwa gagal

Contoh berikut menunjukkan peristiwa di mana SFTP konektor gagal saat mencoba mengirim file ke SFTP server jauh.

```
{
  "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 SFTPPengambilan File Konektor Contoh acara yang lengkap

Contoh berikut menunjukkan peristiwa di mana SFTP konektor berhasil mengambil file yang dikirim dari SFTP server jauh.

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

AS2acara

Berikut ini adalah bidang detail untuk AS2 acara:

- AS2Payload Terima Selesai
- AS2Penerimaan Muatan Gagal
- AS2Muatan Kirim Selesai
- AS2Kirim Muatan Gagal
- AS2MDNMenerima Selesai
- AS2MDNMenerima Gagal
- AS2MDNKirim Selesai
- AS2MDNKirim Gagal

detail-typeBidang source dan disertakan di bawah ini karena mengandung nilai khusus untuk Transfer Family acara. Untuk definisi bidang metadata lain yang disertakan dalam semua peristiwa, lihat [Referensi struktur acara](#) di Amazon EventBridge Panduan Pengguna.

```
{
  . . . ,
  "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

Mengidentifikasi jenis acara.

Untuk acara ini, nilainya adalah salah satu AS2 peristiwa yang tercantum sebelumnya.

source

Mengidentifikasi layanan yang menghasilkan peristiwa. Untuk Transfer Family acara, nilai ini adalah `aws.transfer`.

detail

JSONObjek yang berisi informasi tentang acara tersebut. Layanan yang menghasilkan acara menentukan konten bidang ini.

s3-attributes

Mengidentifikasi bucket dan kunci Amazon S3 untuk file yang ditransfer. Untuk MDN acara, ini juga mengidentifikasi bucket dan kunci untuk MDN file tersebut.

file-bucket

Wadah untuk objek di Amazon S3.

file-key

Nama yang ditetapkan untuk objek di Amazon S3.

json-bucket

Untuk COMPLETED atau FAILED transfer, wadah untuk JSON file.

json-key

Untuk COMPLETED atau FAILED transfer, nama yang ditetapkan ke JSON file di Amazon S3.

`mdn-bucket`

Untuk MDN acara, wadah untuk MDN file.

`mdn-key`

Untuk MDN acara, nama yang ditetapkan ke MDN file di Amazon S3.

`mdn-subject`

Untuk MDN acara, deskripsi teks untuk disposisi pesan.

`mdn-message-id`

Untuk MDN acara, ID unik untuk MDN pesan.

`disposition`

Untuk MDN acara, kategori untuk disposisi.

`bytes`

Jumlah byte dalam pesan.

`as2-from`

Mitra AS2 dagang yang mengirim pesan.

`as2-message-id`

Pengenal unik untuk AS2 pesan yang ditransfer.

`as2-to`

Mitra AS2 dagang yang menerima pesan.

`connector-id`

Untuk AS2 pesan yang dikirim dari server Transfer Family ke mitra dagang, pengenal unik untuk AS2 konektor yang digunakan.

`client-ip`

Untuk kejadian server (transfer dari mitra dagang ke server Transfer Family), alamat IP untuk klien yang terlibat dalam transfer.

`agreement-id`

Untuk peristiwa server, pengenal unik untuk AS2 perjanjian.

`server-id`

Untuk kejadian server, ID unik hanya untuk server Transfer Family.

`requester-file-name`

Untuk acara payload, nama asli untuk file yang diterima selama transfer.

`message-subject`

Deskripsi teks untuk subjek pesan.

`start-timestamp`

Untuk transfer yang berhasil, stempel waktu saat pemrosesan file dimulai.

`end-timestamp`

Untuk transfer yang berhasil, stempel waktu saat pemrosesan file selesai.

`status-code`

Kode yang sesuai dengan keadaan proses transfer AS2 pesan. Nilai yang valid: COMPLETED | FAILED | PROCESSING.

`failure-code`

Untuk transfer yang gagal, kategori mengapa transfer gagal.

`failure-message`

Untuk transfer yang gagal, detail mengapa transfer gagal.

`transfer-id`

Pengenal unik untuk acara transfer.

Example AS2Payload Menerima Contoh acara yang lengkap

```
{
  "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 AS2MDNMenerima contoh peristiwa Gagal

```

{
    "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"
}
}
```

Keamanan di AWS Transfer Family

Keamanan cloud di AWS adalah prioritas tertinggi. Sebagai AWS pelanggan, Anda mendapat manfaat dari pusat data dan arsitektur jaringan yang dibangun untuk memenuhi persyaratan organisasi yang paling sensitif terhadap keamanan.

Keamanan adalah tanggung jawab bersama antara Anda AWS dan Anda. [Model tanggung jawab bersama](#) menjelaskan hal ini sebagai keamanan cloud dan keamanan dalam cloud:

Untuk mempelajari apakah Layanan AWS berada dalam lingkup program kepatuhan tertentu, lihat [Layanan AWS di Lingkup oleh Program Kepatuhan Layanan AWS](#) dan pilih program kepatuhan yang Anda minati. Untuk informasi umum, lihat [Program AWS Kepatuhan Program AWS](#).

Anda dapat mengunduh laporan audit pihak ketiga menggunakan AWS Artifact. Untuk informasi selengkapnya, lihat [Mengunduh Laporan di AWS Artifact](#).

Tanggung jawab kepatuhan Anda saat menggunakan Layanan AWS ditentukan oleh sensitivitas data Anda, tujuan kepatuhan perusahaan Anda, dan hukum dan peraturan yang berlaku. AWS menyediakan sumber daya berikut untuk membantu kepatuhan:

- [Panduan Memulai Cepat Keamanan dan Kepatuhan — Panduan](#) penerapan ini membahas pertimbangan arsitektur dan memberikan langkah-langkah untuk menerapkan lingkungan dasar AWS yang berfokus pada keamanan dan kepatuhan.
- [Arsitektur untuk HIPAA Keamanan dan Kepatuhan di Amazon Web Services](#) — Whitepaper ini menjelaskan bagaimana perusahaan dapat menggunakan AWS untuk membuat HIPAA aplikasi yang memenuhi syarat.

Note

Tidak semua Layanan AWS HIPAA memenuhi syarat. Untuk informasi selengkapnya, lihat [Referensi Layanan yang HIPAA Memenuhi Syarat](#).

- [AWS Sumber Daya AWS](#) — Kumpulan buku kerja dan panduan ini mungkin berlaku untuk industri dan lokasi Anda.
- [AWS Panduan Kepatuhan Pelanggan](#) - Memahami model tanggung jawab bersama melalui lensa kepatuhan. Panduan ini merangkum praktik terbaik untuk mengamankan Layanan AWS dan memetakan panduan untuk kontrol keamanan di berbagai kerangka kerja (termasuk Institut

Standar dan Teknologi Nasional (NIST), Dewan Standar Keamanan Industri Kartu Pembayaran (PCI), dan Organisasi Internasional untuk Standardisasi (ISO).

- [Mengevaluasi Sumber Daya dengan Aturan](#) dalam Panduan AWS Config Pengembang — AWS Config Layanan menilai seberapa baik konfigurasi sumber daya Anda mematuhi praktik internal, pedoman industri, dan peraturan.
- [AWS Security Hub](#)— Ini Layanan AWS memberikan pandangan komprehensif tentang keadaan keamanan Anda di dalamnya AWS. Security Hub menggunakan kontrol keamanan untuk sumber daya AWS Anda serta untuk memeriksa kepatuhan Anda terhadap standar industri keamanan dan praktik terbaik. Untuk daftar layanan dan kontrol yang didukung, lihat [Referensi kontrol Security Hub](#).
- [AWS Audit Manager](#)Ini Layanan AWS membantu Anda terus mengaudit AWS penggunaan Anda untuk menyederhanakan cara Anda mengelola risiko dan kepatuhan terhadap peraturan dan standar industri.

Dokumentasi ini membantu Anda memahami cara menerapkan model tanggung jawab bersama saat menggunakan AWS Transfer Family. Topik berikut menunjukkan cara mengonfigurasi AWS Transfer Family untuk memenuhi tujuan keamanan dan kepatuhan Anda. Anda juga belajar cara menggunakan AWS layanan lain yang membantu Anda memantau dan mengamankan AWS Transfer Family sumber daya Anda.

Topik

- [Kebijakan keamanan untuk AWS Transfer Family](#)
- [Menggunakan pertukaran kunci pasca-kuantum hibrida dengan AWS Transfer Family](#)
- [Perlindungan data di AWS Transfer Family](#)
- [Identitas dan manajemen akses untuk AWS Transfer Family](#)
- [Validasi kepatuhan untuk AWS Transfer Family](#)
- [Ketahanan di AWS Transfer Family](#)
- [Keamanan infrastruktur di AWS Transfer Family](#)
- [Tambahkan firewall aplikasi web](#)
- [Pencegahan confused deputy lintas layanan](#)
- [AWS kebijakan terkelola untuk AWS Transfer Family](#)

Kebijakan keamanan untuk AWS Transfer Family

Kebijakan keamanan server AWS Transfer Family memungkinkan Anda untuk membatasi set algoritma kriptografi (kode otentikasi pesan (), pertukaran kunci (MACs), dan cipher suiteKEXs) yang terkait dengan server Anda. Untuk daftar algoritma kriptografi yang didukung, lihat. [Algoritma kriptografi](#) Untuk daftar algoritme kunci yang didukung untuk digunakan dengan kunci host server dan kunci pengguna yang dikelola layanan, lihat. [Algoritma yang didukung untuk kunci pengguna dan server](#)

Note

Kami sangat menyarankan untuk memperbarui server Anda ke kebijakan keamanan terbaru kami. Kebijakan keamanan terbaru kami adalah default. Setiap pelanggan yang membuat server Transfer Family menggunakan CloudFormation dan menerima kebijakan keamanan default akan secara otomatis ditetapkan kebijakan terbaru. Jika Anda khawatir tentang kompatibilitas klien, harap sebutkan kebijakan keamanan mana yang ingin Anda gunakan saat membuat atau memperbarui server daripada menggunakan kebijakan default, yang dapat berubah sewaktu-waktu.

Untuk mengubah kebijakan keamanan server, lihat[Edit kebijakan keamanan](#).

Untuk informasi selengkapnya tentang keamanan di Transfer Family, lihat postingan blog, [Bagaimana Transfer Family dapat membantu Anda membangun solusi transfer file terkelola yang aman dan sesuai](#).

Topik

- [Algoritma kriptografi](#)
- [TransferSecurityPolicy-2024-01](#)
- [TransferSecurityPolicy-2023-05](#)
- [TransferSecurityPolicy-2022-03](#)
- [TransferSecurityPolicy-2020-06](#)
- [TransferSecurityPolicy-2018-11](#)
- [TransferSecurityPolicy- FIPS -2024-01](#)
- [TransferSecurityPolicy- FIPS -2023-05](#)
- [TransferSecurityPolicy- FIPS -2020-06](#)

- [Pasca kebijakan keamanan Quantum](#)

Note

`TransferSecurityPolicy-2024-01` adalah kebijakan keamanan default yang dilampirkan ke server Anda saat membuat server menggunakan konsol, API, atau CLI.

Algoritma kriptografi

Untuk kunci host, kami mendukung algoritma berikut:

- `rsa-sha-256`
- `rsa-sha-512`
- `ecdsa-sha2-nistp256`
- `ecdsa-sha2-nistp384`
- `ecdsa-sha2-nistp512`
- `ssh-ed25519`

Selain itu, kebijakan keamanan 2018 dan 2020 memungkinkan `ssh-rsa`.

Note

Penting untuk memahami perbedaan antara tipe RSA kunci — yang selalu `ssh-rsa` — dan algoritma kunci RSA host, yang dapat berupa algoritme yang didukung.

Berikut ini adalah daftar algoritma kriptografi yang didukung untuk setiap kebijakan keamanan.

Note

Dalam tabel dan kebijakan berikut, perhatikan penggunaan jenis algoritma berikut.

- SFTP server hanya menggunakan algoritma di `SshCiphers`, `SshKexs`, dan `SshMacsbagian`.
- FTPS server hanya menggunakan algoritma di `TlsCiphersbagian` ini.

- FTPserver, karena mereka tidak menggunakan enkripsi, tidak menggunakan salah satu algoritma ini.

Kebijakan keamanan	2024-01	2023-05	2022-03	2020-06	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								
kurva25519-sha256	◆	◆	◆					◆

Kebijakan keamanan	2024-01	2023-05	2022-03	2020-06	FIPS-2024-01	FIPS-2023-05	FIPS-2020-06	2018-11
curve25519-sha256@libssh.org	◆	◆	◆					◆
diffie-hellman-groupup14-sha1								◆
diffie-hellman-groupup14-sha256				◆			◆	◆
diffie-hellman-groupup16-sha512	◆	◆	◆	◆	◆	◆	◆	◆
diffie-hellman-groupup18-sha512	◆	◆	◆	◆	◆	◆	◆	◆

Kebijakan keamanan	2024-01	2023-05	2022-03	2020-06	FIPS-2024-01	FIPS-2023-05	FIPS-2020-06	2018-11
diffie-hellman-group-exchange-sha256		◆	◆	◆		◆	◆	◆
ecdh-nist-p256-kurve-512r3-sha256-d00@openquantumsafe.org	◆				◆			
ecdh-nist-p384-kurve-768r3-sha384-d00@openquantumsafe.org	◆				◆			

Kebijakan keamanan	2024-01	2023-05	2022-03	2020-06	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	◆							
SshMacs								

Kebijakan keamanan	2024-01	2023-05	2022-03	2020-06	FIPS-2024-01	FIPS-2023-05	FIPS-2020-06	2018-11
hmac-sha1								◆
hmac-sha1-etm@openssh.com								◆
hmac-sha2-256			◆	◆			◆	◆
hmac-sha2-256-etm@openssh.com	◆	◆	◆	◆	◆	◆	◆	◆
hmac-sha2-512			◆	◆			◆	◆
hmac-sha2-512-etm@openssh.com	◆	◆	◆	◆	◆	◆	◆	◆
umac-128-etm@openssh.com				◆				◆

Kebijakan keamanan	2024-01	2023-05	2022-03	2020-06	FIPS-2024-01	FIPS-2023-05	FIPS-2020-06	2018-11
umac-128@openssh.com				◆				◆
umac-64-etm@openssh.com								◆
umac-64@openssh.com								◆
TlsCiphers								
TLS__ECDHE__ECDSA_WITH__AES_CBC__128__SHA256	◆	◆	◆	◆	◆	◆	◆	◆
TLS__ECDHE__ECDSA_WITH__AES_GCM__128__SHA256	◆	◆	◆	◆	◆	◆	◆	◆

Kebijakan keamanan	2024-01	2023-05	2022-03	2020-06	FIPS-2024-01	FIPS-2023-05	FIPS-2020-06	2018-11
TLS__ECDHE	◆	◆	◆	◆	◆	◆	◆	◆
—								
ECDSA								
WITH								
_AES								
CBC								
_256__								
SHA384								
TLS__ECDHE	◆	◆	◆	◆	◆	◆	◆	◆
—								
ECDSA								
WITH								
_AES								
GCM								
_256__								
SHA384								
TLS__ECDHE	◆	◆	◆	◆	◆	◆	◆	◆
_RSA								
WITH								
_AES								
CBC								
_128__								
SHA256								

Kebijakan keamanan	2024-01	2023-05	2022-03	2020-06	FIPS-2024-01	FIPS-2023-05	FIPS-2020-06	2018-11
TLS_ ECDHE_ _RSA WITH_ _AES GCM_ _128_ SHA256	◆	◆	◆	◆	◆	◆	◆	◆
TLS_ ECDHE_ _RSA WITH_ _AES CBC_ _256_ SHA384	◆	◆	◆	◆	◆	◆	◆	◆
TLS_ ECDHE_ _RSA WITH_ _AES GCM_ _256_ SHA384	◆	◆	◆	◆	◆	◆	◆	◆
TLS_ _RSA WITH_ _AES CBC_ _128_ SHA256								◆

Kebijakan keamanan	2024-01	2023-05	2022-03	2020-06	FIPS-2024-01	FIPS-2023-05	FIPS-2020-06	2018-11
TLS_								◆
_RSA								
WITH								
_AES								
CBC								
256								
SHA256								

TransferSecurityPolicy-2024-01

Berikut ini menunjukkan kebijakan keamanan TransferSecurityPolicy -2024-01.

```
{
  "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"
    ]
  }
}

```

TransferSecurityPolicy-2023-05

Berikut ini menunjukkan kebijakan keamanan TransferSecurityPolicy -2023-05.

```

{
  "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"
    ]
}
}

```

TransferSecurityPolicy-2022-03

Berikut ini menunjukkan kebijakan keamanan TransferSecurityPolicy -2022-03.

```

{
  "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"
  ]
}
}

```

TransferSecurityPolicy-2020-06

Berikut ini menunjukkan kebijakan keamanan TransferSecurityPolicy -2020-06.

```

{
  "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"
  ]
}
}

```

TransferSecurityPolicy-2018-11

Berikut ini menunjukkan kebijakan keamanan TransferSecurityPolicy -2018-11.

```

{
  "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"
  ]
}
}

```

TransferSecurityPolicy- FIPS -2024-01

Berikut ini menunjukkan TransferSecurityPolicy - FIPS -2024-01 kebijakan keamanan.

Note

Titik akhir FIPS layanan dan TransferSecurityPolicy - FIPS -2024-01 kebijakan keamanan hanya tersedia di beberapa Wilayah. AWS Untuk informasi selengkapnya, lihat [AWS Transfer Family titik akhir dan kuota](#) di. Referensi Umum AWS

```

{
  "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"
    ]
}

```

TransferSecurityPolicy- FIPS -2023-05

Rincian FIPS sertifikasi untuk AWS Transfer Family dapat ditemukan di <https://csrc.nist.gov/projects/cryptographic-module-validation-program/validated-modules/search/all>

Berikut ini menunjukkan TransferSecurityPolicy - FIPS -2023-05 kebijakan keamanan.

 Note

Titik akhir FIPS layanan dan TransferSecurityPolicy - FIPS -2023-05 kebijakan keamanan hanya tersedia di beberapa Wilayah. AWS Untuk informasi selengkapnya, lihat [AWS Transfer Family titik akhir dan kuota](#) di. Referensi Umum AWS

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

TransferSecurityPolicy- FIPS -2020-06

Rincian FIPS sertifikasi untuk AWS Transfer Family dapat ditemukan di <https://csrc.nist.gov/projects/cryptographic-module-validation-program/validated-modules/search/all>

Berikut ini menunjukkan TransferSecurityPolicy - FIPS -2020-06 kebijakan keamanan.

Note

Titik akhir FIPS layanan dan kebijakan keamanan TransferSecurityPolicy - FIPS -2020-06 hanya tersedia di beberapa Wilayah. AWS Untuk informasi selengkapnya, lihat [AWS Transfer Family titik akhir dan kuota](#) di. Referensi Umum AWS

```
{
  "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"
  ]
}
```

Pasca kebijakan keamanan Quantum

Tabel ini mencantumkan algoritme untuk kebijakan keamanan kuantum pasca Transfer Family. Kebijakan ini dijelaskan secara rinci dalam[Menggunakan pertukaran kunci pasca-kuantum hibrida dengan AWS Transfer Family](#).

Daftar kebijakan mengikuti tabel.

Kebijakan keamanan	TransferSecurityPolicy-PQ- - Eksperimental-2023-04 SSH	TransferSecurityPolicy-PQ- - -Eksperimental-2023-04 SSH FIPS
SSH ciphers		
aes128-ctr		◆
aes128-gcm@openssh.com	◆	◆
aes192-ctr	◆	◆
aes256-ctr	◆	◆
aes256-gcm@openssh.com	◆	◆
KEXs		
ecdh-nistp256-kyber-512r3- sha256-d00@openquan tumsafe.org	◆	◆

Kebijakan keamanan	TransferSecurityPolicy-PQ- - Eksperimental-2023-04 SSH	TransferSecurityPolicy-PQ- - -Eksperimental-2023-04 SSH FIPS
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-group14-sha256		◆
diffie-hellman-group16-sha512	◆	◆
diffie-hellman-group18-sha512	◆	◆
ecdh-sha2-nistp384		◆
ecdh-sha2-nistp521		◆
diffie-hellman-group-exchange-sha256	◆	◆
ecdh-sha2-nistp256		◆
curve25519-sha256@libssh.org	◆	
kurva25519-sha256	◆	
MACs		
hmac-sha2-256-etm@openssh.com	◆	◆
hmac-sha2-256	◆	◆

Kebijakan keamanan	TransferSecurityPolicy-PQ- - Eksperimental-2023-04 SSH	TransferSecurityPolicy-PQ- - -Eksperimental-2023-04 SSH FIPS
hmac-sha2-512-etm@ openssh.com	◆	◆
hmac-sha2-512	◆	◆
TLS ciphers		
TLS_ _ ECDHE _ ECDSA WITH _ AES CBC _128_ _ SHA256	◆	◆
TLS_ _ ECDHE _ ECDSA WITH _ AES GCM _128_ _ SHA256	◆	◆
TLS_ _ ECDHE _ ECDSA WITH _ AES CBC _256_ _ SHA384	◆	◆
TLS_ _ ECDHE _ ECDSA WITH _ AES GCM _256_ _ SHA384	◆	◆
TLS_ _ ECDHE _ RSA WITH _ AES CBC _128_ _ SHA256	◆	◆
TLS_ _ ECDHE _ RSA WITH _ AES GCM _128_ _ SHA256	◆	◆
TLS_ _ ECDHE _ RSA WITH _ AES CBC _256_ _ SHA384	◆	◆
TLS_ _ ECDHE _ RSA WITH _ AES GCM _256_ _ SHA384	◆	◆

TransferSecurityPolicy-PQ- -Eksperimental-2023-04 SSH

Berikut ini menunjukkan kebijakan keamanan TransferSecurityPolicy-PQ-SSH-Experimental-2023-04.

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

}

TransferSecurityPolicy-PQ- -Eksperimental-2023-04 SSH FIPS

Berikut ini menunjukkan kebijakan keamanan TransferSecurityPolicy-PQ-SSH-FIPS - Eksperimental-2023-04.

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

Menggunakan pertukaran kunci pasca-kuantum hibrida dengan AWS Transfer Family

AWS Transfer Family mendukung opsi pembentukan kunci pasca-kuantum hibrida untuk protokol Secure Shell (SSH). Pembentukan kunci pasca-kuantum diperlukan karena sudah memungkinkan untuk merekam lalu lintas jaringan dan menyimpannya untuk dekripsi di masa depan oleh komputer kuantum, yang disebut serangan. store-now-harvest-later

Anda dapat menggunakan opsi ini saat terhubung ke Transfer Family untuk transfer file aman ke dalam dan keluar dari penyimpanan Amazon Simple Storage Service (Amazon S3) atau Amazon Elastic File System (Amazon EFS). Pembentukan kunci hibrida pasca-kuantum dalam SSH memperkenalkan mekanisme pembentukan kunci pasca-kuantum, yang digunakannya bersama dengan algoritma pertukaran kunci klasik. SSH kunci yang dibuat dengan cipher suite klasik aman dari serangan brute-force dengan teknologi saat ini. Namun, enkripsi klasik diperkirakan tidak akan tetap aman setelah munculnya komputasi kuantum skala besar di masa depan.

Jika organisasi Anda bergantung pada kerahasiaan data jangka panjang yang diteruskan melalui koneksi Transfer Family, Anda harus mempertimbangkan rencana untuk bermigrasi ke kriptografi pasca-kuantum sebelum komputer kuantum skala besar tersedia untuk digunakan.

Untuk melindungi data yang dienkripsi hari ini terhadap potensi serangan future, AWS berpartisipasi dengan komunitas kriptografi dalam pengembangan algoritma tahan kuantum atau pasca-kuantum. Kami telah menerapkan rangkaian cipher pertukaran kunci pasca-kuantum hibrida di Transfer Family yang menggabungkan elemen klasik dan pasca-kuantum.

Suite sandi hibrida ini tersedia untuk digunakan pada beban kerja produksi Anda di sebagian besar Wilayah. AWS Namun, karena karakteristik kinerja dan persyaratan bandwidth suite cipher hybrid berbeda dengan mekanisme pertukaran kunci klasik, kami sarankan Anda mengujinya pada koneksi Transfer Family Anda.

Cari tahu lebih lanjut tentang kriptografi pasca-kuantum di posting blog keamanan [Post-Quantum Cryptography](#).

Daftar Isi

- [Tentang pertukaran kunci hibrida pasca-kuantum di SSH](#)
- [Cara kerja pembentukan kunci hibrida pasca-kuantum di Transfer Family](#)
 - [Mengapa Kyber?](#)
 - [Pertukaran SSH kunci hibrida pasca-kuantum dan persyaratan kriptografi \(140\) FIPS](#)
- [Menguji pertukaran kunci hibrida pasca-kuantum di Transfer Family](#)
 - [Aktifkan pertukaran kunci hybrid pasca-kuantum di titik akhir Anda SFTP](#)
 - [Siapkan SFTP klien yang mendukung pertukaran kunci hybrid pasca-kuantum](#)
 - [Konfirmasikan pertukaran kunci hibrida pasca-kuantum di SFTP](#)

Tentang pertukaran kunci hibrida pasca-kuantum di SSH

[Transfer Family mendukung suite cipher pertukaran kunci hibrida pasca-kuantum, yang menggunakan algoritme pertukaran kunci Elliptic Curve Diffie-Hellman \(\) klasik, dan Kyber. ECDH CRYSTALS](#) Kyber adalah enkripsi kunci publik pasca-kuantum dan algoritma pembentukan kunci yang telah ditetapkan oleh [National Institute for Standards and Technology \(NIST\)](#) sebagai algoritma perjanjian kunci pasca-kuantum standar pertama.

Klien dan server masih melakukan pertukaran ECDH kunci. Selain itu, server merangkum rahasia bersama pasca-kuantum ke kunci KEM publik pasca-kuantum klien, yang diiklankan dalam pesan pertukaran kunci klien. SSH Strategi ini menggabungkan jaminan tinggi dari pertukaran kunci klasik dengan keamanan pertukaran kunci pasca-kuantum yang diusulkan, untuk membantu memastikan bahwa jabat tangan dilindungi selama rahasia bersama ECDH atau pasca-kuantum tidak dapat dipatahkan.

Cara kerja pembentukan kunci hibrida pasca-kuantum di Transfer Family

AWS baru-baru ini mengumumkan dukungan untuk pertukaran kunci pasca-kuantum dalam transfer SFTP file di AWS Transfer Family. Transfer Family dengan aman menskalakan transfer business-to-business file ke layanan AWS Storage menggunakan SFTP dan protokol lainnya. SFTP adalah versi yang lebih aman dari File Transfer Protocol (FTP) yang berjalan di atas SSH. Dukungan pertukaran kunci pasca-kuantum dari Transfer Family meningkatkan bilah keamanan untuk transfer data. SFTP

Dukungan pertukaran kunci hibrida pasca-kuantum di Transfer Family termasuk menggabungkan algoritma pasca-kuantum Kyber-512, Kyber-768, dan Kyber-1024, dengan ECDH

lebih dari kurva P256, P384, P521, atau Curve25519. Metode pertukaran SSH kunci yang sesuai berikut ini ditentukan dalam [draf pertukaran SSH kunci hibrida pasca-kuantum](#).

- `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

Metode pertukaran kunci baru ini dapat berubah saat rancangan berkembang menuju standarisasi, atau ketika NIST meratifikasi algoritma Kyber.

Mengapa Kyber?

AWS berkomitmen untuk mendukung standar, algoritma interoperable. [Kyber adalah algoritma enkripsi pasca-kuantum pertama yang dipilih untuk standarisasi oleh proyek Post-Quantum Cryptography. NIST](#) Beberapa badan standar sudah mengintegrasikan Kyber ke dalam protokol. AWS sudah mendukung Kyber TLS di beberapa titik AWS API akhir.

Sebagai bagian dari komitmen ini, AWS telah mengajukan draf proposal ke kriptografi pasca-kuantum yang menggabungkan Kyber dengan kurva yang NIST disetujui seperti P256 IETF untuk SSH. Untuk membantu meningkatkan keamanan bagi pelanggan kami, AWS implementasi pertukaran kunci pasca-kuantum dalam SFTP dan SSH mengikuti draf itu. Kami berencana untuk mendukung pembaruan masa depan sampai proposal kami diadopsi oleh IETF dan menjadi standar.

Metode pertukaran kunci baru (tercantum di bagian [Cara kerja pembentukan kunci hibrida pasca-kuantum di Transfer Family](#)) mungkin berubah saat draf berkembang menuju standarisasi atau ketika NIST meratifikasi algoritma Kyber.

Note

Dukungan algoritme pasca-kuantum saat ini tersedia untuk pertukaran kunci hibrida pasca-kuantum TLS untuk AWS KMS (lihat [Menggunakan pasca-kuantum hibrida TLS dengan AWS KMS](#)), AWS Certificate Manager, dan titik akhir. AWS Secrets Manager API

Pertukaran SSH kunci hibrida pasca-kuantum dan persyaratan kriptografi (140) FIPS

Untuk pelanggan yang membutuhkan FIPS kepatuhan, Transfer Family menyediakan kriptografi yang FIPS disetujui SSH dengan menggunakan pustaka kriptografi sumber terbuka AWS FIPS bersertifikat 140, -LC. AWS Metode pertukaran kunci hibrida pasca-kuantum yang didukung dalam TransferSecurityPolicy-PQ-SSH-FIPS -Experimental-2023-04 di Transfer FIPS Family disetujui [NIST menurut SP 800-56Cr2 \(bagian 2\)](#). Kantor Federal Jerman untuk Keamanan Informasi ([BSI](#)) dan Agence nationale de la sécurité des systèmes d'information ([ANSSI](#)) Prancis juga merekomendasikan metode pertukaran kunci hibrida pasca-kuantum tersebut.

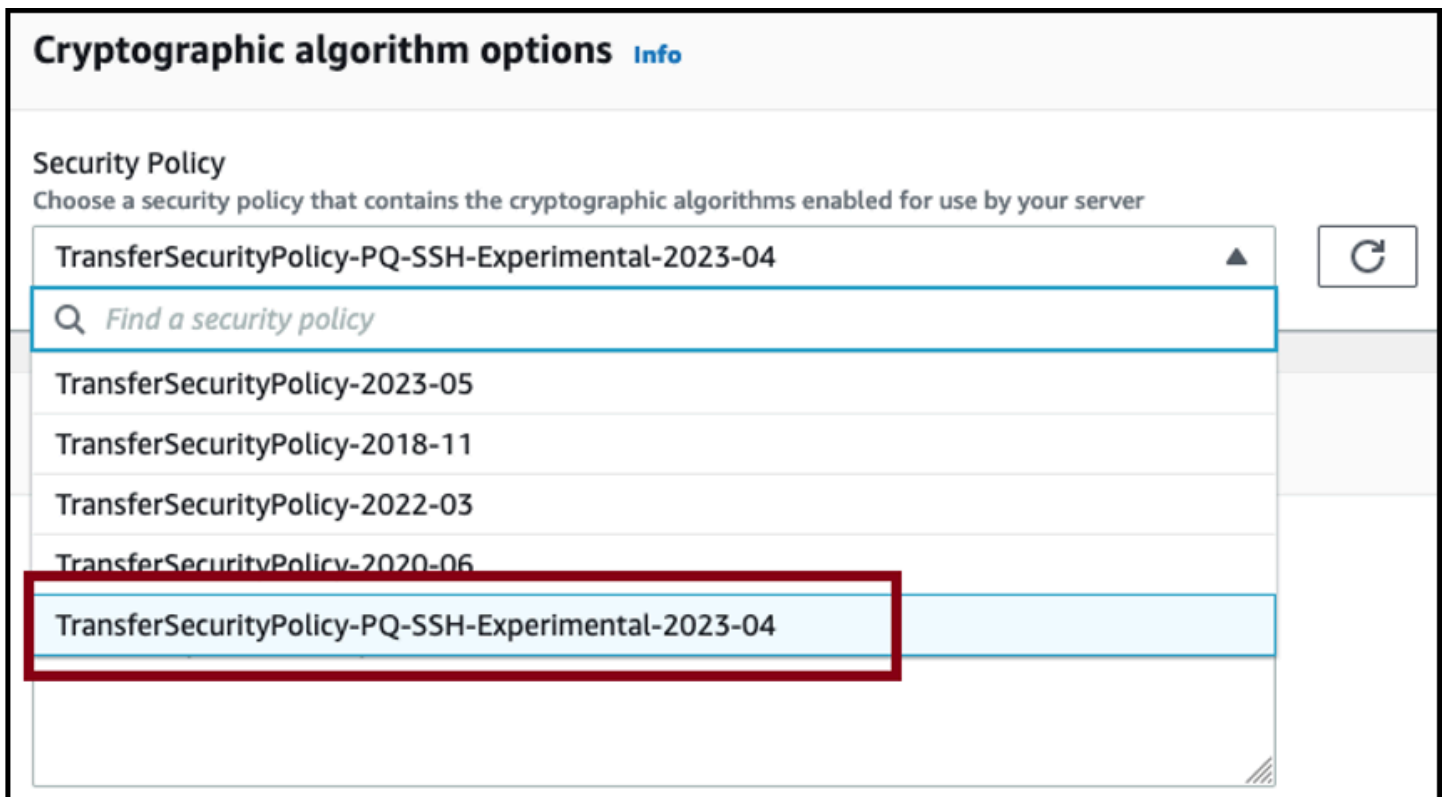
Menguji pertukaran kunci hibrida pasca-kuantum di Transfer Family

Bagian ini menjelaskan langkah-langkah yang Anda ambil untuk menguji pertukaran kunci hibrida pasca-kuantum.

1. [Aktifkan pertukaran kunci hybrid pasca-kuantum di titik akhir Anda SFTP.](#)
2. Gunakan SFTP klien (seperti [Siapkan SFTP klien yang mendukung pertukaran kunci hybrid pasca-kuantum](#)) yang mendukung pertukaran kunci hibrida pasca-kuantum dengan mengikuti panduan dalam spesifikasi draf yang disebutkan di atas.
3. Transfer file menggunakan server Transfer Family.
4. [Konfirmasikan pertukaran kunci hibrida pasca-kuantum di SFTP.](#)

Aktifkan pertukaran kunci hybrid pasca-kuantum di titik akhir Anda SFTP

Anda dapat memilih SSH kebijakan saat membuat endpoint SFTP server baru di Transfer Family, atau dengan mengedit opsi algoritma Kriptografi di titik akhir yang ada SFTP. Snapshot berikut menunjukkan contoh AWS Management Console tempat Anda memperbarui SSH kebijakan.



Nama SSH kebijakan yang mendukung pertukaran kunci pasca-kuantum adalah -PQ- -Eksperimental-2023-04 dan TransferSecurityPolicy-PQ- -SSH -Eksperimental-2023-04. TransferSecurityPolicy SSH FIPS Untuk detail selengkapnya tentang kebijakan Transfer Family, lihat [Kebijakan keamanan untuk AWS Transfer Family](#).

Siapkan SFTP klien yang mendukung pertukaran kunci hybrid pasca-kuantum

Setelah memilih SSH kebijakan pasca-kuantum yang benar di titik akhir SFTP Transfer Family, Anda dapat bereksperimen dengan pasca-kuantum di Transfer SFTP Family. Anda dapat menggunakan SFTP klien (seperti [OQSBuka SSH](#)) yang mendukung pertukaran kunci hibrida pasca-kuantum dengan mengikuti panduan dalam spesifikasi draf yang disebutkan di atas.

OQSOpen SSH adalah fork open source dari Open SSH yang menambahkan kriptografi kuantum aman dengan menggunakan. SSH liboqs liboqs adalah pustaka C open-source yang mengimplementasikan algoritma kriptografi tahan kuantum. OQSBuka SSH dan liboqs merupakan bagian dari proyek Open Quantum Safe (OQS).

Untuk menguji pertukaran kunci hybrid pasca-kuantum di Transfer Family SFTP with OQS OpenSSH, Anda perlu membangun OQS Open SSH seperti yang dijelaskan dalam proyek. [README](#) Setelah Anda membangun OQS OpenSSH, Anda dapat menjalankan contoh SFTP klien untuk

terhubung ke SFTP titik akhir Anda (misalnya, `s-1111aaaa2222bbbb3.server.transfer.us-west-2.amazonaws.com`) dengan menggunakan metode pertukaran kunci hybrid pasca-kuantum, seperti yang ditunjukkan pada perintah berikut.

```
./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
```

Pada perintah sebelumnya, ganti item berikut dengan informasi Anda sendiri:

- Ganti *username_private_key_PEM_file* dengan file kunci pribadi SFTP pengguna yang PEM didekodekan
- Ganti *username* dengan nama SFTP pengguna
- Ganti *server-id* dengan ID server Transfer Family
- Ganti *region-id* dengan wilayah aktual tempat server Transfer Family Anda berada

Konfirmasikan pertukaran kunci hibrida pasca-kuantum di SFTP

Untuk mengonfirmasi bahwa pertukaran kunci hybrid pasca-kuantum digunakan selama SSH koneksi SFTP ke Transfer Family, periksa output klien. Secara opsional, Anda dapat menggunakan program pengambilan paket. Jika Anda menggunakan SSH klien Open Quantum Safe Open, outputnya akan terlihat mirip dengan yang berikut (menghilangkan informasi yang tidak relevan untuk singkatnya):

```
$. /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.Transfer.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>
```

Output menunjukkan bahwa negosiasi klien terjadi menggunakan `ecdh-nistp384-kyber-768r3-sha384-d00@openquantumsafe.org` metode hibrida pasca-kuantum dan berhasil membuat SFTP sesi.

Perlindungan data di AWS Transfer Family

[Model tanggung jawab AWS bersama model](#) berlaku untuk perlindungan data di AWS Transfer Family (Transfer Family). Seperti yang dijelaskan dalam model AWS ini, bertanggung jawab untuk melindungi infrastruktur global yang menjalankan semua AWS Cloud. Anda bertanggung jawab untuk mempertahankan kendali atas konten yang di-host pada infrastruktur ini. Konten ini mencakup konfigurasi keamanan dan tugas manajemen untuk AWS layanan yang Anda gunakan. Untuk informasi selengkapnya tentang privasi data, lihat [Privasi data FAQ](#). Untuk informasi tentang perlindungan data di Eropa, lihat [model tanggung jawab AWS bersama dan](#) posting GDPR blog di Blog AWS Keamanan.

Untuk tujuan perlindungan data, kami menyarankan Anda untuk melindungi kredensial AWS akun dan menyiapkan akun pengguna individu dengan AWS Identity and Access Management (IAM). Dengan cara ini, setiap pengguna hanya diberi izin yang diperlukan untuk memenuhi tanggung jawab tugas mereka. Kami juga merekomendasikan agar Anda mengamankan data Anda dengan cara-cara berikut ini:

- Gunakan otentikasi multi-faktor (MFA) dengan setiap akun.
- Gunakan SSL/TLS untuk berkomunikasi dengan AWS sumber daya. Kami mendukung TLS 1.2.
- Siapkan API dan pencatatan aktivitas pengguna dengan AWS CloudTrail.
- Gunakan solusi AWS enkripsi, bersama dengan semua kontrol keamanan default dalam AWS layanan.
- Gunakan layanan keamanan terkelola lanjutan seperti Amazon Macie, yang membantu menemukan dan mengamankan data pribadi yang disimpan di Amazon S3.
- Jika Anda memerlukan FIPS 140-2 modul kriptografi yang divalidasi saat mengakses AWS melalui antarmuka baris perintah atau, gunakan titik akhir API FIPS Untuk informasi selengkapnya tentang FIPS titik akhir yang tersedia, lihat [Standar pemrosesan informasi Federal \(FIPS\) 140-2](#).

Sebaiknya jangan pernah memasukkan informasi identitas yang sensitif, seperti nomor rekening pelanggan Anda, ke dalam bidang isian bebas seperti bidang Nama. Ini termasuk saat Anda bekerja dengan Transfer Family atau AWS layanan lain menggunakan konsol, API, AWS CLI, atau AWS SDKs. Data konfigurasi apa pun yang Anda masukkan ke dalam konfigurasi layanan Transfer Family, atau konfigurasi layanan lain, dapat diambil untuk dimasukkan dalam log diagnostik. Ketika Anda memberikan URL ke server eksternal, jangan sertakan informasi kredensial dalam URL untuk memvalidasi permintaan Anda ke server tersebut.

Sebaliknya, data dari operasi unggah dan pengunduhan masuk dan keluar dari server Transfer Family diperlakukan sebagai sepenuhnya pribadi dan tidak pernah ada di luar saluran terenkripsi — seperti koneksi atau koneksi. SFTP FTPS Data ini hanya dapat diakses oleh orang yang berwenang.

Topik

- [Enkripsi data di Amazon S3](#)
- [Manajemen kunci](#)

Enkripsi data di Amazon S3

AWS Transfer Family menggunakan opsi enkripsi default yang Anda tetapkan untuk bucket Amazon S3 untuk mengenkripsi data Anda. Saat Anda mengaktifkan enkripsi pada bucket, semua objek dienkripsi saat disimpan di bucket. Objek dienkripsi dengan menggunakan enkripsi sisi server dengan kunci terkelola Amazon S3 (-S3) atau () kunci terkelola (SSE-). AWS Key Management Service AWS KMS SSE KMS Untuk informasi tentang enkripsi sisi server, lihat [Melindungi data menggunakan enkripsi sisi server](#) di Panduan Pengguna Layanan Penyimpanan Sederhana Amazon.

Langkah-langkah berikut menunjukkan cara mengenkripsi data di AWS Transfer Family.

Untuk memungkinkan enkripsi di AWS Transfer Family

1. Aktifkan enkripsi default untuk bucket Amazon S3 Anda. Untuk petunjuknya, lihat [enkripsi default Amazon S3 untuk bucket S3 di Panduan Pengguna](#) Layanan Penyimpanan Sederhana Amazon.
2. Perbarui kebijakan peran AWS Identity and Access Management (IAM) yang dilampirkan ke pengguna untuk memberikan izin required AWS Key Management Service (AWS KMS).
3. Jika Anda menggunakan kebijakan sesi untuk pengguna, kebijakan sesi harus memberikan AWS KMS izin yang diperlukan.

Contoh berikut menunjukkan IAM kebijakan yang memberikan izin minimum yang diperlukan saat menggunakan AWS Transfer Family bucket Amazon S3 yang diaktifkan untuk enkripsi. AWS KMS Sertakan kebijakan contoh ini dalam kebijakan IAM peran pengguna dan kebijakan sesi, jika Anda menggunakannya.

```
{
  "Sid": "Stmt1544140969635",
  "Action": [
    "kms:Decrypt",
    "kms:Encrypt",
    "kms:GenerateDataKey"
  ],
  "Effect": "Allow",
  "Resource": "arn:aws:kms:region:account-id:key/kms-key-id"
}
```

Note

ID KMS kunci yang Anda tentukan dalam kebijakan ini harus sama dengan yang ditentukan untuk enkripsi default pada langkah 1.

Root, atau IAM peran yang digunakan untuk pengguna, harus diizinkan dalam kebijakan AWS KMS kunci. Untuk informasi tentang kebijakan AWS KMS utama, lihat [Menggunakan kebijakan utama AWS KMS di](#) Panduan AWS Key Management Service Pengembang.

Manajemen kunci

Di bagian ini, Anda dapat menemukan informasi tentang SSH kunci, termasuk cara membuatnya dan cara memutarnya. Untuk detail tentang penggunaan Transfer Family with AWS Lambda to manage keys, lihat posting blog [Mengaktifkan manajemen kunci layanan mandiri pengguna dengan A AWS Transfer Family](#) dan. AWS Lambda

Note

AWS Transfer Family menerima RSA, ECDSA, dan ED25519 kunci.

Bagian ini juga mencakup cara membuat dan mengelola kunci Pretty Good Privacy (PGP).

Topik

- [Algoritma yang didukung untuk kunci pengguna dan server](#)
- [Buat SSH kunci untuk pengguna yang dikelola layanan](#)
- [Putar SSH tombol](#)
- [Hasilkan dan kelola PGP kunci](#)
- [PGPKlien yang didukung](#)

Algoritma yang didukung untuk kunci pengguna dan server

Algoritma kunci berikut didukung untuk pasangan kunci pengguna dan server di dalamnya. AWS Transfer Family

 Note

Untuk algoritme yang dapat digunakan dengan PGP dekripsi dalam alur kerja, lihat [Algoritma](#) yang didukung untuk pasangan kunci. PGP

- UntukED25519: `ssh-ed25519`
- UntukRSA:
 - `rsa-sha2-256`
 - `rsa-sha2-512`
- UntukECDSA:
 - `ecdsa-sha2-nistp256`
 - `ecdsa-sha2-nistp384`
 - `ecdsa-sha2-nistp521`

 Note

Kami mendukung `ssh-rsa` kebijakan SHA1 keamanan kami yang lebih lama. Untuk detailnya, lihat [Algoritma kriptografi](#).

Buat SSH kunci untuk pengguna yang dikelola layanan

Anda dapat mengatur server Anda untuk mengautentikasi pengguna menggunakan metode otentikasi terkelola layanan, tempat nama pengguna dan SSH kunci disimpan dalam layanan. SSHKunci publik pengguna diunggah ke server sebagai properti pengguna. Kunci ini digunakan oleh server sebagai bagian dari proses otentikasi berbasis kunci standar. Setiap pengguna dapat memiliki beberapa SSH kunci publik pada file dengan server individual. Untuk batasan jumlah kunci yang dapat disimpan per pengguna, lihat [AWS Transfer Family titik akhir dan kuota](#) di Referensi Umum Amazon Web

Sebagai alternatif metode otentikasi terkelola layanan, Anda dapat mengautentikasi pengguna menggunakan penyedia identitas kustom, atau. AWS Directory Service for Microsoft Active Directory Untuk informasi selengkapnya, lihat [Bekerja dengan penyedia identitas khusus](#) atau [Menggunakan penyedia identitas AWS Directory Service](#).

Server hanya dapat mengautentikasi pengguna menggunakan satu metode (layanan dikelola, layanan direktori, atau penyedia identitas kustom), dan metode itu tidak dapat diubah setelah server dibuat.

Topik

- [Membuat SSH kunci di macOS, Linux, atau Unix](#)
- [Membuat SSH kunci di Microsoft Windows](#)
- [Mengkonversi kunci SSH2 publik ke PEM format](#)

Membuat SSH kunci di macOS, Linux, atau Unix

Pada sistem operasi macOS, Linux, atau Unix, Anda menggunakan `ssh-keygen` perintah untuk membuat kunci SSH publik dan kunci SSH pribadi yang juga dikenal sebagai key pair.

Untuk membuat SSH kunci pada sistem operasi macOS, Linux, atau Unix

1. Pada sistem operasi macOS, Linux, atau Unix, buka terminal perintah.
2. AWS Transfer Family menerima RSA -, ECDSA -, dan ED25519 -kunci yang diformat. Pilih perintah yang sesuai berdasarkan jenis pasangan kunci yang Anda hasilkan.

Note

Dalam contoh berikut, kami tidak menentukan frasa sandi: dalam hal ini, alat meminta Anda untuk memasukkan frasa sandi Anda dan kemudian mengulanginya untuk memverifikasi. Membuat frasa sandi menawarkan perlindungan yang lebih baik untuk kunci pribadi Anda, dan mungkin juga meningkatkan keamanan sistem secara keseluruhan. Anda tidak dapat memulihkan frasa sandi Anda: jika Anda lupa, Anda harus membuat kunci baru.

Namun, jika Anda membuat kunci host server, Anda harus menentukan frasa sandi kosong, dengan menentukan `-N ""` opsi dalam perintah (atau dengan menekan **Enter** dua kali saat diminta), karena server Transfer Family tidak dapat meminta kata sandi saat start-up.

- Untuk menghasilkan key RSA pair 4096-bit:

```
ssh-keygen -t rsa -b 4096 -f key_name
```

- Untuk menghasilkan pasangan kunci ECDSA 521-bit (ECDSA memiliki ukuran bit 256, 384, dan 521):

```
ssh-keygen -t ecdsa -b 521 -f key_name
```

- Untuk menghasilkan ED25519 key pair:

```
ssh-keygen -t ed25519 -f key_name
```

Note

key_name adalah nama file SSH key pair.

Berikut ini menunjukkan contoh ssh-keygen output.

```
ssh-keygen -t rsa -b 4096 -f key_name
Generating public/private rsa key pair.

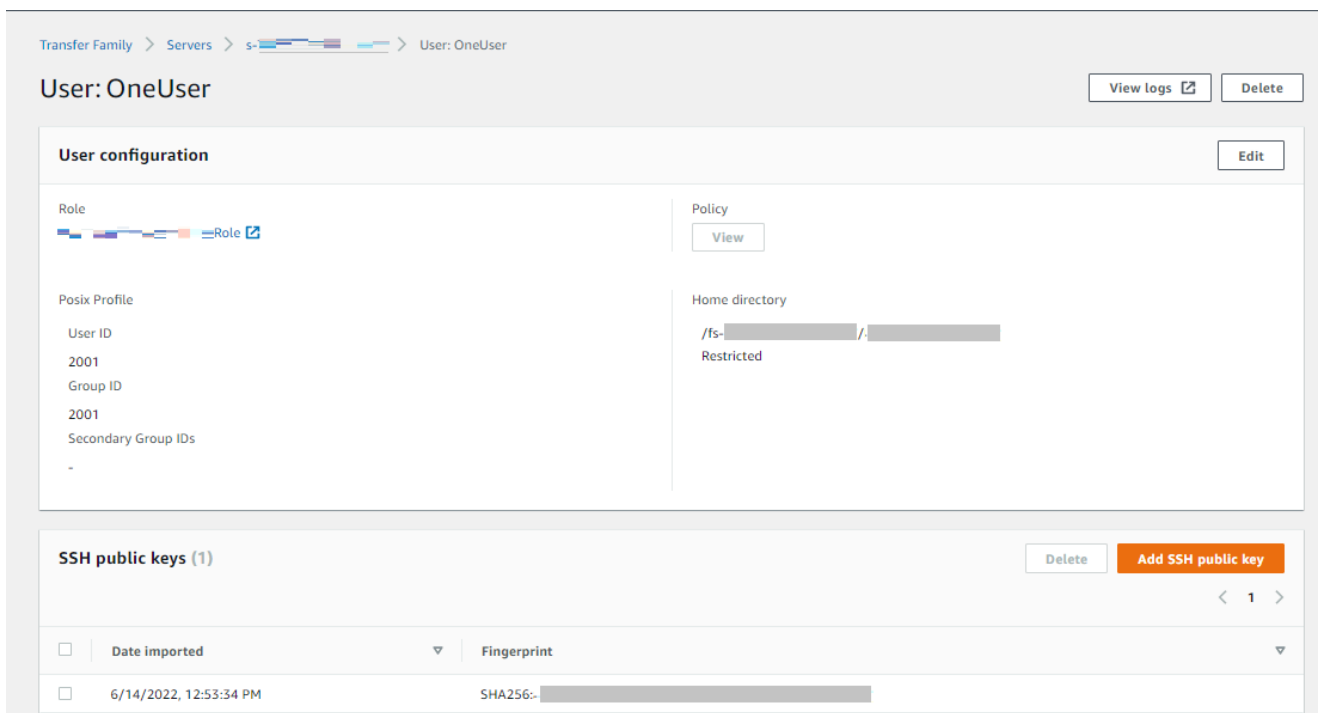
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in key_name.
Your public key has been saved in key_name.pub.
The key fingerprint is:
SHA256:8tDDwPmanTFcEzjTwPGETVW0GW1nVz+gtCCE8hL7PrQ bob.amazon.com
The key's randomart image is:
+---[RSA 4096]-----+
|  . . . . .E      |
|  .   =   ...     |
| . . . = ..o      |
|  . o +   oo =     |
|  + =   .S. = *    |
|  . o o ..B + o    |
|    .o.+.* .      |
|    =o*+*.         |
|    ..*o*+.        |
+----[SHA256]-----+
```


Note

Ketika Anda menjalankan `ssh-keygen` perintah seperti yang ditunjukkan sebelumnya, itu menciptakan kunci publik dan pribadi sebagai file dalam direktori saat ini.

SSHKey pair Anda sekarang siap digunakan. Ikuti langkah 3 dan 4 untuk menyimpan kunci SSH publik bagi pengguna yang dikelola layanan Anda. Pengguna ini menggunakan kunci ketika mereka mentransfer file pada endpoint server Transfer Family.

3. Arahkan ke `key_name`.pub file dan buka.
4. Salin teks dan tempel di kunci SSH publik untuk pengguna yang dikelola layanan.
 - a. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>, lalu pilih Server dari panel navigasi.
 - b. Pada halaman Server, pilih ID Server untuk server yang berisi pengguna yang ingin Anda perbarui.
 - c. Pilih pengguna yang Anda tambahkan kunci publiknya.
 - d. Di panel kunci SSH publik, pilih Tambahkan kunci SSH publik.



- e. Rekatkan teks kunci publik yang Anda buat ke dalam kotak teks kunci SSH publik, lalu pilih Tambah kunci.

Kunci baru tercantum di panel kunci SSH publik.

☐	Date imported	Fingerprint
☐	6/14/2022, 12:53:34 PM	SHA256:-
☐	10/20/2022, 4:26:51 PM	SHA256:-

Membuat SSH kunci di Microsoft Windows

Windows menggunakan format SSH key pair yang sedikit berbeda. Kunci publik harus dalam PUB format, dan kunci pribadi harus dalam PPK format. Di Windows, Anda dapat menggunakan P uTTYgen untuk membuat SSH key pair dalam format yang sesuai. Anda juga dapat menggunakan P uTTYgen untuk mengonversi kunci pribadi yang dihasilkan menggunakan ssh-keygen ke .ppk file.

Note

Jika Anda menyajikan Win SCP dengan file kunci pribadi yang tidak dalam .ppk format, klien itu menawarkan untuk mengubah kunci menjadi .ppk format untuk Anda.

Untuk tutorial tentang membuat SSH kunci dengan menggunakan P uTTYgen di Windows, lihat [situs SSH web.com](https://www.sshweb.com).

Mengkonversi kunci SSH2 publik ke PEM format

AWS Transfer Family hanya menerima kunci publik yang PEM diformat. Jika Anda memiliki kunci SSH2 publik, Anda perlu mengonversinya. Kunci SSH2 publik memiliki format berikut:

```
----- BEGIN SSH2 PUBLIC KEY -----  
Comment: "rsa-key-20160402"  
AAAAB3NzaC1yc2EAAAABJQAAAQEAiL0jjDdFqK/kYThqKt7THrjABTPWvXmB3URI  
:  
:  
----- END SSH2 PUBLIC KEY -----
```

Kunci PEM publik memiliki format berikut:

```
ssh-rsa AAAAB3NzaC1yc2EAAAABJQAAA...
```

Jalankan perintah berikut untuk mengonversi kunci publik SSH2 yang diformat menjadi kunci publik yang PEM diformat. Ganti *ssh2-key* dengan nama SSH2 kunci Anda, dan *PEM-key* dengan nama PEM kunci Anda.

```
ssh-keygen -i -f ssh2-key.pub > PEM-key.pub
```

Putar SSH tombol

Untuk keamanan, kami merekomendasikan praktik terbaik memutar SSH kunci Anda. Biasanya, rotasi ini ditentukan sebagai bagian dari kebijakan keamanan dan diimplementasikan dalam beberapa cara otomatis. Tergantung pada tingkat keamanan, untuk komunikasi yang sangat sensitif, SSH key pair mungkin hanya digunakan sekali. Melakukan hal ini menghilangkan risiko apa pun karena kunci yang disimpan. Namun, jauh lebih umum untuk menyimpan SSH kredensial untuk jangka waktu tertentu dan menetapkan interval yang tidak menempatkan beban yang tidak semestinya pada pengguna. Interval waktu tiga bulan adalah hal biasa.

Ada dua metode yang digunakan untuk melakukan rotasi SSH kunci:

- Di konsol, Anda dapat mengunggah kunci SSH publik baru dan menghapus kunci SSH publik yang ada.
- Menggunakan API, Anda dapat memperbarui pengguna yang ada dengan menggunakan [DeleteSshPublicKey](#) API untuk menghapus kunci publik Secure Shell (SSH) pengguna dan

[ImportSshPublicKey](#) API untuk menambahkan kunci publik Secure Shell (SSH) baru ke akun pengguna.

Console

Untuk melakukan rotasi kunci di konsol

1. Buka AWS Transfer Family konsol di <https://console.aws.amazon.com/transfer/>.
2. Arahkan ke halaman Server.
3. Pilih pengenal di kolom ID Server untuk melihat halaman Detail Server.
4. Di bawah Pengguna, pilih kotak centang pengguna yang kunci SSH publiknya ingin diputar, lalu pilih Tindakan, lalu pilih Tambah kunci untuk melihat halaman Tambah kunci.

atau

Pilih nama pengguna untuk melihat halaman Detail pengguna, lalu pilih Tambahkan kunci SSH publik untuk melihat halaman Tambah kunci.

5. Masukkan kunci SSH publik baru dan pilih Tambah kunci.

Important

Format kunci SSH publik tergantung pada jenis kunci yang Anda hasilkan.

- Untuk RSA kunci, formatnya adalah `ssh-rsa string`.
- Untuk ED25519 kunci, formatnya adalah `ssh-ed25519 string`.
- Untuk ECDSA kunci, kunci dimulai dengan `ecdsa-sha2-nistp256`, `ecdsa-sha2-nistp384`, atau `ecdsa-sha2-nistp521`, tergantung pada ukuran kunci yang Anda hasilkan. String awal kemudian diikuti oleh *string*, mirip dengan jenis kunci lainnya.

Anda dikembalikan ke halaman detail Pengguna, dan kunci SSH publik baru yang baru saja Anda masukkan muncul di bagian kunci SSH publik.

6. Pilih kotak centang kunci lama yang ingin Anda hapus dan kemudian pilih Hapus.
7. Konfirmasikan operasi penghapusan dengan memasukkan kata **delete**, lalu pilih Hapus.

API

Untuk melakukan rotasi kunci menggunakan API

1. Pada sistem operasi macOS, Linux, atau Unix, buka terminal perintah.
2. Ambil SSH kunci yang ingin Anda hapus dengan memasukkan perintah berikut. Untuk menggunakan perintah ini, ganti *serverID* dengan ID server untuk server Transfer Family Anda, dan ganti *username* dengan nama pengguna Anda.

```
aws transfer describe-user --server-id='serverID' --user-name='username'
```

Perintah mengembalikan rincian tentang pengguna. Salin isi "SshPublicKeyId": bidang. Anda harus memasukkan nilai ini nanti dalam prosedur ini.

```
"SshPublicKeys": [ { "SshPublicKeyBody": "public-key", "SshPublicKeyId":  
  "keyID",  
  "DateImported": 1621969331.072 } ],
```

3. Selanjutnya, impor SSH kunci baru untuk pengguna Anda. Di perintah , masukkan perintah berikut. Untuk menggunakan perintah ini, ganti *serverID* dengan ID server untuk server Transfer Family Anda, ganti *username* dengan nama pengguna Anda, dan ganti *public-key* dengan sidik jari kunci publik baru Anda.

```
aws transfer import-ssh-public-key --server-id='serverID' --user-name='username'  
  --ssh-public-key-body='public-key'
```

Jika perintah berhasil, tidak ada output yang dikembalikan.

4. Terakhir, hapus kunci lama dengan menjalankan perintah berikut. Untuk menggunakan perintah ini, ganti *serverID* dengan ID server untuk server Transfer Family Anda, ganti *username* dengan nama pengguna Anda, dan ganti *keyID-from-step-2* dengan nilai ID kunci yang Anda salin di langkah 2 prosedur ini

```
aws transfer delete-ssh-public-key --server-id='serverID' --user-name='username'  
  --ssh-public-key-id='keyID-from-step-2'
```

5. (Opsional) Untuk mengonfirmasi bahwa kunci lama tidak ada lagi, ulangi langkah 2.

Hasilkan dan kelola PGP kunci

Anda dapat menggunakan dekripsi Pretty Good Privacy (PGP) dengan file yang diproses Transfer Family dengan alur kerja. Untuk menggunakan dekripsi dalam langkah alur kerja, Anda harus memberikan kunci. PGP

Blog AWS penyimpanan memiliki posting yang menjelaskan cara mengenkripsi dan mendekripsi file, mengenkripsi dan mendekripsi file dengan [dan](#). PGP AWS Transfer Family

Hasilkan PGP kunci

Metode yang Anda gunakan untuk menghasilkan PGP kunci tergantung pada sistem operasi Anda dan versi perangkat lunak generasi kunci yang Anda gunakan.

Jika Anda menggunakan Linux atau Unix, gunakan penginstal paket Anda untuk menginstal. gpg Tergantung pada distribusi Linux Anda, salah satu perintah berikut akan bekerja untuk Anda.

```
sudo yum install gnupg
```

```
sudo apt-get install gnupg
```

[Untuk Windows atau macOS, Anda dapat mengunduh apa yang Anda butuhkan dari https://gnupg.org/download/.](https://gnupg.org/download/)

Setelah Anda menginstal perangkat lunak generator PGP kunci Anda, Anda menjalankan `gpg --gen-key` perintah `gpg --full-gen-key` or untuk menghasilkan key pair.

Note

Jika Anda menggunakan GnuPG versi 2.3.0 atau yang lebih baru, Anda harus menjalankannya. `gpg --full-gen-key` Ketika diminta untuk jenis kunci yang akan dibuat, pilih RSA atau ECC. Namun, jika Anda memilih ECC, pastikan untuk memilih salah satu NIST atau BrainPool untuk kurva elips. Jangan memilih Curve 25519.

Algoritma yang didukung untuk pasangan kunci PGP

Algoritma berikut didukung untuk pasangan PGP kunci:

- RSA

- Elgamal
- ECC:
 - NIST
 - BrainPool

 Note

Tombol Curve25519 tidak didukung.

gpgSubperintah yang berguna

Berikut ini adalah beberapa subperintah yang berguna untukgpg:

- `gpg --help`— Perintah ini mencantumkan opsi yang tersedia dan mungkin menyertakan beberapa contoh.
- `gpg --list-keys`— Perintah ini mencantumkan detail untuk semua pasangan kunci yang telah Anda buat.
- `gpg --fingerprint`— Perintah ini mencantumkan detail untuk semua pasangan kunci Anda, termasuk sidik jari masing-masing tombol.
- `gpg --export -a user-name`— Perintah ini mengeksport bagian kunci publik dari kunci untuk *user-name* yang digunakan saat kunci dihasilkan.

Kelola PGP kunci

Untuk mengelola PGP kunci Anda, Anda harus menggunakan AWS Secrets Manager.

 Note

Nama rahasia Anda termasuk ID server Transfer Family Anda. Ini berarti Anda seharusnya sudah mengidentifikasi atau membuat server sebelum Anda dapat menyimpan informasi PGP kunci Anda AWS Secrets Manager.

Jika Anda ingin menggunakan satu kunci dan frasa sandi untuk semua pengguna Anda, Anda dapat menyimpan informasi blok PGP kunci di bawah nama rahasia `aws/transfer/server-id/pgp-`

default, di mana **server-id** ID untuk server Transfer Family Anda. Kunci default ini digunakan jika tidak ada kunci di mana **user-name** cocok dengan pengguna yang menjalankan alur kerja.

Atau, Anda dapat membuat kunci untuk pengguna tertentu. Dalam hal ini, format untuk nama rahasia adalah `aws/transfer/server-id/user-name`, di mana **user-name** cocok dengan pengguna yang menjalankan alur kerja untuk server Transfer Family.

Note

Anda dapat menyimpan maksimal 3 kunci PGP pribadi, per server Transfer Family, per pengguna.

Untuk mengkonfigurasi PGP kunci untuk digunakan dengan dekripsi

1. Bergantung pada versi GPG yang Anda gunakan, jalankan salah satu perintah berikut untuk menghasilkan PGP key pair yang tidak menggunakan algoritma enkripsi Curve 25519.
 - Jika Anda menggunakan **GnuPG** versi 2.3.0 atau yang lebih baru, jalankan perintah berikut:

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

Anda dapat memilih **RSA**, atau, jika Anda memilih **ECC**, Anda dapat memilih salah satu **NIST** atau **BrainPool** untuk kurva elips. Jika Anda `gpg --gen-key` menjalankannya, Anda membuat key pair yang menggunakan algoritma enkripsi ECC Curve 25519, yang saat ini tidak kami dukung untuk PGP kunci.

- Untuk versi **GnuPG** sebelum 2.3.0, Anda dapat menggunakan perintah berikut, karena RSA adalah jenis enkripsi default.

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

Important

Selama proses pembuatan kunci, Anda harus memberikan frasa sandi dan alamat email. Pastikan untuk mencatat nilai-nilai ini. Anda harus memberikan frasa sandi ketika Anda memasukkan detail kunci ke dalam prosedur ini AWS Secrets Manager nanti. Dan Anda

harus memberikan alamat email yang sama untuk mengekspor kunci pribadi di langkah berikutnya.

2. Jalankan perintah berikut untuk mengekspor kunci pribadi. Untuk menggunakan perintah ini, ganti *private.pgp* dengan nama file untuk menyimpan blok kunci pribadi, dan *marymajor@example.com* dengan alamat email yang Anda gunakan saat Anda membuat key pair.

```
gpg --output private.pgp --armor --export-secret-key marymajor@example.com
```

3. Gunakan AWS Secrets Manager untuk menyimpan PGP kunci Anda.
 - a. Masuk ke AWS Management Console dan buka AWS Secrets Manager konsol di <https://console.aws.amazon.com/secretsmanager/>.
 - b. Pada panel navigasi kiri, pilih Rahasia.
 - c. Pada halaman Rahasia, pilih Simpan rahasia baru.
 - d. Pada halaman Choose secret type, untuk Secret type, pilih Other type of secret.
 - e. Di bagian pasangan kunci/Nilai, pilih tab kunci/Nilai.
 - Kunci — Masukkan **PGPPrivateKey**.

 Note

Anda harus memasukkan **PGPPrivateKey** string dengan tepat: jangan menambahkan spasi sebelum atau di antara karakter.

- nilai — Tempelkan teks kunci pribadi Anda ke bidang nilai. Anda dapat menemukan teks kunci pribadi Anda dalam file (misalnya, *private.pgp*) yang Anda tentukan saat Anda mengekspor kunci Anda sebelumnya dalam prosedur ini. Kuncinya dimulai dengan -----BEGIN PGP PRIVATE KEY BLOCK----- dan diakhiri dengan -----END PGP PRIVATE KEY BLOCK-----.

 Note

Pastikan bahwa blok teks hanya berisi kunci pribadi dan tidak mengandung kunci publik juga.

- f. Pilih Tambahkan baris dan di bagian pasangan kunci/Nilai, pilih tab kunci/Nilai.

- Kunci — Masukkan **PGPPassphrase**.

Note

Anda harus memasukkan **PGPPassphrase** string dengan tepat: jangan menambahkan spasi sebelum atau di antara karakter.

- value — Masukkan kata sandi yang Anda gunakan saat membuat key pair PGP.

Choose secret type

Secret type [Info](#)

☐ Credentials for Amazon RDS database
 ☐ Credentials for Amazon DocumentDB database
 ☐ Credentials for Amazon Redshift cluster

☐ Credentials for other database
 ☒ Other type of secret
API key, OAuth token, other.

Key/value pairs [Info](#)

Key/value	Plaintext	
PGPPrivateKey	-----BEGIN PGP PRIVATE KEY BLOCK-----	Remove
PGPPassphrase	my passphrase	Remove
+ Add row		

Encryption key [Info](#)

You can encrypt using the KMS key that Secrets Manager creates or a customer managed KMS key that you create.

aws/secretsmanager [Add new key](#)

Note

Anda dapat menambahkan hingga 3 set kunci dan frasa sandi. Untuk menambahkan set kedua, tambahkan dua baris baru, dan masukkan **PGPPrivateKey2** dan **PGPPassphrase2** untuk kunci, dan tempel di kunci pribadi dan frasa sandi lainnya. Untuk menambahkan set ketiga, nilai kunci harus **PGPPrivateKey3** dan **PGPPassphrase3**.

- g. Pilih Berikutnya.

- h. Pada halaman Konfigurasi rahasia, masukkan nama dan deskripsi untuk rahasia Anda.
 - Jika Anda membuat kunci default, yaitu kunci yang dapat digunakan oleh pengguna Transfer Family, masukkan `aws/transfer/server-id/@pgp-default`. Ganti `server-id` dengan ID server yang berisi alur kerja yang memiliki langkah dekripsi.
 - Jika Anda membuat kunci untuk digunakan oleh pengguna Transfer Family tertentu, masukkan `aws/transfer/server-id/user-name`. Ganti `server-id` dengan ID server yang berisi alur kerja yang memiliki langkah dekripsi, dan ganti `user-name` dengan nama pengguna yang menjalankan alur kerja. `user-name` ini disimpan di penyedia identitas yang digunakan server Transfer Family.
- i. Pilih Berikutnya dan terima default pada halaman Konfigurasi rotasi. Lalu pilih Berikutnya.
- j. Pada halaman Review, pilih Store untuk membuat dan menyimpan rahasia.

Tangkapan layar berikut menunjukkan detail untuk pengguna **marymajor** untuk server Transfer Family tertentu. Contoh ini menunjukkan tiga kunci dan frasa sandi yang sesuai.

The screenshot shows the AWS Secrets Manager console for a secret named `/aws/transfer/s-[redacted]/marymajor`. The secret is encrypted with `aws/secretsmanager`. The secret value is displayed in a table with three rows: `PGPPrivateKey`, `PGPPassphrase`, and `PGPPrivateKey2`, each followed by a corresponding `PGPPassphrase`. The secret value is masked with a grey bar.

Secret key	Secret value
PGPPrivateKey	-----BEGIN PGP PRIVATE KEY BLOCK----- [redacted]
PGPPassphrase	mypassphrase
PGPPrivateKey2	-----BEGIN PGP PRIVATE KEY BLOCK----- [redacted]
PGPPassphrase2	mypassphrase2
PGPPrivateKey3	-----BEGIN PGP PRIVATE KEY BLOCK----- [redacted]
PGPPassphrase3	mypassphrase3

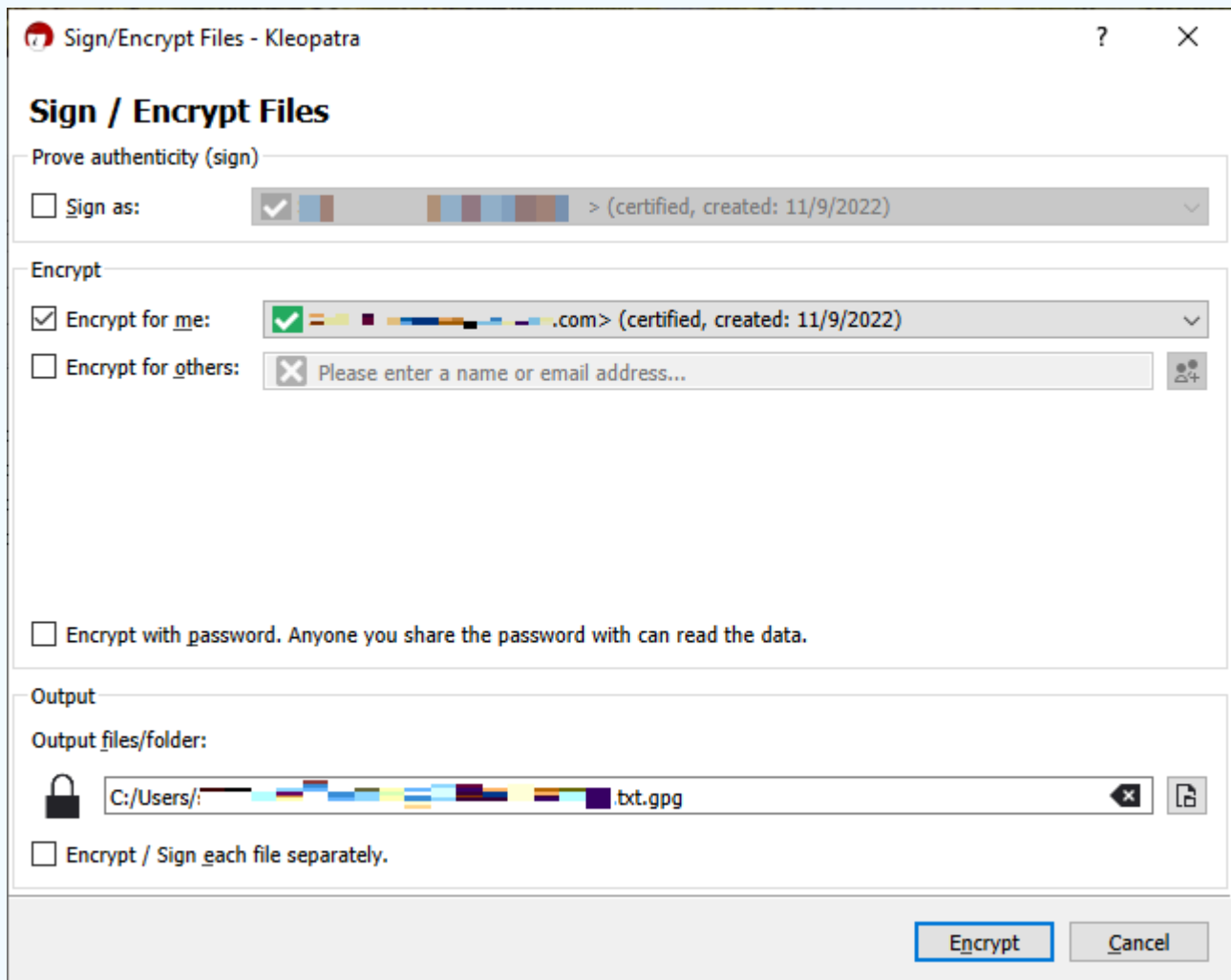
PGPKlien yang didukung

Klien berikut telah diuji dengan Transfer Family dan dapat digunakan untuk menghasilkan PGP kunci, dan untuk mengenkripsi file yang ingin Anda dekripsi dengan alur kerja.

- GPG4win + Kleopatra.

 Note

Saat Anda memilih Sign/Encrypt Files, pastikan untuk menghapus pilihan untuk Sign as: saat ini kami tidak mendukung penandatanganan untuk file terenkripsi.



- Versi GnuPG utama: 2.4, 2.3, 2.2, 2.0, dan 1.4.

Perhatikan bahwa PGP klien lain mungkin bekerja juga, tetapi hanya klien yang disebutkan di sini yang telah diuji dengan Transfer Family.

Identitas dan manajemen akses untuk AWS Transfer Family

AWS Identity and Access Management (IAM) adalah Layanan AWS yang membantu administrator mengontrol akses ke AWS sumber daya dengan aman. IAM administrator mengontrol siapa yang dapat diautentikasi (masuk) dan diberi wewenang (memiliki izin) untuk menggunakan sumber daya. AWS Transfer Family IAM adalah Layanan AWS yang dapat Anda gunakan tanpa biaya tambahan.

Topik

- [Audiens](#)
- [Mengautentikasi dengan identitas](#)
- [Mengelola akses menggunakan kebijakan](#)
- [Bagaimana AWS Transfer Family bekerja dengan IAM](#)
- [AWS Transfer Family contoh kebijakan berbasis identitas](#)
- [AWS Transfer Family contoh kebijakan berbasis tag](#)
- [Memecahkan masalah AWS Transfer Family identitas dan akses](#)

Audiens

Cara Anda menggunakan AWS Identity and Access Management (IAM) berbeda, tergantung pada pekerjaan yang Anda lakukan AWS Transfer Family.

Pengguna layanan — Jika Anda menggunakan AWS Transfer Family layanan untuk melakukan pekerjaan Anda, administrator Anda memberi Anda kredensial dan izin yang Anda butuhkan. Saat Anda menggunakan lebih banyak AWS Transfer Family fitur untuk melakukan pekerjaan Anda, Anda mungkin memerlukan izin tambahan. Memahami cara mengelola akses dapat membantu Anda meminta izin yang tepat dari administrator Anda. Jika Anda tidak dapat mengakses fitur di AWS Transfer Family, lihat [Memecahkan masalah AWS Transfer Family identitas dan akses](#).

Administrator layanan — Jika Anda bertanggung jawab atas AWS Transfer Family sumber daya di perusahaan Anda, Anda mungkin memiliki akses penuh ke AWS Transfer Family. Tugas Anda adalah menentukan AWS Transfer Family fitur dan sumber daya mana yang harus diakses pengguna layanan Anda. Anda kemudian harus mengirimkan permintaan ke IAM administrator Anda

untuk mengubah izin pengguna layanan Anda. Tinjau informasi di halaman ini untuk memahami konsep dasar IAM. Untuk mempelajari lebih lanjut tentang bagaimana perusahaan Anda dapat menggunakannya IAM AWS Transfer Family, lihat [Bagaimana AWS Transfer Family bekerja dengan IAM](#).

IAM administrator - Jika Anda seorang IAM administrator, Anda mungkin ingin mempelajari detail tentang cara menulis kebijakan untuk mengelola akses AWS Transfer Family. Untuk melihat contoh kebijakan AWS Transfer Family berbasis identitas yang dapat Anda gunakan, lihat. IAM [AWS Transfer Family contoh kebijakan berbasis identitas](#)

Mengautentikasi dengan identitas

Otentikasi adalah cara Anda masuk AWS menggunakan kredensi identitas Anda. Anda harus diautentikasi (masuk ke AWS) sebagai Pengguna root akun AWS, sebagai IAM pengguna, atau dengan mengambil peran IAM.

Anda dapat masuk AWS sebagai identitas federasi dengan menggunakan kredensial yang disediakan melalui sumber identitas. AWS IAM Identity Center Pengguna (Pusat IAM Identitas), autentikasi masuk tunggal perusahaan Anda, dan kredensial Google atau Facebook Anda adalah contoh identitas federasi. Saat Anda masuk sebagai identitas federasi, administrator Anda sebelumnya menyiapkan federasi identitas menggunakan IAM peran. Ketika Anda mengakses AWS dengan menggunakan federasi, Anda secara tidak langsung mengambil peran.

Bergantung pada jenis pengguna Anda, Anda dapat masuk ke AWS Management Console atau portal AWS akses. Untuk informasi selengkapnya tentang masuk AWS, lihat [Cara masuk ke Panduan AWS Sign-In Pengguna Anda Akun AWS](#).

Jika Anda mengakses AWS secara terprogram, AWS sediakan kit pengembangan perangkat lunak (SDK) dan antarmuka baris perintah (CLI) untuk menandatangani permintaan Anda secara kriptografis dengan menggunakan kredensial Anda. Jika Anda tidak menggunakan AWS alat, Anda harus menandatangani permintaan sendiri. Untuk informasi selengkapnya tentang menggunakan metode yang disarankan untuk menandatangani permintaan sendiri, lihat [Menandatangani AWS API permintaan](#) di Panduan IAM Pengguna.

Apa pun metode autentikasi yang digunakan, Anda mungkin diminta untuk menyediakan informasi keamanan tambahan. Misalnya, AWS merekomendasikan agar Anda menggunakan otentikasi multi-faktor (MFA) untuk meningkatkan keamanan akun Anda. Untuk mempelajari lebih lanjut, lihat [Autentikasi multi-faktor](#) di Panduan AWS IAM Identity Center Pengguna dan [Menggunakan autentikasi multi-faktor \(MFA\) AWS di](#) Panduan Pengguna. IAM

Pengguna root akun AWS

Saat Anda membuat Akun AWS, Anda mulai dengan satu identitas masuk yang memiliki akses lengkap ke semua Layanan AWS dan sumber daya di akun. Identitas ini disebut pengguna Akun AWS root dan diakses dengan masuk dengan alamat email dan kata sandi yang Anda gunakan untuk membuat akun. Kami sangat menyarankan agar Anda tidak menggunakan pengguna root untuk tugas sehari-hari. Lindungi kredensial pengguna root Anda dan gunakan kredensial tersebut untuk melakukan tugas yang hanya dapat dilakukan pengguna root. Untuk daftar lengkap tugas yang mengharuskan Anda masuk sebagai pengguna root, lihat [Tugas yang memerlukan kredensial pengguna root](#) di IAMPanduan Pengguna.

Identitas gabungan

Sebagai praktik terbaik, mewajibkan pengguna manusia, termasuk pengguna yang memerlukan akses administrator, untuk menggunakan federasi dengan penyedia identitas untuk mengakses Layanan AWS dengan menggunakan kredensial sementara.

Identitas federasi adalah pengguna dari direktori pengguna perusahaan Anda, penyedia identitas web, direktori Pusat Identitas AWS Directory Service, atau pengguna mana pun yang mengakses Layanan AWS dengan menggunakan kredensial yang disediakan melalui sumber identitas. Ketika identitas federasi mengakses Akun AWS, mereka mengambil peran, dan peran memberikan kredensial sementara.

Untuk manajemen akses terpusat, kami sarankan Anda menggunakan AWS IAM Identity Center. Anda dapat membuat pengguna dan grup di Pusat IAM Identitas, atau Anda dapat menghubungkan dan menyinkronkan ke sekumpulan pengguna dan grup di sumber identitas Anda sendiri untuk digunakan di semua aplikasi Akun AWS dan aplikasi Anda. Untuk informasi tentang Pusat IAM Identitas, lihat [Apa itu Pusat IAM Identitas?](#) dalam AWS IAM Identity Center User Guide.

Pengguna dan grup IAM

[IAMPengguna](#) adalah identitas dalam diri Anda Akun AWS yang memiliki izin khusus untuk satu orang atau aplikasi. Jika memungkinkan, sebaiknya mengandalkan kredensial sementara daripada membuat IAM pengguna yang memiliki kredensial jangka panjang seperti kata sandi dan kunci akses. Namun, jika Anda memiliki kasus penggunaan khusus yang memerlukan kredensial jangka panjang dengan IAM pengguna, kami sarankan Anda memutar kunci akses. Untuk informasi selengkapnya, lihat [Memutar kunci akses secara teratur untuk kasus penggunaan yang memerlukan kredensial jangka panjang](#) di IAMPanduan Pengguna.

[IAMGrup](#) adalah identitas yang menentukan kumpulan IAM pengguna. Anda tidak dapat masuk sebagai grup. Anda dapat menggunakan grup untuk menentukan izin bagi beberapa pengguna sekaligus. Grup mempermudah manajemen izin untuk sejumlah besar pengguna sekaligus. Misalnya, Anda dapat memiliki grup bernama IAMAdmins dan memberikan izin grup tersebut untuk mengelola sumber daya IAM.

Pengguna berbeda dari peran. Pengguna secara unik terkait dengan satu orang atau aplikasi, tetapi peran dimaksudkan untuk dapat digunakan oleh siapa pun yang membutuhkannya. Pengguna memiliki kredensial jangka panjang permanen, tetapi peran memberikan kredensial sementara. Untuk mempelajari lebih lanjut, lihat [Kapan membuat IAM pengguna \(bukan peran\)](#) di Panduan IAM Pengguna.

IAMperan

[IAMPeran](#) adalah identitas dalam diri Anda Akun AWS yang memiliki izin khusus. Ini mirip dengan IAM pengguna, tetapi tidak terkait dengan orang tertentu. Anda dapat mengambil IAM peran sementara AWS Management Console dengan [beralih peran](#). Anda dapat mengambil peran dengan memanggil AWS CLI atau AWS API operasi atau dengan menggunakan kustom URL. Untuk informasi selengkapnya tentang metode penggunaan peran, lihat [Menggunakan IAM peran](#) di Panduan IAM Pengguna.

IAMperan dengan kredensi sementara berguna dalam situasi berikut:

- Akses pengguna terfederasi – Untuk menetapkan izin ke identitas terfederasi, Anda membuat peran dan menentukan izin untuk peran tersebut. Ketika identitas terfederasi mengautentikasi, identitas tersebut terhubung dengan peran dan diberi izin yang ditentukan oleh peran. Untuk informasi tentang peran untuk federasi, lihat [Membuat peran untuk Penyedia Identitas pihak ketiga](#) di Panduan IAM Pengguna. Jika Anda menggunakan Pusat IAM Identitas, Anda mengonfigurasi set izin. Untuk mengontrol apa yang dapat diakses identitas Anda setelah diautentikasi, Pusat IAM Identitas menghubungkan izin yang disetel ke peran. Untuk informasi tentang set izin, lihat [Set izin](#) dalam Panduan Pengguna AWS IAM Identity Center .
- Izin IAM pengguna sementara — IAM Pengguna atau peran dapat mengambil IAM peran untuk sementara mengambil izin yang berbeda untuk tugas tertentu.
- Akses lintas akun — Anda dapat menggunakan IAM peran untuk memungkinkan seseorang (prinsipal tepercaya) di akun lain mengakses sumber daya di akun Anda. Peran adalah cara utama untuk memberikan akses lintas akun. Namun, dengan beberapa Layanan AWS, Anda dapat melampirkan kebijakan secara langsung ke sumber daya (alih-alih menggunakan peran sebagai proxy). Untuk mempelajari perbedaan antara peran dan kebijakan berbasis sumber daya untuk

akses lintas akun, [lihat Perbedaan IAM peran dari kebijakan berbasis sumber daya di Panduan Pengguna](#). IAM

- Akses lintas layanan — Beberapa Layanan AWS menggunakan fitur lain Layanan AWS. Misalnya, saat Anda melakukan panggilan dalam suatu layanan, biasanya layanan tersebut menjalankan aplikasi di Amazon EC2 atau menyimpan objek di Amazon S3. Sebuah layanan mungkin melakukannya menggunakan izin prinsipal yang memanggil, menggunakan peran layanan, atau peran terkait layanan.
- Sesi akses teruskan (FAS) — Saat Anda menggunakan IAM pengguna atau peran untuk melakukan tindakan AWS, Anda dianggap sebagai prinsipal. Ketika Anda menggunakan beberapa layanan, Anda mungkin melakukan sebuah tindakan yang kemudian menginisiasi tindakan lain di layanan yang berbeda. FAS menggunakan izin dari pemanggilan utama Layanan AWS, dikombinasikan dengan permintaan Layanan AWS untuk membuat permintaan ke layanan hilir. FAS permintaan hanya dibuat ketika layanan menerima permintaan yang memerlukan interaksi dengan orang lain Layanan AWS atau sumber daya untuk menyelesaikannya. Dalam hal ini, Anda harus memiliki izin untuk melakukan kedua tindakan tersebut. Untuk detail kebijakan saat membuat FAS permintaan, lihat [Meneruskan sesi akses](#).
- Peran layanan — Peran layanan adalah [IAM peran](#) yang diasumsikan layanan untuk melakukan tindakan atas nama Anda. IAM Administrator dapat membuat, memodifikasi, dan menghapus peran layanan dari dalam IAM. Untuk informasi selengkapnya, lihat [Membuat peran untuk mendelegasikan izin ke Layanan AWS](#) dalam IAM Panduan Pengguna.
- Peran terkait layanan — Peran terkait layanan adalah jenis peran layanan yang ditautkan ke peran layanan. Layanan AWS Layanan tersebut dapat menjalankan peran untuk melakukan tindakan atas nama Anda. Peran terkait layanan muncul di Anda Akun AWS dan dimiliki oleh layanan. IAM Administrator dapat melihat, tetapi tidak mengedit izin untuk peran terkait layanan.
- Aplikasi yang berjalan di Amazon EC2 — Anda dapat menggunakan IAM peran untuk mengelola kredensial sementara untuk aplikasi yang berjalan pada EC2 instance dan pembuatan AWS CLI atau AWS API permintaan. Ini lebih baik untuk menyimpan kunci akses dalam EC2 instance. Untuk menetapkan AWS peran ke EC2 instance dan membuatnya tersedia untuk semua aplikasinya, Anda membuat profil instance yang dilampirkan ke instance. Profil instance berisi peran dan memungkinkan program yang berjalan pada EC2 instance untuk mendapatkan kredensial sementara. Untuk informasi selengkapnya, lihat [Menggunakan IAM peran untuk memberikan izin ke aplikasi yang berjalan di EC2 instans Amazon](#) di IAM Panduan Pengguna.

Untuk mempelajari apakah akan menggunakan IAM peran atau IAM pengguna, lihat [Kapan membuat IAM peran \(bukan pengguna\)](#) di Panduan IAM Pengguna.

Mengelola akses menggunakan kebijakan

Anda mengontrol akses AWS dengan membuat kebijakan dan melampirkannya ke AWS identitas atau sumber daya. Kebijakan adalah objek AWS yang, ketika dikaitkan dengan identitas atau sumber daya, menentukan izinnya. AWS mengevaluasi kebijakan ini ketika prinsipal (pengguna, pengguna root, atau sesi peran) membuat permintaan. Izin dalam kebijakan menentukan apakah permintaan diizinkan atau ditolak. Sebagian besar kebijakan disimpan AWS sebagai JSON dokumen. Untuk informasi selengkapnya tentang struktur dan isi dokumen JSON kebijakan, lihat [Ringkasan JSON kebijakan](#) di Panduan IAM Pengguna.

Administrator dapat menggunakan AWS JSON kebijakan untuk menentukan siapa yang memiliki akses ke apa. Yaitu, principal dapat melakukan tindakan pada suatu sumber daya, dan dalam suatu syarat.

Secara default, pengguna dan peran tidak memiliki izin. Untuk memberikan izin kepada pengguna untuk melakukan tindakan pada sumber daya yang mereka butuhkan, IAM administrator dapat membuat IAM kebijakan. Administrator kemudian dapat menambahkan IAM kebijakan ke peran, dan pengguna dapat mengambil peran.

IAMkebijakan menentukan izin untuk tindakan terlepas dari metode yang Anda gunakan untuk melakukan operasi. Misalnya, anggaplah Anda memiliki kebijakan yang mengizinkan tindakan `iam:GetRole`. Pengguna dengan kebijakan itu bisa mendapatkan informasi peran dari AWS Management Console, AWS CLI, atau AWS API.

Kebijakan berbasis identitas

Kebijakan berbasis identitas adalah dokumen kebijakan JSON izin yang dapat dilampirkan ke identitas, seperti pengguna, grup IAM pengguna, atau peran. Kebijakan ini mengontrol jenis tindakan yang dapat dilakukan oleh pengguna dan peran, di sumber daya mana, dan berdasarkan kondisi seperti apa. Untuk mempelajari cara membuat kebijakan berbasis identitas, lihat [Membuat IAM kebijakan di Panduan](#) Pengguna. IAM

Kebijakan berbasis identitas dapat dikategorikan lebih lanjut sebagai kebijakan inline atau kebijakan yang dikelola. Kebijakan inline disematkan langsung ke satu pengguna, grup, atau peran. Kebijakan terkelola adalah kebijakan mandiri yang dapat Anda lampirkan ke beberapa pengguna, grup, dan peran dalam. Akun AWS Kebijakan AWS terkelola mencakup kebijakan terkelola dan kebijakan yang dikelola pelanggan. Untuk mempelajari cara memilih antara kebijakan terkelola atau kebijakan sebaris, lihat [Memilih antara kebijakan terkelola dan kebijakan sebaris](#) di IAMPanduan Pengguna.

Kebijakan berbasis sumber daya

Kebijakan berbasis sumber daya adalah dokumen JSON kebijakan yang Anda lampirkan ke sumber daya. Contoh kebijakan berbasis sumber daya adalah kebijakan kepercayaan IAM peran dan kebijakan bucket Amazon S3. Dalam layanan yang mendukung kebijakan berbasis sumber daya, administrator layanan dapat menggunakannya untuk mengontrol akses ke sumber daya tertentu. Untuk sumber daya tempat kebijakan dilampirkan, kebijakan menentukan tindakan apa yang dapat dilakukan oleh prinsipal tertentu pada sumber daya tersebut dan dalam kondisi apa. Anda harus [menentukan prinsipal](#) dalam kebijakan berbasis sumber daya. Prinsipal dapat mencakup akun, pengguna, peran, pengguna federasi, atau Layanan AWS

Kebijakan berbasis sumber daya merupakan kebijakan inline yang terletak di layanan tersebut. Anda tidak dapat menggunakan kebijakan AWS terkelola IAM dalam kebijakan berbasis sumber daya.

Daftar kontrol akses (ACLs)

Access control lists (ACLs) mengontrol prinsipal mana (anggota akun, pengguna, atau peran) yang memiliki izin untuk mengakses sumber daya. ACLs mirip dengan kebijakan berbasis sumber daya, meskipun mereka tidak menggunakan format dokumen kebijakan. JSON

Amazon S3, AWS WAF, dan Amazon VPC adalah contoh layanan yang mendukung ACLs. Untuk mempelajari selengkapnya ACLs, lihat [Ikhtisar daftar kontrol akses \(ACL\)](#) di Panduan Pengembangan Layanan Penyimpanan Sederhana Amazon.

Jenis-jenis kebijakan lain

AWS mendukung jenis kebijakan tambahan yang kurang umum. Jenis-jenis kebijakan ini dapat mengatur izin maksimum yang diberikan kepada Anda oleh jenis kebijakan yang lebih umum.

- **Batas izin** — Batas izin adalah fitur lanjutan tempat Anda menetapkan izin maksimum yang dapat diberikan oleh kebijakan berbasis identitas kepada entitas (pengguna atau peran). IAM IAM Anda dapat menetapkan batasan izin untuk suatu entitas. Izin yang dihasilkan adalah perpotongan antara kebijakan berbasis identitas milik entitas dan batasan izinnya. Kebijakan berbasis sumber daya yang menentukan pengguna atau peran dalam bidang `Principal` tidak dibatasi oleh batasan izin. Penolakan eksplisit dalam salah satu kebijakan ini akan menggantikan pemberian izin. Untuk informasi selengkapnya tentang batas izin, lihat [Batas izin untuk IAM entitas](#) di IAMPanduan Pengguna.
- **Kebijakan kontrol layanan (SCPs)** — SCPs adalah JSON kebijakan yang menentukan izin maksimum untuk organisasi atau unit organisasi (OU) di AWS Organizations. AWS Organizations

adalah layanan untuk mengelompokkan dan mengelola secara terpusat beberapa Akun AWS yang dimiliki bisnis Anda. Jika Anda mengaktifkan semua fitur dalam organisasi, Anda dapat menerapkan kebijakan kontrol layanan (SCPs) ke salah satu atau semua akun Anda. SCP Membatasi izin untuk entitas di akun anggota, termasuk masing-masing Pengguna root akun AWS. Untuk informasi selengkapnya tentang Organizations dan SCPs, lihat [Cara SCPs kerja](#) di Panduan AWS Organizations Pengguna.

- Kebijakan sesi – Kebijakan sesi adalah kebijakan lanjutan yang Anda berikan sebagai parameter ketika Anda membuat sesi sementara secara programatis untuk peran atau pengguna terfederasi. Izin sesi yang dihasilkan adalah perpotongan antara kebijakan berbasis identitas pengguna atau peran dan kebijakan sesi. Izin juga bisa datang dari kebijakan berbasis sumber daya. Penolakan secara tegas dalam salah satu kebijakan ini membatalkan izin. Untuk informasi selengkapnya, lihat [Kebijakan sesi](#) di Panduan IAM Pengguna.

Berbagai jenis kebijakan

Ketika beberapa jenis kebijakan berlaku pada suatu permintaan, izin yang dihasilkan lebih rumit untuk dipahami. Untuk mempelajari cara AWS menentukan apakah akan mengizinkan permintaan saat beberapa jenis kebijakan terlibat, lihat [Logika evaluasi kebijakan](#) di Panduan IAM Pengguna.

Bagaimana AWS Transfer Family bekerja dengan IAM

Sebelum Anda menggunakan AWS Identity and Access Management (IAM) untuk mengelola akses ke AWS Transfer Family, Anda harus memahami IAM fitur apa yang tersedia untuk digunakan AWS Transfer Family. Untuk mendapatkan tampilan tingkat tinggi tentang cara AWS Transfer Family dan AWS layanan lain bekerja dengan IAM, lihat [AWS layanan yang bekerja dengan IAM](#) dalam Panduan IAM Pengguna.

Topik

- [Kebijakan berbasis identitas AWS Transfer Family](#)
- [AWS Transfer Family Kebijakan berbasis sumber daya](#)
- [Otorisasi berdasarkan tanda AWS Transfer Family](#)
- [AWS Transfer Family IAM peran](#)

Kebijakan berbasis identitas AWS Transfer Family

Dengan kebijakan IAM berbasis identitas, Anda dapat menentukan tindakan dan sumber daya yang diizinkan atau ditolak serta kondisi di mana tindakan diizinkan atau ditolak. AWS Transfer Family mendukung tindakan, sumber daya, dan kunci kondisi tertentu. Untuk mempelajari semua elemen yang Anda gunakan dalam JSON kebijakan, lihat [referensi elemen IAM JSON kebijakan](#) di Panduan AWS Identity and Access Management Pengguna.

Tindakan

Administrator dapat menggunakan AWS JSON kebijakan untuk menentukan siapa yang memiliki akses ke apa. Yaitu, principal dapat melakukan tindakan pada suatu sumber daya, dan dalam suatu syarat.

ActionElemen JSON kebijakan menjelaskan tindakan yang dapat Anda gunakan untuk mengizinkan atau menolak akses dalam kebijakan. Tindakan kebijakan biasanya memiliki nama yang sama dengan AWS API operasi terkait. Ada beberapa pengecualian, seperti tindakan khusus izin yang tidak memiliki operasi yang cocok. API Ada juga beberapa operasi yang memerlukan beberapa tindakan dalam suatu kebijakan. Tindakan tambahan ini disebut tindakan dependen.

Menyertakan tindakan dalam kebijakan untuk memberikan izin untuk melakukan operasi terkait.

Tindakan kebijakan AWS Transfer Family menggunakan awalan berikut sebelum tindakan: `transfer:`. Misalnya, untuk memberikan izin kepada seseorang untuk membuat server, dengan `CreateServer` API operasi Transfer Family, Anda menyertakan `transfer>CreateServer` tindakan tersebut dalam kebijakan mereka. Pernyataan kebijakan harus menyertakan elemen `Action` atau `NotAction`. AWS Transfer Family menentukan serangkaian tindakan sendiri yang menjelaskan tugas yang dapat Anda lakukan dengan layanan ini.

Untuk menetapkan beberapa tindakan dalam satu pernyataan, pisahkan dengan koma seperti berikut.

```
"Action": [  
    "transfer:action1",  
    "transfer:action2"
```

Anda juga dapat menentukan beberapa tindakan menggunakan wildcard (*). Misalnya, untuk menentukan semua tindakan yang dimulai dengan kata `Describe`, sertakan tindakan berikut.

```
"Action": "transfer:Describe*"
```

Untuk melihat daftar AWS Transfer Family tindakan, lihat [Tindakan yang ditentukan oleh AWS Transfer Family](#) dalam Referensi Otorisasi Layanan.

Sumber daya

Administrator dapat menggunakan AWS JSON kebijakan untuk menentukan siapa yang memiliki akses ke apa. Yaitu, principal dapat melakukan tindakan pada suatu sumber daya, dan dalam suatu syarat.

Elemen Resource JSON kebijakan menentukan objek atau objek yang tindakan tersebut berlaku. Pernyataan harus menyertakan elemen Resource atau NotResource. Sebagai praktik terbaik, tentukan sumber daya menggunakan [Amazon Resource Name \(ARN\)](#). Anda dapat melakukan ini untuk tindakan yang mendukung jenis sumber daya tertentu, yang dikenal sebagai izin tingkat sumber daya.

Untuk tindakan yang tidak mendukung izin di tingkat sumber daya, misalnya operasi pencantuman, gunakan wildcard (*) untuk menunjukkan bahwa pernyataan tersebut berlaku untuk semua sumber daya.

```
"Resource": ""
```

Sumber daya server Transfer Family memiliki yang berikut iniARN.

```
arn:aws:transfer:${Region}:${Account}:server/${ServerId}
```

Misalnya, untuk menentukan server s-01234567890abcdef Transfer Family dalam pernyataan Anda, gunakan yang berikut iniARN.

```
"Resource": "arn:aws:transfer:us-east-1:123456789012:server/s-01234567890abcdef"
```

Untuk informasi selengkapnya tentang formatARNs, lihat [Amazon Resource Names \(ARNs\)](#) di Referensi Otorisasi Layanan, atau [IAMARNs](#) di Panduan IAM Pengguna.

Untuk menentukan semua instance milik akun tertentu, gunakan wildcard (*).

```
"Resource": "arn:aws:transfer:us-east-1:123456789012:server/*"
```

Beberapa AWS Transfer Family tindakan dilakukan pada beberapa sumber daya, seperti yang digunakan dalam IAM kebijakan. Dalam kasus tersebut, Anda harus menggunakan wildcard (*).

```
"Resource": "arn:aws:transfer:*:123456789012:server/*"
```

Dalam beberapa kasus, Anda perlu menentukan lebih dari satu jenis sumber daya, misalnya, jika Anda membuat kebijakan yang memungkinkan akses ke server dan pengguna Transfer Family. Untuk menentukan beberapa sumber daya dalam satu pernyataan, pisahkan ARNs dengan koma.

```
"Resource": [  
    "resource1",  
    "resource2"  
]
```

Untuk melihat daftar AWS Transfer Family sumber daya, lihat [Jenis sumber daya yang ditentukan oleh AWS Transfer Family](#) dalam Referensi Otorisasi Layanan.

Kunci syarat

Administrator dapat menggunakan AWS JSON kebijakan untuk menentukan siapa yang memiliki akses ke apa. Yaitu, di mana utama dapat melakukan tindakan pada sumber daya, dan dalam kondisi apa.

Elemen Condition (atau blok Condition) akan memungkinkan Anda menentukan kondisi yang menjadi dasar suatu pernyataan berlaku. Elemen Condition bersifat opsional. Anda dapat membuat ekspresi bersyarat yang menggunakan [operator kondisi](#), misalnya sama dengan atau kurang dari, untuk mencocokkan kondisi dalam kebijakan dengan nilai-nilai yang diminta.

Jika Anda menentukan beberapa elemen Condition dalam sebuah pernyataan, atau beberapa kunci dalam elemen Condition tunggal, maka AWS akan mengevaluasinya menggunakan operasi AND logis. Jika Anda menentukan beberapa nilai untuk satu kunci kondisi, AWS mengevaluasi kondisi menggunakan OR operasi logis. Semua kondisi harus dipenuhi sebelum izin pernyataan diberikan.

Anda juga dapat menggunakan variabel placeholder saat menentukan kondisi. Misalnya, Anda dapat memberikan izin IAM pengguna untuk mengakses sumber daya hanya jika ditandai dengan nama IAM pengguna mereka. Untuk informasi selengkapnya, lihat [elemen IAM kebijakan: variabel dan tag](#) di Panduan IAM Pengguna.

AWS mendukung kunci kondisi global dan kunci kondisi khusus layanan. Untuk melihat semua kunci kondisi AWS global, lihat [kunci konteks kondisi AWS global](#) di Panduan IAM Pengguna.

AWS Transfer Family mendefinisikan kumpulan kunci kondisinya sendiri dan juga mendukung penggunaan beberapa kunci kondisi global. Untuk melihat daftar kunci AWS Transfer Family kondisi, lihat [Kunci kondisi untuk AWS Transfer Family](#) dalam Referensi Otorisasi Layanan.

Contoh

Untuk melihat contoh kebijakan AWS Transfer Family berbasis identitas, lihat. [AWS Transfer Family contoh kebijakan berbasis identitas](#)

AWS Transfer Family Kebijakan berbasis sumber daya

Kebijakan berbasis sumber daya adalah dokumen JSON kebijakan yang menentukan tindakan apa yang dapat dilakukan oleh prinsipal tertentu pada AWS Transfer Family sumber daya dan dalam kondisi apa. Amazon S3 mendukung kebijakan izin berbasis sumber daya untuk Amazon S3 *buckets*. Kebijakan berbasis sumber daya memungkinkan Anda memberikan izin penggunaan ke akun lain berdasarkan per sumber daya. Anda juga dapat menggunakan kebijakan berbasis sumber daya untuk mengizinkan AWS layanan mengakses Amazon S3 *buckets*.

Untuk mengaktifkan akses lintas akun, Anda dapat menentukan seluruh akun atau IAM entitas di akun lain sebagai [prinsipal dalam kebijakan berbasis sumber daya](#). Menambahkan prinsipal akun silang ke kebijakan berbasis sumber daya hanya setengah dari membangun hubungan kepercayaan. Ketika prinsipal dan sumber daya berada di AWS akun yang berbeda, Anda juga harus memberikan izin entitas utama untuk mengakses sumber daya. Berikan izin dengan melampirkan kebijakan berbasis identitas ke entitas tersebut. Namun, jika kebijakan berbasis sumber daya memberikan akses ke prinsipal dalam akun yang sama, tidak diperlukan kebijakan berbasis identitas tambahan. Untuk informasi selengkapnya, [lihat Perbedaan IAM peran dari kebijakan berbasis sumber daya](#) di Panduan Pengguna.AWS Identity and Access Management

Layanan Amazon S3 hanya mendukung satu jenis kebijakan berbasis sumber daya yang disebut a *bucket* kebijakan, yang dilampirkan pada *bucket*. Kebijakan ini menentukan entitas utama (akun, pengguna, peran, dan pengguna gabungan) yang dapat melakukan tindakan pada objek.

Contoh

Untuk melihat contoh kebijakan AWS Transfer Family berbasis sumber daya, lihat. [AWS Transfer Family contoh kebijakan berbasis tag](#)

Otorisasi berdasarkan tanda AWS Transfer Family

Anda dapat melampirkan tag ke AWS Transfer Family sumber daya atau meneruskan tag dalam permintaan AWS Transfer Family. Untuk mengendalikan akses berdasarkan tag, berikan informasi tentang tag di [elemen kondisi](#) dari kebijakan menggunakan kunci kondisi `transfer:ResourceTag/key-name`, `aws:RequestTag/key-name`, atau `aws:TagKeys`. Untuk informasi tentang cara menggunakan tag untuk mengontrol akses ke AWS Transfer Family sumber daya, lihat [AWS Transfer Family contoh kebijakan berbasis tag](#).

AWS Transfer Family IAMperan

[IAMPeran](#) adalah entitas dalam AWS akun Anda yang memiliki izin khusus.

Menggunakan kredensi sementara dengan AWS Transfer Family

Anda dapat menggunakan kredensi sementara untuk masuk dengan federasi, mengambil IAM peran, atau untuk mengambil peran lintas akun. Anda memperoleh kredensial keamanan sementara dengan memanggil AWS STS API operasi seperti [AssumeRole](#) atau [GetFederationToken](#).

AWS Transfer Family mendukung menggunakan kredensial sementara.

AWS Transfer Family contoh kebijakan berbasis identitas

Secara default, IAM pengguna dan peran tidak memiliki izin untuk membuat atau memodifikasi AWS Transfer Family sumber daya. Mereka juga tidak dapat melakukan tugas menggunakan AWS Management Console, AWS CLI, atau AWS API. IAMAdministrator harus membuat IAM kebijakan yang memberikan izin kepada pengguna dan peran untuk melakukan API operasi tertentu pada sumber daya tertentu yang mereka butuhkan. Administrator kemudian harus melampirkan kebijakan tersebut ke IAM pengguna atau grup yang memerlukan izin tersebut.

Untuk mempelajari cara membuat kebijakan IAM berbasis identitas menggunakan contoh dokumen JSON kebijakan ini, lihat [Membuat kebijakan pada JSON tab di Panduan Pengguna](#). AWS Identity and Access Management

Topik

- [Praktik terbaik kebijakan](#)
- [Menggunakan konsol AWS Transfer Family](#)
- [Mengizinkan pengguna melihat izin mereka sendiri](#)

Praktik terbaik kebijakan

Kebijakan berbasis identitas menentukan apakah seseorang dapat membuat, mengakses, atau menghapus AWS Transfer Family sumber daya di akun Anda. Tindakan ini membuat Akun AWS Anda dikenai biaya. Ketika Anda membuat atau mengedit kebijakan berbasis identitas, ikuti panduan dan rekomendasi ini:

- Mulailah dengan kebijakan AWS terkelola dan beralih ke izin hak istimewa paling sedikit — Untuk mulai memberikan izin kepada pengguna dan beban kerja Anda, gunakan kebijakan AWS terkelola yang memberikan izin untuk banyak kasus penggunaan umum. Mereka tersedia di Anda Akun AWS. Kami menyarankan Anda mengurangi izin lebih lanjut dengan menentukan kebijakan yang dikelola AWS pelanggan yang khusus untuk kasus penggunaan Anda. Untuk informasi selengkapnya, lihat [kebijakan AWSAWS terkelola](#) atau [kebijakan terkelola untuk fungsi pekerjaan](#) di Panduan IAM Pengguna.
- Menerapkan izin hak istimewa paling sedikit — Saat Anda menetapkan izin dengan IAM kebijakan, berikan hanya izin yang diperlukan untuk melakukan tugas. Anda melakukannya dengan mendefinisikan tindakan yang dapat diambil pada sumber daya tertentu dalam kondisi tertentu, yang juga dikenal sebagai izin dengan hak akses paling rendah. Untuk informasi selengkapnya tentang penggunaan IAM untuk menerapkan izin, lihat [Kebijakan dan izin IAM di IAM](#) Panduan Pengguna.
- Gunakan ketentuan dalam IAM kebijakan untuk membatasi akses lebih lanjut — Anda dapat menambahkan kondisi ke kebijakan Anda untuk membatasi akses ke tindakan dan sumber daya. Misalnya, Anda dapat menulis kondisi kebijakan untuk menentukan bahwa semua permintaan harus dikirim menggunakan SSL. Anda juga dapat menggunakan ketentuan untuk memberikan akses ke tindakan layanan jika digunakan melalui yang spesifik Layanan AWS, seperti AWS CloudFormation. Untuk informasi selengkapnya, lihat [elemen IAM JSON kebijakan: Kondisi](#) dalam Panduan IAM Pengguna.
- Gunakan IAM Access Analyzer untuk memvalidasi IAM kebijakan Anda guna memastikan izin yang aman dan fungsional — IAM Access Analyzer memvalidasi kebijakan baru dan yang sudah ada sehingga kebijakan mematuhi bahasa IAM kebijakan () JSON dan praktik terbaik. IAM IAMAccess Analyzer menyediakan lebih dari 100 pemeriksaan kebijakan dan rekomendasi yang dapat ditindaklanjuti untuk membantu Anda membuat kebijakan yang aman dan fungsional. Untuk informasi selengkapnya, lihat [Validasi kebijakan IAM Access Analyzer](#) di IAMPanduan Pengguna.
- Memerlukan otentikasi multi-faktor (MFA) - Jika Anda memiliki skenario yang mengharuskan IAM pengguna atau pengguna root di Anda Akun AWS, aktifkan MFA untuk keamanan tambahan. Untuk meminta MFA kapan API operasi dipanggil, tambahkan MFA kondisi ke kebijakan

Anda. Untuk informasi selengkapnya, lihat [Mengonfigurasi API akses MFA yang dilindungi](#) di IAMPanduan Pengguna.

Untuk informasi selengkapnya tentang praktik terbaik diIAM, lihat [Praktik terbaik keamanan IAM di](#) Panduan IAM Pengguna.

Menggunakan konsol AWS Transfer Family

Untuk mengakses AWS Transfer Family konsol, Anda harus memiliki set izin minimum. Izin ini harus memungkinkan Anda untuk membuat daftar dan melihat detail tentang AWS Transfer Family sumber daya di AWS akun Anda. Jika Anda membuat kebijakan berbasis identitas yang lebih ketat daripada izin minimum yang diperlukan, konsol tidak akan berfungsi sebagaimana dimaksud untuk entitas (IAMpengguna atau peran) dengan kebijakan tersebut. Untuk informasi selengkapnya, lihat [Menambahkan izin ke pengguna](#) di Panduan AWS Identity and Access Management Pengguna.

Anda tidak perlu mengizinkan izin konsol minimum untuk pengguna yang melakukan panggilan hanya ke AWS CLI atau. AWS API Sebagai gantinya, izinkan akses hanya ke tindakan yang cocok dengan API operasi yang Anda coba lakukan.

Mengizinkan pengguna melihat izin mereka sendiri

Contoh ini menunjukkan cara Anda membuat kebijakan yang memungkinkan IAM pengguna melihat kebijakan sebaris dan terkelola yang dilampirkan pada identitas pengguna mereka. Kebijakan ini mencakup izin untuk menyelesaikan tindakan ini di konsol atau secara terprogram menggunakan atau. AWS CLI AWS API

```
{
  "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 contoh kebijakan berbasis tag

Berikut ini adalah contoh cara mengontrol akses ke AWS Transfer Family sumber daya berdasarkan tag.

Menggunakan tag untuk mengontrol akses ke AWS Transfer Family sumber daya

Ketentuan dalam IAM kebijakan adalah bagian dari sintaks yang Anda gunakan untuk menentukan izin ke AWS Transfer Family sumber daya. Anda dapat mengontrol akses ke AWS Transfer Family sumber daya (seperti pengguna, server, peran, dan entitas lain) berdasarkan tag pada sumber daya tersebut. Tag adalah pasangan nilai kunci. Untuk informasi selengkapnya tentang menandai sumber daya, lihat [Menandai AWS sumber daya](#) di Referensi Umum AWS

Di AWS Transfer Family, sumber daya dapat memiliki tag, dan beberapa tindakan dapat menyertakan tag. Saat membuat IAM kebijakan, Anda dapat menggunakan kunci kondisi tag untuk mengontrol hal-hal berikut:

- Pengguna mana yang dapat melakukan tindakan pada AWS Transfer Family sumber daya, berdasarkan tag yang dimiliki sumber daya.
- Tag apa yang dapat diteruskan dalam permintaan tindakan.
- Apakah kunci tag tertentu dapat digunakan dalam permintaan.

Dengan menggunakan kontrol akses berbasis tag, Anda dapat menerapkan kontrol yang lebih baik daripada di level API. Anda juga dapat menerapkan kontrol yang lebih dinamis daripada dengan menggunakan kontrol akses berbasis sumber daya. Anda dapat membuat IAM kebijakan yang mengizinkan atau menolak operasi berdasarkan tag yang disediakan dalam permintaan (tag permintaan). Anda juga dapat membuat IAM kebijakan berdasarkan tag pada sumber daya yang sedang dioperasikan (tag sumber daya). Secara umum, tag sumber daya adalah untuk tag yang sudah ada di sumber daya, tag permintaan digunakan saat Anda menambahkan tag atau menghapus tag dari sumber daya.

Untuk sintaks dan semantik lengkap kunci kondisi tag, lihat [Mengontrol akses ke AWS sumber daya menggunakan tag sumber daya di Panduan Pengguna](#). Untuk detail tentang menentukan IAM kebijakan dengan API Gateway, lihat [Mengontrol akses ke izin API dengan IAM izin di Panduan Pengembang API Gateway](#).

Contoh 1: Tolak tindakan berdasarkan tag sumber daya

Anda dapat menolak tindakan yang akan dilakukan pada sumber daya berdasarkan tag. Contoh kebijakan berikut

menyangkal `TagResource`, `UntagResource`, `StartServer`, `StopServer`, `DescribeServer`, dan `DescribeUser` operasi jika sumber daya pengguna atau server ditandai dengan kunci `stage` dan nilainya `prod`.

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

```

    }
  ]
}

```

Contoh 2: Izinkan tindakan berdasarkan tag sumber daya

Anda dapat mengizinkan tindakan dilakukan pada sumber daya berdasarkan tag. Contoh kebijakan berikut

memungkinkan `TagResource`, `UntagResource`, `StartServer`, `StopServer`, `DescribeServer`, dan `DescribeUser` operasi jika sumber daya pengguna atau server ditandai dengan kunci `stage` dan nilainya `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"
        }
      }
    }
  ]
}

```

Contoh 3: Tolak pembuatan pengguna atau server berdasarkan tag permintaan

Contoh kebijakan berikut berisi dua pernyataan. Pernyataan pertama menyangkal `CreateServer` operasi pada semua sumber daya jika kunci pusat biaya untuk tag tidak memiliki nilai.

Pernyataan kedua menyangkal `CreateServer` operasi jika kunci pusat biaya untuk tag berisi nilai lain selain 1, 2 atau 3.

Note

Kebijakan ini memungkinkan pembuatan atau penghapusan sumber daya yang berisi kunci yang dipanggil `costcenter` dan nilai `1,2`, atau `3`.

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

Memecahkan masalah AWS Transfer Family identitas dan akses

Gunakan informasi berikut untuk membantu Anda mendiagnosis dan memperbaiki masalah umum yang mungkin Anda temui saat bekerja dengan AWS Transfer Family dan IAM.

Topik

- [Saya tidak berwenang untuk melakukan tindakan di AWS Transfer Family](#)
- [Saya tidak berwenang untuk melakukan iam: PassRole](#)
- [Saya ingin mengizinkan orang di luar AWS akun saya untuk mengakses AWS Transfer Family sumber daya saya](#)

Saya tidak berwenang untuk melakukan tindakan di AWS Transfer Family

Jika AWS Management Console memberitahu Anda bahwa Anda tidak berwenang untuk melakukan tindakan, maka Anda harus menghubungi administrator Anda untuk bantuan. Administrator Anda adalah orang yang memberi Anda kredensial masuk.

Contoh kesalahan berikut terjadi ketika `mateojackson` IAM pengguna mencoba menggunakan konsol untuk melihat detail tentang `widget` tetapi tidak memiliki `transfer:GetWidget` izin kerja.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
transfer:GetWidget on resource: my-example-widget
```

Dalam hal ini, Mateo meminta administratornya untuk memperbarui kebijakannya untuk mengizinkan dia mengakses sumber daya `my-example-widget` menggunakan tindakan `transfer:GetWidget`.

Saya tidak berwenang untuk melakukan iam: PassRole

Jika Anda menerima kesalahan yang tidak diizinkan untuk melakukan `iam:PassRole` tindakan, kebijakan Anda harus diperbarui agar Anda dapat meneruskan peran AWS Transfer Family.

Beberapa Layanan AWS memungkinkan Anda untuk meneruskan peran yang ada ke layanan tersebut alih-alih membuat peran layanan baru atau peran terkait layanan. Untuk melakukannya, Anda harus memiliki izin untuk meneruskan peran ke layanan.

Contoh kesalahan berikut terjadi ketika IAM pengguna bernama `marymajor` mencoba menggunakan konsol untuk melakukan tindakan di AWS Transfer Family. Namun, tindakan tersebut memerlukan

layanan untuk mendapatkan izin yang diberikan oleh peran layanan. Mary tidak memiliki izin untuk meneruskan peran tersebut pada layanan.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
iam:PassRole
```

Dalam kasus ini, kebijakan Mary harus diperbarui agar dia mendapatkan izin untuk melakukan tindakan `iam:PassRole` tersebut.

Jika Anda memerlukan bantuan, hubungi AWS administrator Anda. Administrator Anda adalah orang yang memberi Anda kredensial masuk.

Kebijakan contoh berikut berisi izin untuk meneruskan peran AWS Transfer Family.

```
{
  "Version": "2012-10-17",
  "Statement": [
    { "Action": "iam:PassRole",
      "Resource": "arn:aws:iam::123456789012:role/*",
      "Effect": "Allow"
    }
  ]
}
```

Saya ingin mengizinkan orang di luar AWS akun saya untuk mengakses AWS Transfer Family sumber daya saya

Anda dapat membuat peran yang dapat digunakan pengguna di akun lain atau orang-orang di luar organisasi Anda untuk mengakses sumber daya Anda. Anda dapat menentukan siapa saja yang dipercaya untuk mengambil peran tersebut. Untuk layanan yang mendukung kebijakan berbasis sumber daya atau daftar kontrol akses (ACLs), Anda dapat menggunakan kebijakan tersebut untuk memberi orang akses ke sumber daya Anda.

Untuk mempelajari selengkapnya, periksa referensi berikut:

- Untuk mempelajari apakah AWS Transfer Family mendukung fitur-fitur ini, lihat [Bagaimana AWS Transfer Family bekerja dengan IAM](#).
- Untuk mempelajari cara menyediakan akses ke sumber daya Anda di seluruh sumber daya Akun AWS yang Anda miliki, lihat [Menyediakan akses ke IAM pengguna lain Akun AWS yang Anda miliki](#) di Panduan IAM Pengguna.

- Untuk mempelajari cara menyediakan akses ke sumber daya Anda kepada pihak ketiga Akun AWS, lihat [Menyediakan akses yang Akun AWS dimiliki oleh pihak ketiga](#) dalam Panduan IAM Pengguna.
- Untuk mempelajari cara menyediakan akses melalui federasi identitas, lihat [Menyediakan akses ke pengguna yang diautentikasi secara eksternal \(federasi identitas\) di Panduan Pengguna](#). IAM
- Untuk mempelajari perbedaan antara menggunakan peran dan kebijakan berbasis sumber daya untuk akses lintas akun, [lihat Perbedaan IAM peran dari kebijakan berbasis sumber daya di Panduan Pengguna](#). IAM

Validasi kepatuhan untuk AWS Transfer Family

Auditor pihak ketiga menilai keamanan dan kepatuhan AWS Transfer Family sebagai bagian dari beberapa program AWS kepatuhan. Ini termasuk SOC, PCI, HIPAA, dan lain-lain. Untuk daftar lengkapnya, lihat [AWS Layanan dalam Lingkup menurut Program Kepatuhan](#).

Untuk daftar AWS layanan dalam lingkup program kepatuhan tertentu, lihat [AWS layanan dalam lingkup oleh program kepatuhan](#). Untuk informasi umum, lihat [Program kepatuhan AWS](#).

Anda dapat mengunduh laporan audit pihak ketiga menggunakan AWS Artifact. Untuk informasi selengkapnya, lihat [Mengunduh laporan di AWS Artifact](#).

Tanggung jawab kepatuhan Anda saat menggunakan AWS Transfer Family ditentukan oleh sensitivitas data Anda, tujuan kepatuhan perusahaan Anda, dan hukum dan peraturan yang berlaku. AWS menyediakan sumber daya berikut untuk membantu kepatuhan:

- [Panduan memulai cepat keamanan dan kepatuhan — Panduan](#) penerapan ini membahas pertimbangan arsitektur dan memberikan langkah-langkah untuk menerapkan lingkungan dasar yang berfokus pada keamanan dan kepatuhan. AWS
- [Arsitektur untuk whitepaper HIPAA keamanan dan kepatuhan — Whitepaper](#) ini menjelaskan bagaimana perusahaan dapat menggunakan AWS untuk membuat aplikasi yang sesuai. HIPAA
- [Sumber daya kepatuhan AWS](#) – Kumpulan buku kerja dan panduan ini mungkin berlaku untuk industri dan lokasi Anda.
- [AWS Config](#) AWS Layanan ini menilai seberapa baik konfigurasi sumber daya Anda mematuhi praktik internal, pedoman industri, dan peraturan.
- [AWS Security Hub](#)— AWS Layanan ini memberikan pandangan komprehensif tentang keadaan keamanan Anda di dalamnya AWS yang membantu Anda memeriksa kepatuhan Anda terhadap standar industri keamanan dan praktik terbaik.

Ketahanan di AWS Transfer Family

Infrastruktur AWS global dibangun di sekitar AWS Wilayah dan Zona Ketersediaan. AWS Wilayah menyediakan beberapa Availability Zone yang terpisah secara fisik dan terisolasi, yang terhubung dengan latensi rendah, throughput tinggi, dan jaringan yang sangat redundan. Dengan Zona Ketersediaan, Anda dapat merancang serta mengoperasikan aplikasi dan basis data yang secara otomatis melakukan failover di antara Zona Ketersediaan tanpa gangguan. Zona Ketersediaan memiliki ketersediaan dan toleransi kesalahan yang lebih baik, dan dapat diskalakan dibandingkan infrastruktur biasa yang terdiri dari satu atau beberapa pusat data.

AWS Transfer Family mendukung hingga 3 Availability Zones dan didukung oleh auto scaling, armada redundan untuk koneksi Anda dan permintaan transfer.

Perhatikan hal berikut:

- Untuk titik akhir publik:
 - Ketersediaan Redundansi tingkat zona dibangun ke dalam layanan
 - Ada armada redundan untuk setiap AZ.
 - Redundansi ini disediakan secara otomatis
- Untuk titik akhir di Virtual Private Cloud (VPC), lihat [Buat server di cloud pribadi virtual](#).

Lihat juga

- Untuk informasi selengkapnya tentang Wilayah AWS dan Availability Zone, lihat [infrastruktur AWS global](#).
- [Untuk contoh tentang cara membangun redundansi yang lebih tinggi dan meminimalkan latensi jaringan dengan menggunakan perutean berbasis Latensi, lihat posting blog Minimalkan latensi jaringan dengan server Anda. AWS Transfer Family](#)

Keamanan infrastruktur di AWS Transfer Family

Sebagai layanan terkelola, AWS Transfer Family dilindungi oleh keamanan jaringan AWS global. Untuk informasi tentang layanan AWS keamanan dan cara AWS melindungi infrastruktur, lihat [Keamanan AWS Cloud](#). Untuk mendesain AWS lingkungan Anda menggunakan praktik terbaik untuk keamanan infrastruktur, lihat [Perlindungan Infrastruktur dalam Kerangka Kerja](#) yang AWS Diarsiteksikan dengan Baik Pilar Keamanan.

Anda menggunakan API panggilan yang AWS dipublikasikan untuk mengakses AWS Transfer Family melalui jaringan. Klien harus mendukung hal-hal berikut:

- Keamanan Lapisan Transportasi (TLS). Kami membutuhkan TLS 1.2 dan merekomendasikan TLS 1.3.
- Suite cipher dengan kerahasiaan maju yang sempurna (PFS) seperti (Ephemeral Diffie-Hellman) atau DHE (Elliptic Curve Ephemeral Diffie-Hellman). ECDHE Sebagian besar sistem modern seperti Java 7 dan versi lebih baru mendukung mode-mode ini.

Selain itu, permintaan harus ditandatangani dengan menggunakan ID kunci akses dan kunci akses rahasia yang terkait dengan IAM prinsipal. Atau Anda dapat menggunakan [AWS Security Token Service](#) (AWS STS) untuk menghasilkan kredensial keamanan sementara untuk menandatangani permintaan.

Tambahkan firewall aplikasi web

AWS WAF adalah firewall aplikasi web yang membantu melindungi aplikasi web dan APIs dari serangan. Anda dapat menggunakannya untuk mengonfigurasi seperangkat aturan yang dikenal sebagai daftar kontrol akses web (webACL) yang mengizinkan, memblokir, atau menghitung permintaan web berdasarkan aturan dan kondisi keamanan web yang dapat disesuaikan yang Anda tentukan. Untuk informasi selengkapnya, lihat [Menggunakan AWS WAF untuk melindungi Anda APIs](#).

Untuk menambahkan AWS WAF

1. Buka konsol API Gateway di <https://console.aws.amazon.com/apigateway/>.
2. Di panel APIsnavigasi, lalu pilih templat penyedia identitas kustom Anda.
3. Memilih Tahapan.
4. Di panel Tahapan, pilih nama panggung.
5. Di panel Editor Panggung, pilih Pengaturan tab.
6. Lakukan salah satu hal berikut ini:
 - Di bawah Web Application Firewall (WAF)ACL, untuk Web, pilih web ACL yang ingin Anda kaitkan dengan tahap ini.
 - Jika web yang ACL Anda butuhkan tidak ada, Anda harus membuatnya dengan melakukan hal berikut:
 1. Pilih Buat Web ACL.

2. Di beranda AWS WAF layanan, pilih Buat web ACL.
 3. Dalam ACLdetail Web, untuk Nama, ketikkan nama webACL.
 4. Di Aturan, pilih Tambahkan aturan, lalu pilih Tambahkan aturan dan grup aturan saya sendiri.
 5. Untuk jenis Aturan, pilih IP set untuk mengidentifikasi daftar alamat IP tertentu.
 6. Untuk Aturan, masukkan nama aturan.
 7. Untuk set IP, pilih set IP yang ada. Untuk membuat kumpulan IP, lihat [Membuat set IP](#).
 8. Untuk alamat IP yang akan digunakan sebagai alamat asal, pilih alamat IP di header.
 9. Untuk nama bidang Header, masukkanSourceIP.
 - 10.Untuk Posisi di dalam header, pilih Alamat IP Pertama.
 - 11.Untuk Fallback untuk alamat IP yang hilang, pilih Cocokkan atau Tidak Cocokkan tergantung pada bagaimana Anda ingin menangani alamat IP yang tidak valid (atau hilang) di header.
 - 12.Untuk Tindakan, pilih tindakan set IP.
 - 13.Untuk ACLtindakan web default untuk permintaan yang tidak cocok dengan aturan apa pun, pilih Izinkan atau Blokir, lalu klik Berikutnya.
 - 14.Untuk langkah 4 dan 5, pilih Berikutnya.
 - 15Di Tinjau dan buat, tinjau pilihan Anda, lalu pilih Buat web ACL.
7. Pilih Simpan Perubahan.
 8. Pilih Sumber daya.
 9. Untuk Tindakan, pilih Terapkan API.

Untuk informasi tentang seberapa aman AWS Transfer Family dengan firewall aplikasi AWS web, lihat [Mengamankan AWS Transfer Family dengan firewall AWS aplikasi dan Amazon API Gateway](#) di blog AWS penyimpanan.

Pencegahan confused deputy lintas layanan

Masalah deputy yang bingung adalah masalah keamanan di mana entitas yang tidak memiliki izin untuk melakukan tindakan dapat memaksa entitas yang lebih istimewa untuk melakukan tindakan. Pada tahun AWS, peniruan lintas layanan dapat mengakibatkan masalah wakil yang membingungkan. Peniruan identitas lintas layanan dapat terjadi ketika satu layanan (layanan yang dipanggil) memanggil layanan lain (layanan yang dipanggil). Layanan panggilan dapat dimanipulasi

untuk menggunakan izinnya untuk bertindak atas sumber daya pelanggan lain dengan cara yang seharusnya tidak memiliki izin untuk mengakses. Untuk mencegah hal ini, AWS menyediakan alat yang membantu Anda melindungi data untuk semua layanan dengan pengguna utama layanan yang telah diberi akses ke sumber daya di akun Anda. Untuk penjelasan rinci tentang masalah ini, lihat [masalah wakil yang membingungkan](#) di Panduan IAM Pengguna.

Sebaiknya gunakan kunci konteks kondisi [aws:SourceAccount](#) global [aws:SourceArn](#) dan global dalam kebijakan sumber daya untuk membatasi izin yang dimiliki AWS Transfer Family untuk sumber daya. Jika Anda menggunakan kedua kunci konteks kondisi global, `aws:SourceAccount` nilai dan akun dalam `aws:SourceArn` nilai harus menggunakan ID akun yang sama saat digunakan dalam pernyataan kebijakan yang sama.

Cara paling efektif untuk melindungi dari masalah wakil yang membingungkan adalah dengan menggunakan Amazon Resource Name (ARN) yang tepat dari sumber daya yang ingin Anda izinkan. Jika Anda menentukan beberapa sumber daya, gunakan kunci konteks kondisi `aws:SourceArn` global dengan karakter wildcard (*) untuk bagian yang tidak diketahui. ARN Misalnya, `arn:aws:transfer::region::account-id:server/*`.

AWS Transfer Family menggunakan jenis peran berikut:

- Peran pengguna — Memungkinkan pengguna yang dikelola layanan mengakses sumber daya Transfer Family yang diperlukan. AWS Transfer Family mengambil peran ini dalam konteks pengguna ARN Transfer Family.
- Peran akses - Menyediakan akses hanya ke file Amazon S3 yang sedang ditransfer. Untuk AS2 transfer masuk, peran akses menggunakan Amazon Resource Name (ARN) untuk perjanjian. Untuk AS2 transfer keluar, peran akses menggunakan konektor ARN untuk.
- Peran pemanggilan — Untuk digunakan dengan Amazon API Gateway sebagai penyedia identitas kustom server. Transfer Family mengambil peran ini dalam konteks server ARN Transfer Family.
- Peran logging - Digunakan untuk log entri ke Amazon CloudWatch. Transfer Family menggunakan peran ini untuk mencatat detail keberhasilan dan kegagalan bersama dengan informasi tentang transfer file. Transfer Family mengambil peran ini dalam konteks server ARN Transfer Family. Untuk AS2 transfer keluar, peran logging menggunakan konektor ARN.
- Peran eksekusi — Memungkinkan pengguna Transfer Family memanggil dan meluncurkan alur kerja. Transfer Family mengasumsikan peran ini dalam konteks alur kerja ARN Transfer Family.

Untuk informasi selengkapnya, lihat [Kebijakan dan izin IAM di](#) Panduan IAM Pengguna.

Note

Dalam contoh berikut, ganti masing-masing *user input placeholder* dengan informasi Anda sendiri.

Note

Dalam contoh kami, kami menggunakan keduanya `ArnLike` dan `ArnEquals`. Mereka identik secara fungsional, dan oleh karena itu Anda dapat menggunakan keduanya ketika Anda membuat kebijakan Anda. Dokumentasi Transfer Family digunakan `ArnLike` ketika kondisi berisi karakter wildcard, dan `ArnEquals` untuk menunjukkan kondisi kecocokan yang tepat.

AWS Transfer Family peran pengguna lintas layanan pencegahan wakil yang membingungkan

Contoh kebijakan berikut memungkinkan setiap pengguna dari server mana pun di akun untuk mengambil peran.

```
{
  "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/*"
        }
      }
    }
  ]
}
```

```
}
```

Contoh kebijakan berikut memungkinkan setiap pengguna dari server tertentu untuk mengambil peran.

```
{
  "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/*"
        }
      }
    }
  ]
}
```

Contoh kebijakan berikut memungkinkan pengguna tertentu dari server tertentu untuk mengambil peran.

```
{
  "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 Alur kerja Transfer Family peran lintas layanan membingungkan pencegahan wakil

Contoh kebijakan berikut memungkinkan alur kerja apa pun di akun untuk mengambil peran.

```

{
  "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/*"
        }
      }
    }
  ]
}

```

Contoh kebijakan berikut memungkinkan alur kerja tertentu untuk mengambil peran.

```

{
  "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 logging dan peran doa lintas layanan membingungkan deputi pencegahan

Note

Contoh berikut dapat digunakan dalam peran logging dan pemanggilan. Dalam contoh ini, Anda dapat menghapus ARN detail alur kerja jika server Anda tidak memiliki alur kerja yang melekat padanya.

Contoh kebijakan logging/pemanggilan berikut memungkinkan server apa pun (dan alur kerja) di akun untuk mengambil peran.

```

{
  "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/*"
        ]
    }
}
]
}

```

Contoh kebijakan logging/pemanggilan berikut memungkinkan server tertentu (dan alur kerja) untuk mengambil peran.

```

{
  "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 kebijakan terkelola untuk AWS Transfer Family

Untuk menambahkan izin ke pengguna, grup, dan peran, lebih mudah menggunakan kebijakan AWS terkelola daripada menulis kebijakan sendiri. Dibutuhkan waktu dan keahlian untuk [membuat AWS Identity and Access Management \(IAM\) kebijakan terkelola pelanggan](#) yang hanya memberi tim Anda izin yang mereka butuhkan. Untuk memulai dengan cepat, Anda dapat menggunakan kebijakan AWS terkelola kami. Kebijakan ini mencakup kasus penggunaan umum dan tersedia di Akun AWS Anda. Untuk informasi selengkapnya tentang kebijakan AWS [AWS terkelola](#), lihat [kebijakan terkelola](#) di Panduan IAM Pengguna. Untuk daftar detail semua kebijakan AWS terkelola, lihat [panduan referensi kebijakan AWS terkelola](#).

AWS layanan memelihara dan memperbarui kebijakan AWS terkelola. Anda tidak dapat mengubah izin dalam kebijakan AWS terkelola. Layanan terkadang menambahkan izin tambahan ke kebijakan yang dikelola AWS untuk mendukung fitur-fitur baru. Jenis pembaruan ini akan memengaruhi semua identitas (pengguna, grup, dan peran) di mana kebijakan tersebut dilampirkan. Layanan kemungkinan besar akan memperbarui kebijakan yang dikelola AWS saat ada fitur baru yang diluncurkan atau saat ada operasi baru yang tersedia. Layanan tidak menghapus izin dari kebijakan AWS terkelola, sehingga pembaruan kebijakan tidak akan merusak izin yang ada.

Selain itu, AWS mendukung kebijakan terkelola untuk fungsi pekerjaan yang mencakup beberapa layanan. Misalnya, kebijakan `ReadOnlyAccess` AWS terkelola menyediakan akses hanya-baca ke semua AWS layanan dan sumber daya. Saat layanan meluncurkan fitur baru, AWS menambahkan izin hanya-baca untuk operasi dan sumber daya baru. Untuk daftar dan deskripsi kebijakan fungsi pekerjaan, lihat [kebijakan AWS terkelola untuk fungsi pekerjaan](#) di Panduan IAM Pengguna.

AWS kebijakan terkelola: `AWSTransferConsoleFullAccess`

`AWSTransferConsoleFullAccess` kebijakan ini menyediakan akses penuh ke Transfer Family melalui Konsol AWS Manajemen.

Detail izin

Kebijakan ini mencakup izin berikut.

- `acm:ListCertificates`— Memberikan izin untuk mengambil daftar sertifikat Amazon Resource Names (ARNs) dan nama domain untuk masing-masing ARN
- `ec2:DescribeAddresses`— Memberikan izin untuk menjelaskan satu atau lebih alamat IP Elastis.

- `ec2:DescribeAvailabilityZones`— Memberikan izin untuk menjelaskan satu atau beberapa Availability Zone yang tersedia untuk Anda.
- `ec2:DescribeNetworkInterfaces` Memberikan izin untuk mendeskripsikan satu atau lebih antarmuka jaringan elastis.
- `ec2:DescribeSecurityGroups` Memberikan izin untuk mendeskripsikan satu atau lebih kelompok keamanan.
- `ec2:DescribeSubnets` Memberikan izin untuk mendeskripsikan satu atau lebih subnet.
- `ec2:DescribeVpcs`— Memberikan izin untuk mendeskripsikan satu atau lebih virtual private cloud (VPCs).
- `ec2:DescribeVpcEndpoints` Memberikan izin untuk mendeskripsikan satu atau lebih titik VPC akhir.
- `health:DescribeEventAggregates`— Mengembalikan jumlah acara dari setiap jenis acara (masalah, perubahan terjadwal, dan pemberitahuan akun).
- `iam:GetPolicyVersion` Memberikan izin untuk mengambil informasi tentang versi kebijakan dikelola yang ditentukan, termasuk dokumen kebijakan.
- `iam:ListPolicies`— Memberikan izin untuk membuat daftar semua kebijakan dikelola.
- `iam:ListRoles`— Memberikan izin untuk membuat daftar IAM peran yang memiliki awalan jalur yang ditentukan.
- `iam:PassRole`— Memberikan izin untuk meloloskan IAM peran ke Transfer Family. Untuk detail selengkapnya, lihat [Memberikan izin pengguna untuk meneruskan peran ke peran](#). Layanan AWS
- `route53:ListHostedZones`— Memberikan izin untuk mendapatkan daftar zona host publik dan pribadi yang terkait dengan saat ini Akun AWS.
- `s3:ListAllMyBuckets`— Memberikan izin untuk membuat daftar semua ember yang dimiliki oleh pengirim permintaan yang diautentikasi.
- `transfer:*`— Memberikan akses ke sumber daya Transfer Family. Tanda bintang (*) memberikan akses ke semua sumber daya.

```
{
  "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 kebijakan terkelola: AWSTransferFullAccess

AWSTransferFullAccessKebijakan ini menyediakan akses penuh ke layanan Transfer Family.

Detail izin

Kebijakan ini mencakup izin berikut.

- `transfer:*`— Memberikan izin untuk mengakses sumber daya Transfer Family. Tanda bintang (*) memberikan akses ke semua sumber daya.
- `iam:PassRole`— Memberikan izin untuk meloloskan IAM peran ke Transfer Family. Untuk detail selengkapnya, lihat [Memberikan izin pengguna untuk meneruskan peran ke peran](#). Layanan AWS

- `ec2:DescribeAddresses`— Memberikan izin untuk menjelaskan satu atau lebih alamat IP Elastis.
- `ec2:DescribeNetworkInterfaces` Memberikan izin untuk mendeskripsikan satu atau lebih antarmuka jaringan.
- `ec2:DescribeVpcEndpoints` Memberikan izin untuk mendeskripsikan satu atau lebih titik VPC akhir.

```
{
  "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 kebijakan terkelola: AWSTransferLoggingAccess

AWSTransferLoggingAccessKebijakan ini memberikan akses penuh kepada AWS Transfer Family untuk membuat aliran log dan grup serta memasukkan peristiwa log ke akun Anda.

Detail izin

Kebijakan ini mencakup izin berikut untuk Amazon CloudWatch Logs.

- **CreateLogStream**— Memberikan izin bagi kepala sekolah untuk membuat aliran log.
- **DescribeLogStreams**— Memberikan izin bagi kepala sekolah untuk membuat daftar aliran log untuk grup log.
- **CreateLogGroup**— Memberikan izin bagi kepala sekolah untuk membuat grup log.
- **PutLogEvents**— Memberikan izin bagi kepala sekolah untuk mengunggah sekumpulan peristiwa log ke aliran log.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "logs:CreateLogStream",
        "logs:DescribeLogStreams",
        "logs:CreateLogGroup",
        "logs:PutLogEvents"
      ],
      "Resource": "*"
    }
  ]
}
```

AWS kebijakan terkelola: AWSTransferReadOnlyAccess

AWSTransferReadOnlyAccessKebijakan ini menyediakan akses hanya-baca ke layanan Transfer Family.

Detail izin

Kebijakan ini mencakup izin berikut untuk Transfer Family.

- **DescribeUser**— Memberikan izin bagi kepala sekolah untuk melihat deskripsi bagi pengguna.
- **DescribeServer**— Memberikan izin bagi kepala sekolah untuk melihat deskripsi untuk server.
- **ListUsers**— Memberikan izin bagi kepala sekolah untuk mencantumkan pengguna untuk server.
- **ListServers**— Memberikan izin bagi kepala sekolah untuk membuat daftar server untuk akun tersebut.
- **TestIdentityProvider**— Memberikan izin kepada kepala sekolah untuk menguji apakah penyedia identitas yang dikonfigurasi sudah diatur dengan benar.
- **ListTagsForResource**— Memberikan izin bagi kepala sekolah untuk membuat daftar tag untuk sumber daya.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": [
        "transfer:DescribeUser",
        "transfer:DescribeServer",
        "transfer:ListUsers",
        "transfer:ListServers",
        "transfer:TestIdentityProvider",
        "transfer:ListTagsForResource"
      ],
      "Resource": "*"
    }
  ]
}
```

AWS Transfer Family memperbarui kebijakan AWS terkelola

Lihat detail tentang pembaruan kebijakan AWS terkelola untuk AWS Transfer Family sejak layanan ini mulai melacak perubahan ini. Untuk peringatan otomatis tentang perubahan pada halaman ini, berlangganan RSS umpan di [Riwayat dokumen untuk AWS Transfer Family](#) halaman.

Perubahan	Deskripsi	Tanggal
Pembaruan dokumentasi	Menambahkan bagian untuk setiap kebijakan yang dikelola Transfer Family.	27 Januari 2022
<u>AWSTransferReadOnlyAccess</u> – Pembaruan ke kebijakan yang ada	AWS Transfer Family menambahkan izin baru agar kebijakan dapat dibaca AWS Managed Microsoft AD.	30 September 2021
AWS Transfer Family mulai melacak perubahan	AWS Transfer Family mulai melacak perubahan kebijakan yang AWS dikelola.	15 Juni 2021

Pemecahan masalah AWS Transfer Family

Gunakan informasi berikut untuk membantu Anda mendiagnosis dan memperbaiki masalah umum yang mungkin Anda temui saat bekerja dengannya AWS Transfer Family.

Untuk masalah IAM di Transfer Family, lihat [Memecahkan masalah AWS Transfer Family identitas dan akses](#).

Topik

- [Memecahkan masalah pengguna yang dikelola layanan](#)
- [Memecahkan masalah Amazon API Gateway](#)
- [Memecahkan masalah kebijakan untuk bucket Amazon S3 terenkripsi](#)
- [Memecahkan masalah otentikasi](#)
- [Memecahkan masalah alur kerja terkelola](#)
- [Memecahkan masalah dekripsi alur kerja](#)
- [Memecahkan masalah Amazon EFS](#)
- [Memecahkan masalah pengujian penyedia identitas Anda](#)
- [Memecahkan masalah menambahkan kunci host tepercaya untuk konektor Anda SFTP](#)
- [Memecahkan masalah pengunggahan file](#)
- [Memecahkan masalah pengecualian ResourceNotFound](#)
- [Memecahkan masalah SFTP konektor](#)
- [Memecahkan masalah AS2](#)

Memecahkan masalah pengguna yang dikelola layanan

Bagian ini menjelaskan kemungkinan solusi untuk masalah berikut.

Topik

- [Memecahkan masalah pengguna yang dikelola layanan Amazon EFS](#)
- [Memecahkan masalah badan kunci publik terlalu lama](#)
- [Pemecahan masalah gagal menambahkan kunci publik SSH](#)

Memecahkan masalah pengguna yang dikelola layanan Amazon EFS

Deskripsi

Anda menjalankan `sftp` perintah dan prompt tidak muncul, dan sebagai gantinya Anda melihat pesan berikut:

```
Couldn't canonicalize: Permission denied
Need cwd
```

Menyebabkan

Peran pengguna AWS Identity and Access Management (IAM) Anda tidak memiliki izin untuk mengakses Amazon Elastic File System (AmazonEFS).

Solusi

Tingkatkan izin kebijakan untuk peran pengguna Anda. Anda dapat menambahkan kebijakan AWS terkelola, seperti `AmazonElasticFileSystemClientFullAccess`.

Memecahkan masalah badan kunci publik terlalu lama

Deskripsi

Saat Anda mencoba membuat pengguna yang dikelola layanan, Anda menerima kesalahan berikut:

```
Failed to create user (1 validation error detected:
'sshPublicKeyBody' failed to satisfy constraint: Member must have length less than or
equal to 2048)
```

Menyebabkan

Anda mungkin memasukkan PGP kunci untuk badan kunci publik, dan AWS Transfer Family tidak mendukung PGP kunci untuk pengguna yang dikelola layanan.

Solusi

Jika PGP kuncinya RSA berbasis, Anda dapat mengonversinya ke PEM format. Misalnya, Ubuntu menyediakan alat konversi di sini: <https://manpages.ubuntu.com/manpages/xenial/man1/openpgp2ssh.1.html>

Pemecahan masalah gagal menambahkan kunci publik SSH

Deskripsi

Saat Anda mencoba menambahkan kunci publik untuk pengguna yang dikelola layanan, Anda menerima kesalahan berikut:

```
Failed to add SSH public key (Unsupported or invalid SSH public key format)
```

Menyebabkan

Anda mungkin mencoba mengimpor kunci publik yang SSH2 diformat -, dan AWS Transfer Family tidak mendukung kunci publik SSH2 berformat -untuk pengguna yang dikelola layanan.

Solusi

Anda perlu mengubah kunci menjadi SSH format Terbuka. Proses ini dijelaskan di [Mengkonversi kunci SSH2 publik ke PEM format](#).

Memecahkan masalah Amazon API Gateway

Bagian ini menjelaskan kemungkinan solusi untuk masalah API Gateway berikut.

Topik

- [Terlalu banyak kegagalan otentikasi](#)
- [Koneksi ditutup](#)

Terlalu banyak kegagalan otentikasi

Deskripsi

Ketika Anda mencoba untuk terhubung ke server Anda menggunakan Secure Shell (SSH) File Transfer Protocol (SFTP), Anda mendapatkan kesalahan berikut:

```
Received disconnect from 3.15.127.197 port 22:2: Too many authentication failures  
Authentication failed.  
Couldn't read packet: Connection reset by peer
```

Menyebabkan

Anda mungkin telah memasukkan kata sandi yang salah untuk pengguna Anda. Coba lagi untuk memasukkan kata sandi yang benar.

Jika kata sandi sudah benar, masalah mungkin disebabkan oleh peran Amazon Resource Name (ARN) yang tidak valid. Untuk mengonfirmasi bahwa ini masalahnya, uji penyedia identitas untuk server Anda. Jika Anda melihat respons yang mirip dengan berikut ini, peran hanya ARN merupakan placeholder, seperti yang ditunjukkan oleh nilai ID peran dari semua nol:

```
{
  "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\"
}
```

Solusi

Ganti peran placeholder ARN dengan peran aktual yang memiliki izin untuk mengakses server.

Untuk memperbarui peran

1. Buka AWS CloudFormation konsol di <https://console.aws.amazon.com/cloudformation>.
2. Di panel navigasi sebelah kiri, pilih Tumpukan.
3. Dalam daftar Stacks, pilih tumpukan Anda, lalu pilih tab Parameter.
4. Pilih Perbarui. Pada halaman Update stack, pilih Use current template, lalu pilih Next.
5. Ganti UserRoleArn dengan peran ARN yang memiliki izin yang cukup untuk mengakses server Transfer Family Anda.

Note

Untuk memberikan izin yang diperlukan, Anda dapat menambahkan AmazonAPIGatewayAdministrator dan kebijakan AmazonS3FullAccess terkelola ke peran Anda.

6. Pilih Berikutnya, lalu pilih Berikutnya lagi. Pada Ulasan **stack** halaman, pilih Saya mengakui yang AWS CloudFormation mungkin membuat IAM sumber daya, dan kemudian pilih Perbarui tumpukan.

Koneksi ditutup

Deskripsi

Ketika Anda mencoba untuk terhubung ke server Anda menggunakan Secure Shell (SSH) File Transfer Protocol (SFTP), Anda mendapatkan kesalahan berikut:

```
Connection closed
```

Menyebabkan

Salah satu kemungkinan penyebab masalah ini adalah bahwa peran CloudWatch pencatatan Amazon Anda tidak memiliki hubungan kepercayaan dengan Transfer Family.

Solusi

Pastikan bahwa peran logging untuk server memiliki hubungan kepercayaan dengan Transfer Family. Untuk informasi selengkapnya, lihat [Untuk membangun hubungan kepercayaan](#).

Memecahkan masalah kebijakan untuk bucket Amazon S3 terenkripsi

Deskripsi

Anda memiliki bucket Amazon S3 terenkripsi yang Anda gunakan sebagai penyimpanan untuk server Transfer Family Anda. Jika Anda mencoba mengunggah file ke server, Anda menerima kesalahan `Couldn't close file: Permission denied`.

Dan jika Anda melihat log server, Anda melihat kesalahan berikut:

```
ERROR Message="Access denied" Operation=CLOSE Path=/bucket/user/test.txt BytesIn=13  
ERROR Message="Access denied"
```

Menyebabkan

Kebijakan untuk IAM pengguna Anda tidak memiliki izin untuk mengakses bucket terenkripsi.

Solusi

Anda harus menentukan izin tambahan dalam kebijakan Anda untuk memberikan izin required AWS Key Management Service (AWS KMS). Untuk detailnya, lihat [Enkripsi data di Amazon S3](#).

Memecahkan masalah otentikasi

Bagian ini menjelaskan kemungkinan solusi untuk masalah otentikasi berikut.

Topik

- [Kegagalan otentikasi—/SSHSFTP](#)
- [Masalah alam tidak cocok AD terkelola](#)
- [Masalah otentikasi lain-lain](#)

Kegagalan otentikasi—/SSHSFTP

Deskripsi

Ketika Anda mencoba untuk terhubung ke server Anda menggunakan Secure Shell (SSH) File Transfer Protocol (SFTP), Anda menerima pesan yang mirip dengan berikut ini:

```
Received disconnect from 3.130.115.105 port 22:2: Too many authentication failures
Authentication failed.
```

Note

Jika Anda menggunakan API Gateway dan menerima kesalahan ini, lihat [Terlalu banyak kegagalan otentikasi](#).

Menyebabkan

Anda belum menambahkan RSA key pair untuk pengguna Anda, jadi Anda harus mengautentikasi menggunakan kata sandi sebagai gantinya.

Solusi

Saat Anda menjalankan `sftp` perintah, tentukan `-o PubkeyAuthentication=no` opsi. Opsi ini memaksa sistem untuk meminta kata sandi Anda. Sebagai contoh:

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

Masalah alam tidak cocok AD terkelola

Deskripsi

Ranah pengguna dan ranah grup mereka harus cocok. Mereka berdua harus berada di ranah default, atau keduanya harus berada di ranah tepercaya.

Menyebabkan

Jika pengguna dan grup mereka tidak cocok, pengguna tidak dapat diautentikasi oleh Transfer Family. Jika Anda menguji penyedia identitas untuk pengguna, Anda menerima kesalahan Tidak ada akses terkait yang ditemukan untuk grup pengguna.

Solusi

Referensi grup di ranah pengguna yang cocok dengan ranah grup (baik default atau tepercaya).

Masalah otentikasi lain-lain

Deskripsi

Anda menerima kesalahan otentikasi dan tidak ada pemecahan masalah lainnya yang berfungsi

Menyebabkan

Anda mungkin telah menentukan target untuk direktori logis yang berisi garis miring (/).

Solusi

Perbarui target direktori logis Anda, untuk memastikannya dimulai dengan garis miring, dan tidak berisi garis miring. Misalnya, dapat `/DOC-EXAMPLE-BUCKET/images` diterima, tetapi `DOC-EXAMPLE-BUCKET/images` dan `/DOC-EXAMPLE-BUCKET/images/` tidak.

Memecahkan masalah alur kerja terkelola

Bagian ini menjelaskan kemungkinan solusi untuk masalah alur kerja berikut.

Topik

- [Memecahkan masalah kesalahan terkait alur kerja menggunakan Amazon CloudWatch](#)
- [Memecahkan masalah kesalahan penyalinan alur kerja](#)

Memecahkan masalah kesalahan terkait alur kerja menggunakan Amazon CloudWatch

Deskripsi

Jika Anda mengalami masalah dengan alur kerja Anda, Anda dapat menggunakan Amazon CloudWatch untuk menyelidiki penyebabnya.

Menyebabkan

Mungkin ada beberapa penyebab. Gunakan Amazon CloudWatch Logs untuk menyelidiki.

Solusi

Transfer Family memancarkan status eksekusi alur kerja ke dalam CloudWatch Log. Jenis kesalahan alur kerja berikut dapat muncul di CloudWatch Log:

- "type": "StepErrored"
- "type": "ExecutionErrored"
- "type": "ExecutionThrottled"
- "Service failure on starting workflow"

Anda dapat memfilter log eksekusi alur kerja Anda menggunakan sintaks filter dan pola yang berbeda. Misalnya, Anda dapat membuat filter log di CloudWatch log Anda untuk menangkap log eksekusi alur kerja yang berisi ExecutionErrored pesan. Untuk detailnya, lihat [Pemrosesan data log secara real-time dengan langganan](#) serta [Filter dan sintaks pola](#) di Panduan Pengguna Amazon CloudWatch Logs.

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"}
```

Di sini, `StepErrored` menunjukkan bahwa langkah dalam alur kerja telah menghasilkan kesalahan. Dalam satu alur kerja, Anda dapat memiliki beberapa langkah yang dikonfigurasi. Kesalahan ini memberi tahu Anda di langkah mana kesalahan terjadi dan memberikan pesan kesalahan. Dalam contoh khusus ini, langkah dikonfigurasi untuk menandai file; Namun, menandai file dalam sistem EFS file Amazon tidak didukung, sehingga langkah tersebut menghasilkan kesalahan.

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"}}
```

Ketika alur kerja tidak dapat mengeksekusi langkah apa pun, itu menghasilkan `ExecutionErrored` pesan. Misalnya, jika Anda telah mengonfigurasi satu langkah dalam alur kerja tertentu, dan jika langkah tersebut tidak dapat dijalankan, alur kerja keseluruhan gagal.

EksekusiTerbatas

Eksekusi dibatasi jika alur kerja dipicu pada tingkat yang lebih cepat daripada yang dapat didukung sistem. Pesan log ini menunjukkan bahwa Anda harus memperlambat laju eksekusi untuk alur kerja.

[Jika Anda tidak dapat menurunkan tingkat eksekusi alur kerja Anda, hubungi AWS Dukungan di Kontak. AWS](#)

Kegagalan layanan saat memulai alur kerja

Setiap kali Anda menghapus alur kerja dari server dan menggantinya dengan yang baru, atau memperbarui konfigurasi server (yang memengaruhi peran eksekusi alur kerja), Anda harus menunggu sekitar 10 menit sebelum menjalankan alur kerja baru. Server Transfer Family menyimpan cache detail alur kerja, dan dibutuhkan waktu 10 menit bagi server untuk menyegarkan cache-nya.

Selain itu, Anda harus keluar dari SFTP sesi aktif apa pun, dan kemudian masuk kembali setelah masa tunggu 10 menit untuk melihat perubahannya.

Memecahkan masalah kesalahan penyalinan alur kerja

Deskripsi

Jika Anda menjalankan alur kerja yang berisi langkah untuk menyalin file yang diunggah, Anda dapat mengalami kesalahan berikut:

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

Menyebabkan

File sumber ada di bucket Amazon S3 yang berbeda Wilayah AWS dari bucket tujuan.

Solusi

Jika Anda menjalankan alur kerja yang menyertakan langkah penyalinan, pastikan bucket sumber dan tujuan berada di tempat yang sama. Wilayah AWS

Memecahkan masalah dekripsi alur kerja

Deskripsi

Alur kerja dekripsi Anda gagal, dan pesan log menyerupai berikut ini:

```
{
  "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"
  }
}

```

Menyebabkan

Server Transfer Family Anda mengaktifkan FIPS mode dan langkah Dekripsi alur kerja terkait. Saat mengenkripsi file sebelum mengunggah ke server Transfer Family Anda, klien enkripsi mungkin menghasilkan file terenkripsi yang menggunakan algoritme enkripsi simetris yang tidak disetujui. FIPS Dalam skenario seperti itu, alur kerja tidak dapat mendekripsi file. Dalam contoh berikut, GnuPG versi 2.4.0 OCB menggunakan (mode sandi FIPS non-blok) untuk mengenkripsi file: ini menyebabkan alur kerja gagal.

Solusi

Anda harus mengedit GPG kunci yang Anda gunakan untuk mengenkripsi file Anda, dan kemudian mengenkripsi ulang mereka. Prosedur berikut menjelaskan langkah-langkah yang harus Anda ambil.

Untuk mengedit PGP kunci Anda

1. Identifikasi kunci yang harus Anda edit dengan menjalankan `gpg --list-keys`

Ini mengembalikan daftar kunci. Setiap kunci memiliki detail yang mirip dengan yang berikut:

```

pub   ed25519 2022-07-07 [SC]
      wJa1rXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
uid           [ultimate] Mary Major <marymajor@example.com>
sub   cv25519 2022-07-07 [E]

```

2. Identifikasi kunci yang ingin Anda edit. Pada contoh yang ditunjukkan pada langkah sebelumnya, ID adalah `wJa1rXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY`.
3. Jalankan `gpg --edit-key wJa1rXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY`.

Sistem merespons dengan detail tentang program GnuPG dan kunci yang ditentukan.

4. Pada `gpg>` prompt, masukkan `showpref`. Rincian berikut dikembalikan:

```
[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
```

Perhatikan bahwa algoritma pilihan yang disimpan pada kunci terdaftar.

5. Kami ingin mengedit kunci untuk mempertahankan semua algoritma kecuali untuk OCB. Jalankan `setpref` perintah, tentukan semua algoritma untuk mempertahankan:

```
gpg> setpref AES256, AES192, AES, 3DES, SHA512, SHA384, SHA256, SHA224, SHA1, ZLIB,
  BZIP2, ZIP, Uncompressed
```

Ini mengembalikan rincian berikut:

```
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. Masukkan `y` untuk memperbarui, lalu masukkan kata sandi Anda saat diminta untuk mengonfirmasi perubahan.
7. Simpan perubahan.

```
gpg> save
```

Sebelum menjalankan kembali alur kerja dekripsi Anda, Anda harus mengenkripsi ulang file Anda, menggunakan kunci yang diedit.

Memecahkan masalah Amazon EFS

Bagian ini menjelaskan kemungkinan solusi untuk EFS masalah Amazon berikut.

Topik

- [Memecahkan masalah profil yang hilang POSIX](#)
- [Memecahkan masalah direktori logis dengan Amazon EFS](#)

Memecahkan masalah profil yang hilang POSIX

Deskripsi

Jika Anda menggunakan EFS penyimpanan Amazon untuk server Anda dan Anda menggunakan penyedia identitas khusus, Anda harus menyediakan AWS Lambda fungsi Anda dengan POSIX profil.

Menyebabkan

Salah satu kemungkinan penyebabnya adalah template yang kami sediakan untuk membuat metode Amazon API Gateway yang AWS Lambda didukung saat ini tidak berisi POSIX informasi.

Jika Anda memberikan POSIX informasi, format yang Anda gunakan untuk memberikan POSIX informasi mungkin tidak diuraikan dengan benar oleh Transfer Family.

Solusi

Pastikan bahwa Anda menyediakan JSON elemen untuk Transfer Family untuk PosixProfile parameter.

Misalnya, jika Anda menggunakan Python, Anda dapat menambahkan baris berikut di mana Anda mengurai parameter: PosixProfile

```
if PosixProfile:
    response_data["PosixProfile"] = json.loads(PosixProfile)
```

Atau, di JavaScript, Anda bisa menambahkan baris berikut, di mana *uid-value* dan *gid-value* bilangan bulat, 0 atau lebih besar, yang mewakili User ID (UID) dan Group ID (GID) masing-masing:

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

Contoh kode ini mengirim PosixProfile parameter ke Transfer Family sebagai JSON objek, bukan sebagai string.

Juga, di dalam AWS Secrets Manager, Anda harus menyimpan PosixProfile parameter sebagai berikut. Ganti *your-uid* dan *your-gid* dengan nilai aktual Anda untuk GID dan UID.

```
{"Uid": your-uid, "Gid": your-gid, "SecondaryGids": []}
```

Memecahkan masalah direktori logis dengan Amazon EFS

Deskripsi

Jika direktori home pengguna tidak ada, dan mereka menjalankan `ls` perintah, sistem merespons sebagai berikut:

```
sftp> ls
remote readdir ("/"): No such file or directory
```

Menyebabkan

Jika server Transfer Family Anda menggunakan AmazonEFS, direktori home untuk pengguna harus dibuat dengan akses baca dan tulis sebelum pengguna dapat bekerja di direktori home logisnya. Pengguna tidak dapat membuat direktori ini sendiri, karena mereka akan kekurangan izin untuk `mkdir` direktori home logis mereka.

Solusi

Seorang pengguna dengan akses administratif ke direktori induk perlu membuat direktori home logis pengguna.

Memecahkan masalah pengujian penyedia identitas Anda

Deskripsi

Jika Anda menguji penyedia identitas menggunakan konsol atau TestIdentityProvider API panggilan, Response bidang tersebut kosong. Sebagai contoh:

```
{
  "Response": "{}",
  "StatusCode": 200,
```



```
"Message": ""  
}
```

Menyebabkan

Penyebab yang paling mungkin adalah otentikasi gagal karena nama pengguna atau kata sandi yang salah.

Solusi

Pastikan Anda menggunakan kredensi yang benar untuk pengguna Anda, dan buat pembaruan pada nama pengguna atau kata sandi, jika perlu.

Memecahkan masalah menambahkan kunci host tepercaya untuk konektor Anda SFTP

Deskripsi

Saat Anda membuat atau mengedit SFTP konektor, dan Anda menambahkan kunci host tepercaya, Anda menerima kesalahan berikut: `Failed to edit connector details (Invalid host key format.)`

Menyebabkan

Jika Anda menempelkan kunci publik yang benar, masalahnya mungkin Anda memasukkan `comment` bagian kunci tersebut. AWS Transfer Family saat ini tidak menerima bagian komentar dari kunci.

Solusi

Hapus bagian komentar dari kunci, saat Anda menempelkannya ke bidang teks. Misalnya, anggap kunci Anda terlihat mirip dengan yang berikut ini:

```
ssh-rsa AAAA...== marymajor@dev-dsk-marymajor-1d-c1234567.us-east-1.amazon.com
```

Hapus teks yang mengikuti `==` karakter dan hanya tempel di bagian tombol hingga dan termasuk `==`.

```
ssh-rsa AAAA...==
```

Memecahkan masalah pengunggahan file

Bagian ini menjelaskan kemungkinan solusi untuk masalah unggahan file berikut.

Topik

- [Memecahkan masalah kesalahan unggahan file Amazon S3](#)
- [Memecahkan masalah nama file yang tidak dapat dibaca](#)

Memecahkan masalah kesalahan unggahan file Amazon S3

Deskripsi

Saat Anda mencoba mengunggah file ke penyimpanan Amazon S3 menggunakan Transfer Family, Anda menerima pesan galat berikut AWS : Transfer tidak mendukung penulisan akses acak ke objek S3.

Menyebabkan

Saat Anda menggunakan Amazon S3 untuk penyimpanan server Anda, Transfer Family tidak mendukung beberapa koneksi untuk satu transfer.

Solusi

Jika server Transfer Family Anda menggunakan Amazon S3 untuk penyimpanannya, nonaktifkan opsi apa pun untuk perangkat lunak klien Anda yang menyebutkan menggunakan beberapa koneksi untuk satu transfer.

Memecahkan masalah nama file yang tidak dapat dibaca

Deskripsi

Anda melihat nama file rusak di beberapa file yang Anda unggah. Pengguna terkadang mengalami masalah dengan FTP dan SFTP transfer yang mengacaukan karakter tertentu dalam nama file, seperti umlauts, huruf beraksen, atau skrip tertentu, seperti bahasa Mandarin atau Arab.

Menyebabkan

Meskipun SFTP protokol FTP dan dapat memungkinkan pengkodean karakter nama file untuk dinegosiasikan oleh klien, Amazon S3 dan Amazon tidak. EFS Sebaliknya, mereka membutuhkan UTF -8 pengkodean karakter. Akibatnya, karakter tertentu tidak dirender dengan benar.

Solusi

Untuk mengatasi masalah ini, tinjau aplikasi klien Anda untuk pengkodean karakter nama file dan pastikan itu diatur ke UTF -8.

Memecahkan masalah pengecualian **ResourceNotFound**

Deskripsi

Anda menerima kesalahan di mana sumber daya tidak dapat ditemukan. Misalnya, jika Anda menjalankan `UpdateServer`, Anda mungkin mendapatkan kesalahan berikut:

```
An error occurred (ResourceNotFoundException) when calling the UpdateServer operation:  
Unknown server
```

Menyebabkan

Ada beberapa alasan untuk menerima `ResourceNotFoundException` pesan. Dalam kebanyakan kasus, sumber daya yang Anda tentukan dalam API perintah Anda tidak ada. Jika Anda memang menentukan sumber daya yang ada, maka penyebab yang paling mungkin adalah bahwa wilayah default Anda berbeda dari wilayah untuk sumber daya Anda. Misalnya, jika wilayah default Anda adalah `us-east-1`, dan server Transfer Family Anda berada di `us-east-2`, Anda akan menerima pengecualian sumber daya Tidak Dikenal.

Untuk detail tentang menyetel wilayah default, lihat [Konfigurasi cepat dengan aws configure](#).

Solusi

Tambahkan parameter wilayah ke API perintah Anda untuk secara eksplisit menentukan di mana menemukan sumber daya tertentu.

```
aws transfer -describe-server --server-id server-id --region us-east-2
```

Memecahkan masalah SFTP konektor

Bagian ini menjelaskan kemungkinan solusi untuk masalah SFTP konektor berikut.

Topik

- [Negosiasi kunci gagal](#)
- [Masalah konektor lain-lain SFTP](#)

Negosiasi kunci gagal

Deskripsi

Anda menerima kesalahan di mana negosiasi pertukaran kunci gagal. Sebagai contoh:

```
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]
```

Menyebabkan

Kesalahan ini karena tidak ada tumpang tindih antara algoritma kunci host yang didukung oleh server dan yang didukung oleh konektor.

Solusi

Pastikan server jarak jauh mendukung setidaknya satu dari algoritma kunci host Klien yang tercantum dalam pesan kesalahan. Untuk daftar algoritma yang didukung, lihat [SFTPAlgoritma konektor](#).

Masalah konektor lain-lain SFTP

Deskripsi

Anda menerima kesalahan setelah menjalankan `StartFileTransfer`, tetapi tidak tahu penyebab masalah, dan hanya ID konektor yang dikembalikan setelah API panggilan.

Menyebabkan

Kesalahan ini dapat memiliki beberapa penyebab. Untuk memecahkan masalah, kami sarankan Anda menguji konektor Anda dan mencari log Anda CloudWatch .

Solusi

- Uji konektor Anda: Lihat [Uji SFTP konektor](#). Jika tes gagal, sistem memberikan pesan kesalahan berdasarkan alasan pengujian gagal. Bagian itu menjelaskan cara menguji konektor Anda baik dari konsol atau dengan menggunakan [TestConnection](#) API perintah.

- Lihat CloudWatch log untuk konektor Anda: Lihat [Contoh entri log untuk konektor SFTP](#). Topik ini memberikan contoh entri log SFTP konektor, dan konvensi penamaan untuk membantu Anda menemukan log yang sesuai.

Memecahkan masalah AS2

Pesan kesalahan dan tips pemecahan masalah untuk server yang mendukung Pernyataan Penerapan 2 (AS2) dijelaskan di sini: [Kode kesalahan AS2](#)

API referensi

Bagian berikut mendokumentasikan panggilan AWS Transfer Family API layanan, tipe data, parameter, dan kesalahan.

Topik

- [Selamat datang di AWS Transfer Family API](#)
- [Tindakan](#)
- [Tipe Data](#)
- [Membuat permintaan API](#)
- [Parameter Umum](#)
- [Kesalahan Umum](#)

Selamat datang di AWS Transfer Family API

AWS Transfer Family adalah layanan transfer aman yang dapat Anda gunakan untuk mentransfer file masuk dan keluar dari penyimpanan Amazon Simple Storage Service (Amazon S3) melalui protokol berikut:

- Secure Shell (SSH) Protokol Transfer File (SFTP)
- Protokol Transfer File Aman (FTPS)
- Protokol Transfer File (FTP)
- Pernyataan Penerapan 2 () AS2

Protokol transfer file digunakan dalam alur kerja pertukaran data di berbagai industri seperti layanan keuangan, perawatan kesehatan, periklanan, dan ritel, antara lain. AWS Transfer Family menyederhanakan migrasi alur kerja transfer file ke AWS.

Untuk menggunakan AWS Transfer Family layanan ini, Anda membuat instance server di AWS Wilayah pilihan Anda. Anda dapat membuat server, daftar server yang tersedia, dan memperbarui dan menghapus server. Server adalah entitas yang meminta operasi file dari AWS Transfer Family. Server memiliki sejumlah properti penting. Server adalah contoh bernama seperti yang diidentifikasi oleh `ServerId` pengidentifikasi yang ditetapkan sistem. Anda dapat secara opsional

menetapkan nama host, atau bahkan nama host khusus ke server. Tagihan layanan untuk setiap server instantiated (bahkan yang OFFLINE), dan untuk jumlah data yang ditransfer.

Pengguna harus diketahui oleh server yang meminta operasi file. Seorang pengguna yang diidentifikasi oleh nama pengguna mereka ditugaskan ke server. Nama pengguna digunakan untuk mengautentikasi permintaan. Server hanya dapat memiliki satu metode otentikasi: `AWS_DIRECTORY_SERVICE`, `SERVICE_MANAGED_AWS_LAMBDA`, atau `API_GATEWAY`.

Anda dapat menggunakan salah satu jenis penyedia identitas berikut untuk mengautentikasi pengguna:

- Untuk `SERVICE_MANAGED`, kunci SSH publik disimpan dengan properti pengguna di server. Seorang pengguna dapat memiliki satu atau lebih kunci SSH publik pada file untuk metode `SERVICE_MANAGED` otentikasi. Ketika klien meminta operasi file untuk `SERVICE_MANAGED` metode, klien menyediakan nama pengguna dan kunci SSH pribadi, yang diautentikasi, dan akses disediakan.
- Anda dapat mengelola otentikasi dan akses pengguna dengan grup Microsoft Active Directory. Anda dengan memilih metode `AWS_DIRECTORY_SERVICE` otentikasi.
- Anda dapat terhubung ke penyedia identitas kustom dengan menggunakan AWS Lambda. Pilih metode `AWS_LAMBDA` otentikasi.
- Anda juga dapat mengautentikasi permintaan pengguna menggunakan metode otentikasi khusus yang menyediakan otentikasi dan akses pengguna. Metode ini bergantung pada Amazon API Gateway untuk menggunakan API panggilan Anda dari penyedia identitas Anda untuk memvalidasi permintaan pengguna. Metode ini disebut sebagai `API_GATEWAY` dalam API panggilan, dan sebagai Kustom di konsol. Anda dapat menggunakan metode kustom ini untuk mengautentikasi pengguna terhadap layanan direktori, pasangan nama/kata sandi database, atau mekanisme lainnya.

Pengguna diberi kebijakan dengan hubungan kepercayaan antara mereka dan bucket Amazon S3. Mereka mungkin dapat mengakses semua atau sebagian dari ember. Agar server bertindak atas nama pengguna, server harus mewarisi hubungan kepercayaan dari pengguna. Peran AWS Identity and Access Management (IAM) dibuat yang berisi hubungan kepercayaan, dan peran tersebut diberi `AssumeRole` tindakan. Server kemudian dapat melakukan operasi file seolah-olah itu adalah pengguna.

Pengguna yang memiliki set properti home direktori akan memiliki direktori (atau folder) yang bertindak sebagai target dan sumber operasi file. Ketika tidak ada home direktori yang disetel, root direktori bucket menjadi direktori pendaratan.

Server, pengguna, dan peran semuanya diidentifikasi oleh Amazon Resource Name (ARN). Anda dapat menetapkan tag, yang merupakan pasangan nilai kunci, ke entitas dengan. ARN Tag adalah metadata yang dapat digunakan untuk mengelompokkan atau mencari entitas ini. Salah satu contoh di mana tag berguna adalah untuk tujuan akuntansi.

Konvensi berikut diamati dalam format AWS Transfer Family ID:

- `ServerId` nilai mengambil bentuk `s-01234567890abcdef`.
- `SshPublicKeyId` nilai mengambil bentuk `key-01234567890abcdef`.

Format Amazon Resource Name (ARN) mengambil bentuk berikut:

- Untuk server, ARNs ambil formulir `arn:aws:transfer:region:account-id:server/server-id`.

Contoh server ARN adalah `arn:aws:transfer:us-east-1:123456789012:server/s-01234567890abcdef`.

- Untuk pengguna, ARNs ambil formulir `arn:aws:transfer:region:account-id:user/server-id/username`.

Contohnya adalah `arn:aws:transfer:us-east-1:123456789012:user/s-01234567890abcdef/user1`.

DNSentri (titik akhir) yang digunakan adalah sebagai berikut:

- API titik akhir mengambil formulir `transfer.region.amazonaws.com`.
- Titik akhir server mengambil formulir `server.transfer.region.amazonaws.com`.

Untuk daftar titik akhir Transfer Family menurut AWS Wilayah, lihat [AWS Transfer Family titik akhir dan kuota](#) di. Referensi Umum AWS

Referensi API antarmuka ini AWS Transfer Family berisi dokumentasi untuk antarmuka pemrograman yang dapat Anda gunakan untuk mengelola AWS Transfer Family. Struktur referensi adalah sebagai berikut:

- Untuk daftar API tindakan menurut abjad, lihat [Actions](#).
- Untuk daftar alfabet tipe data, lihat [Data Types](#).
- Untuk daftar parameter kueri umum, lihat [Parameter Umum](#).
- Untuk deskripsi kode kesalahan, lihat [Kesalahan Umum](#).

Tip

Daripada benar-benar menjalankan perintah, Anda dapat menggunakan `--generate-cli-skeleton` parameter dengan API panggilan apa pun untuk menghasilkan dan menampilkan template parameter. Anda kemudian dapat menggunakan template yang dihasilkan untuk menyesuaikan dan digunakan sebagai input pada perintah selanjutnya. Untuk detailnya, lihat [Menghasilkan dan menggunakan file kerangka parameter](#).

Tindakan

Tindakan berikut didukung:

- [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](#)
- [StartFileTransfer](#)

- [StartServer](#)
- [StopServer](#)
- [TagResource](#)
- [TestConnection](#)
- [TestIdentityProvider](#)
- [UntagResource](#)
- [UpdateAccess](#)
- [UpdateAgreement](#)
- [UpdateCertificate](#)
- [UpdateConnector](#)
- [UpdateHostKey](#)
- [UpdateProfile](#)
- [UpdateServer](#)
- [UpdateUser](#)

CreateAccess

Digunakan oleh administrator untuk memilih grup mana dalam direktori yang harus memiliki akses untuk mengunggah dan mengunduh file melalui protokol yang diaktifkan menggunakan. AWS Transfer Family Misalnya, Microsoft Active Directory mungkin berisi 50.000 pengguna, tetapi hanya sebagian kecil yang mungkin memerlukan kemampuan untuk mentransfer file ke server. Administrator dapat menggunakan CreateAccess untuk membatasi akses ke set pengguna yang benar yang membutuhkan kemampuan ini.

Sintaksis Permintaan

```
{
  "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"
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[ExternalId](#)

Pengidentifikasi unik yang diperlukan untuk mengidentifikasi grup tertentu dalam direktori Anda. Pengguna grup yang Anda asosiasikan memiliki akses ke EFS sumber daya Amazon S3 atau

Amazon Anda melalui protokol yang diaktifkan. AWS Transfer Family Jika Anda tahu nama grup, Anda dapat melihat SID nilai dengan menjalankan perintah berikut menggunakan Windows PowerShell.

```
Get-ADGroup -Filter {samAccountName -like "YourGroupName*"} -Properties  
* | Select SamAccountName, ObjectSid
```

Dalam perintah itu, ganti YourGroupName dengan nama grup Active Directory Anda.

Ekspresi reguler yang digunakan untuk memvalidasi parameter ini adalah string karakter yang terdiri dari huruf besar dan huruf kecil karakter alfanumerik tanpa spasi. Anda juga dapat menyertakan garis bawah atau salah satu karakter berikut: =, . @: /-

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 256.

Pola: S-1-[\d-]+

Wajib: Ya

[HomeDirectory](#)

Direktori arahan (folder) untuk pengguna ketika mereka masuk ke server menggunakan klien.

Contoh HomeDirectory adalah /bucket_name/home/mydirectory.

Note

Parameter HomeDirectory hanya digunakan jika HomeDirectoryType diatur ke PATH.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 1024.

Pola: (| / . *)

Wajib: Tidak

[HomeDirectoryMappings](#)

Pemetaan direktori logis yang menentukan jalur dan kunci Amazon S3 atau EFS Amazon apa yang harus terlihat oleh pengguna Anda dan bagaimana Anda ingin membuatnya terlihat. Anda

harus menentukan Entry dan Target memasangkan, di mana Entry menunjukkan bagaimana jalur dibuat terlihat dan Target merupakan jalur Amazon S3 atau Amazon yang sebenarnya. EFS Jika Anda hanya menentukan target, itu ditampilkan apa adanya. Anda juga harus memastikan bahwa peran AWS Identity and Access Management (IAM) Anda menyediakan akses ke jalur diTarget. Nilai ini dapat diatur hanya ketika HomeDirectoryType diatur ke LOGICAL.

Berikut ini adalah contoh Entry dan Target pair.

```
[ { "Entry": "/directory1", "Target": "/bucket_name/home/mydirectory" } ]
```

Dalam kebanyakan kasus, Anda dapat menggunakan nilai ini alih-alih kebijakan sesi untuk mengunci pengguna Anda ke direktori home yang ditunjuk (chroot[™]). Untuk melakukan ini, Anda dapat mengatur Entry ke / dan mengatur Target ke nilai HomeDirectory parameter.

Berikut ini adalah contoh Entry dan Target pair untukchroot.

```
[ { "Entry": "/", "Target": "/bucket_name/home/mydirectory" } ]
```

Tipe: Array objek [HomeDirectoryMapEntry](#)

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 50000 item.

Wajib: Tidak

[HomeDirectoryType](#)

Jenis direktori pendaratan (folder) yang Anda inginkan direktori home pengguna Anda ketika mereka masuk ke server. Jika Anda mengaturnyaPATH, pengguna akan melihat bucket Amazon S3 absolut atau EFS jalur Amazon seperti pada klien protokol transfer file mereka. Jika Anda menyetelnyaLOGICAL, Anda harus menyediakan pemetaan HomeDirectoryMappings untuk bagaimana Anda ingin membuat jalur Amazon S3 atau EFS Amazon terlihat oleh pengguna Anda.

Note

Jika HomeDirectoryType yaLOGICAL, Anda harus memberikan pemetaan, menggunakan parameter. HomeDirectoryMappings Jika, di sisi lain, HomeDirectoryType adalahPATH, Anda memberikan jalur absolut menggunakan HomeDirectory parameter. Anda tidak dapat memiliki keduanya HomeDirectory dan HomeDirectoryMappings di template Anda.

Tipe: String

Nilai yang Valid: PATH | LOGICAL

Wajib: Tidak

[Policy](#)

Kebijakan sesi untuk pengguna Anda sehingga Anda dapat menggunakan peran AWS Identity and Access Management (IAM) yang sama di beberapa pengguna. Kebijakan ini mencakup akses pengguna ke sebagian bucket Amazon S3 mereka. Variabel yang dapat Anda gunakan dalam kebijakan ini meliputi `${Transfer:UserName}`, `${Transfer:HomeDirectory}`, dan `${Transfer:HomeBucket}`.

Note

Kebijakan ini hanya berlaku jika domainnya `ServerId` adalah Amazon S3. Amazon EFS tidak menggunakan kebijakan sesi.

Untuk kebijakan sesi, AWS Transfer Family menyimpan kebijakan sebagai JSON gumpalan, bukan Amazon Resource Name (ARN) kebijakan. Anda menyimpan kebijakan sebagai JSON gumpalan dan meneruskannya dalam `Policy` argumen.

Untuk contoh kebijakan sesi, lihat [Contoh kebijakan sesi](#).

Untuk informasi lebih lanjut, lihat [AssumeRole](#) di AWS Security Token Service API Referensi.

Tipe: String

Batasan Panjang: Panjang minimum sebesar 0. Panjang maksimum 2048.

Wajib: Tidak

[PosixProfile](#)

POSIXIdentitas lengkap, termasuk ID pengguna (`Uid`), ID grup (`Gid`), dan grup sekunder IDs (`SecondaryGids`) apa pun, yang mengontrol akses pengguna ke sistem EFS file Amazon Anda. POSIXIzin yang ditetapkan pada file dan direktori di sistem file Anda menentukan tingkat akses yang didapat pengguna saat mentransfer file masuk dan keluar dari sistem file Amazon EFS Anda.

Tipe: Objek [PosixProfile](#)

Wajib: Tidak

Role

Nama Sumber Daya Amazon (ARN) dari peran AWS Identity and Access Management (IAM) yang mengontrol akses pengguna ke bucket Amazon S3 atau sistem file EFS Amazon. Kebijakan yang dilampirkan pada peran ini menentukan tingkat akses yang ingin Anda berikan kepada pengguna saat mentransfer file masuk dan keluar dari bucket Amazon S3 atau sistem file EFS Amazon Anda. IAM Peran tersebut juga harus berisi hubungan kepercayaan yang memungkinkan server mengakses sumber daya Anda saat melayani permintaan transfer pengguna Anda.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 2048.

Pola: `arn:.*role/\S+`

Wajib: Ya

ServerId

Pengidentifikasi unik yang ditetapkan sistem untuk instans server. Ini adalah server tertentu tempat Anda menambahkan pengguna.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: `s-([0-9a-f]{17})`

Diperlukan: Ya

Sintaksis Respons

```
{
  "ExternalId": "string",
  "ServerId": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

ExternalId

Pengenal eksternal grup yang penggunanya memiliki akses ke sumber daya Amazon S3 atau EFS Amazon Anda melalui protokol yang diaktifkan menggunakan AWS Transfer Family

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 256.

Pola: S-1-[\d-]+

ServerId

Pengidentifikasi server tempat pengguna dilampirkan.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s-([0-9a-f]{17})

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServiceError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceExistsException

Sumber daya yang diminta tidak ada, atau ada di wilayah selain yang ditentukan untuk perintah.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

CreateAgreement

Membuat perjanjian. Perjanjian adalah perjanjian mitra dagang bilateral, atau kemitraan, antara AWS Transfer Family server dan AS2 proses. Perjanjian mendefinisikan hubungan transfer file dan pesan antara server dan AS2 proses. Untuk menentukan perjanjian, Transfer Family menggabungkan server, profil lokal, profil mitra, sertifikat, dan atribut lainnya.

Mitra diidentifikasi dengan `PartnerProfileId`, dan AS2 prosesnya diidentifikasi dengan `LocalProfileId`.

Sintaksis Permintaan

```
{
  "AccessRole": "string",
  "BaseDirectory": "string",
  "Description": "string",
  "LocalProfileId": "string",
  "PartnerProfileId": "string",
  "ServerId": "string",
  "Status": "string",
  "Tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ]
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

AccessRole

Konektor digunakan untuk mengirim file menggunakan SFTP protokol AS2 atau. Untuk peran akses, berikan Amazon Resource Name (ARN) AWS Identity and Access Management peran yang akan digunakan.

Untuk AS2 konektor

Dengan AS2, Anda dapat mengirim file dengan memanggil `StartFileTransfer` dan menentukan jalur file dalam parameter permintaan, `SendFilePaths`. Kami menggunakan direktori induk file (misalnya, untuk `--send-file-paths /bucket/dir/file.txt`, direktori induk `/bucket/dir/`) untuk sementara menyimpan file AS2 pesan yang diproses, menyimpan MDN ketika kami menerimanya dari mitra, dan menulis JSON file akhir yang berisi metadata transmisi yang relevan. Jadi, `AccessRole` kebutuhan untuk menyediakan akses baca dan tulis ke direktori induk dari lokasi file yang digunakan dalam `StartFileTransfer` permintaan. Selain itu, Anda perlu menyediakan akses baca dan tulis ke direktori induk dari file yang ingin Anda `kirimStartFileTransfer`.

Jika Anda menggunakan otentikasi Dasar untuk AS2 konektor Anda, peran akses memerlukan `secretsmanager:GetSecretValue` izin untuk rahasia tersebut. Jika rahasia dienkripsi menggunakan kunci yang dikelola pelanggan alih-alih kunci terkelola di AWS Secrets Manager, maka peran tersebut juga memerlukan `kms:Decrypt` izin untuk kunci tersebut.

Untuk SFTP konektor

Pastikan bahwa peran akses menyediakan akses baca dan tulis ke direktori induk dari lokasi file yang digunakan dalam `StartFileTransfer` permintaan. Selain itu, pastikan bahwa peran tersebut memberikan `secretsmanager:GetSecretValue` izin untuk AWS Secrets Manager.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 2048.

Pola: `arn:.*role/\S+`

Wajib: Ya

BaseDirectory

Direktori pendaratan (folder) untuk file yang ditransfer dengan menggunakan AS2 protokol.

Contoh `BaseDirectory` adalah `/DOC-EXAMPLE-BUCKET/home/mydirectory`.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 1024.

Pola: `(|/.*)`

Wajib: Ya

Description

Nama atau deskripsi singkat untuk mengidentifikasi perjanjian.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 200.

Pola: $[\backslash p\{Graph\}]^+$

Wajib: Tidak

LocalProfileId

Pengenal unik untuk profil AS2 lokal.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: $p-([\backslash 0-9a-f]\{17\})$

Wajib: Ya

PartnerProfileId

Pengenal unik untuk profil mitra yang digunakan dalam perjanjian.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: $p-([\backslash 0-9a-f]\{17\})$

Wajib: Ya

ServerId

Pengidentifikasi unik yang ditetapkan sistem untuk instans server. Ini adalah server khusus yang digunakan perjanjian.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: $s-([\backslash 0-9a-f]\{17\})$

Wajib: Ya

Status

Status perjanjian. Kesepakatan itu bisa berupa ACTIVE atau INACTIVE.

Tipe: String

Nilai yang Valid: ACTIVE | INACTIVE

Wajib: Tidak

Tags

Pasangan nilai kunci yang dapat digunakan untuk mengelompokkan dan mencari perjanjian.

Tipe: Array objek [Tag](#)

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 50 item.

Diperlukan: Tidak

Sintaksis Respons

```
{  
  "AgreementId": "string"  
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

AgreementId

Pengidentifikasi unik untuk perjanjian. Gunakan ID ini untuk menghapus, atau memperbarui perjanjian, serta dalam API panggilan lain yang mengharuskan Anda menentukan ID perjanjian.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: a-([0-9a-f]{17})

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceExistsException

Sumber daya yang diminta tidak ada, atau ada di wilayah selain yang ditentukan untuk perintah.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

ThrottlingException

Permintaan ditolak karena throttling permintaan.

HTTPKode Status: 400

Contoh

Contoh

Contoh berikut membuat perjanjian, dan mengembalikan ID perjanjian.

```
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
```

Contoh Respons

API Panggilan mengembalikan ID perjanjian untuk perjanjian baru.

```
{
  "AgreementId": "a-11112222333344444"
}
```

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

CreateConnector

Membuat konektor, yang menangkap parameter untuk koneksi untuk SFTP protokol AS2 atau. Untuk AS2, konektor diperlukan untuk mengirim file ke server yang dihosting AS2 secara eksternal. Untuk SFTP, konektor diperlukan saat mengirim file ke SFTP server atau menerima file dari SFTP server. Untuk detail selengkapnya tentang konektor, lihat [Mengkonfigurasi AS2 konektor](#) dan [Membuat SFTP konektor](#).

Note

Anda harus menentukan dengan tepat satu objek konfigurasi: baik untuk AS2 (As2Config) atau SFTP (SftpConfig).

Sintaksis Permintaan

```
{
  "AccessRole": "string",
  "As2Config": {
    "BasicAuthSecretId": "string",
    "Compression": "string",
    "EncryptionAlgorithm": "string",
    "LocalProfileId": "string",
    "MdnResponse": "string",
    "MdnSigningAlgorithm": "string",
    "MessageSubject": "string",
    "PartnerProfileId": "string",
    "SigningAlgorithm": "string"
  },
  "LoggingRole": "string",
  "SftpConfig": {
    "TrustedHostKeys": [ "string" ],
    "UserSecretId": "string"
  },
  "Tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ],
  "Url": "string"
```

```
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[AccessRole](#)

Konektor digunakan untuk mengirim file menggunakan SFTP protokol AS2 atau. Untuk peran akses, berikan Amazon Resource Name (ARN) AWS Identity and Access Management peran yang akan digunakan.

Untuk AS2 konektor

Dengan AS2, Anda dapat mengirim file dengan memanggil `StartFileTransfer` dan menentukan jalur file dalam parameter permintaan, `SendFilePaths`. Kami menggunakan direktori induk file (misalnya, untuk `--send-file-paths /bucket/dir/file.txt`, direktori induk `/bucket/dir/`) untuk sementara menyimpan file AS2 pesan yang diproses, menyimpan MDN ketika kami menerimanya dari mitra, dan menulis JSON file akhir yang berisi metadata transmisi yang relevan. Jadi, `AccessRole` kebutuhan untuk menyediakan akses baca dan tulis ke direktori induk dari lokasi file yang digunakan dalam `StartFileTransfer` permintaan. Selain itu, Anda perlu menyediakan akses baca dan tulis ke direktori induk dari file yang ingin Anda kirim `StartFileTransfer`.

Jika Anda menggunakan otentikasi Dasar untuk AS2 konektor Anda, peran akses memerlukan `secretsmanager:GetSecretValue` izin untuk rahasia tersebut. Jika rahasia dienkripsi menggunakan kunci yang dikelola pelanggan alih-alih kunci terkelola di AWS Secrets Manager, maka peran tersebut juga memerlukan `kms:Decrypt` izin untuk kunci tersebut.

Untuk SFTP konektor

Pastikan bahwa peran akses menyediakan akses baca dan tulis ke direktori induk dari lokasi file yang digunakan dalam `StartFileTransfer` permintaan. Selain itu, pastikan bahwa peran tersebut memberikan `secretsmanager:GetSecretValue` izin untuk AWS Secrets Manager.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 2048.

Pola: `arn:.*role/\S+`

Wajib: Ya

[As2Config](#)

Struktur yang berisi parameter untuk objek AS2 konektor.

Tipe: Objek [As2ConnectorConfig](#)

Wajib: Tidak

[LoggingRole](#)

Amazon Resource Name (ARN) dari peran AWS Identity and Access Management (IAM) yang memungkinkan konektor mengaktifkan CloudWatch logging untuk peristiwa Amazon S3. Saat disetel, Anda dapat melihat aktivitas konektor di CloudWatch log Anda.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 2048.

Pola: `arn:.*role/\S+`

Wajib: Tidak

[SftpConfig](#)

Struktur yang berisi parameter untuk objek SFTP konektor.

Tipe: Objek [SftpConnectorConfig](#)

Wajib: Tidak

[Tags](#)

Pasangan nilai kunci yang dapat digunakan untuk mengelompokkan dan mencari konektor. Tag adalah metadata yang terpasang pada konektor untuk tujuan apa pun.

Tipe: Array objek [Tag](#)

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 50 item.

Wajib: Tidak

Url

URL Dari pasangan AS2 atau SFTP titik akhir.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum sebesar 255.

Wajib: Ya

Sintaksis Respons

```
{  
  "ConnectorId": "string"  
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

ConnectorId

Pengenal unik untuk konektor, dikembalikan setelah API panggilan berhasil.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: c-([0-9a-f]{17})

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTP Kode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceExistsException

Sumber daya yang diminta tidak ada, atau ada di wilayah selain yang ditentukan untuk perintah.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

ThrottlingException

Permintaan ditolak karena throttling permintaan.

HTTPKode Status: 400

Contoh

Contoh

Contoh berikut membuat AS2 konektor. Dalam perintah, ganti item sebagai berikut:

- `url`: menyediakan URL untuk AS2 server mitra dagang.
- `your-IAM-role-for-bucket-access`: IAM peran yang memiliki akses ke bucket Amazon S3 yang Anda gunakan untuk menyimpan file Anda.
- Gunakan peran ARN untuk pencatatan Anda, yang menyertakan Akun AWS ID Anda.
- Berikan jalur ke file yang berisi parameter konfigurasi AS2 konektor. Objek konfigurasi AS2 konektor dijelaskan dalam [As2 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
```

Contoh

Contoh berikut membuat SFTP konektor. Dalam perintah, ganti item sebagai berikut:

- `sftp-server-url`: menyediakan URL untuk SFTP server tempat Anda bertukar file.
- `your-IAM-role-for-bucket-access`: IAM peran yang memiliki akses ke bucket Amazon S3 yang Anda gunakan untuk menyimpan file Anda.
- Gunakan peran ARN untuk pencatatan Anda, yang menyertakan Akun AWS ID Anda.
- Berikan jalur ke file yang berisi parameter konfigurasi SFTP konektor. Objek konfigurasi SFTP konektor dijelaskan dalam [SftpConnectorConfig](#).

```
// 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
```

Contoh

API Panggilan mengembalikan ID konektor untuk konektor baru.

Contoh Respons

```
{  
  "ConnectorId": "a-11112222333344444"  
}
```

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

CreateProfile

Membuat profil lokal atau mitra untuk digunakan untuk AS2 transfer.

Sintaksis Permintaan

```
{
  "As2Id": "string",
  "CertificateIds": [ "string" ],
  "ProfileType": "string",
  "Tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ]
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[As2Id](#)

As2Id itu adalah AS2-nama, seperti yang didefinisikan dalam [RFC4130](#). Untuk transfer masuk, ini adalah AS2-From header untuk AS2 pesan yang dikirim dari mitra. Untuk konektor keluar, ini adalah AS2-To header untuk AS2 pesan yang dikirim ke mitra menggunakan StartFileTransfer API operasi. ID ini tidak dapat menyertakan spasi.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 128.

Pola: [\p{Print}\s]*

Wajib: Ya

[CertificateIds](#)

Array pengidentifikasi untuk sertifikat yang diimpor. Anda menggunakan pengenalan ini untuk bekerja dengan profil dan profil mitra.

Tipe: Array string

Kendala Panjang: Panjang tetap 22.

Pola: cert-([0-9a-f]{17})

Wajib: Tidak

[ProfileType](#)

Menentukan jenis profil yang akan dibuat:

- Tentukan LOCAL untuk membuat profil lokal. Profil lokal mewakili organisasi atau pihak server Transfer Family yang AS2 diaktifkan.
- Tentukan PARTNER untuk membuat profil mitra. Profil mitra mewakili organisasi jarak jauh, di luar Transfer Family.

Tipe: String

Nilai yang Valid: LOCAL | PARTNER

Wajib: Ya

[Tags](#)

Pasangan nilai kunci yang dapat digunakan untuk mengelompokkan dan mencari AS2 profil.

Tipe: Array objek [Tag](#)

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 50 item.

Diperlukan: Tidak

Sintaksis Respons

```
{  
  "ProfileId": "string"  
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

ProfileId

Pengidentifikasi unik untuk AS2 profil, dikembalikan setelah API panggilan berhasil.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: p-([0-9a-f]{17})

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

ThrottlingException

Permintaan ditolak karena throttling permintaan.

HTTPKode Status: 400

Contoh

Contoh

Contoh berikut membuat profil, dan mengembalikan ID profil.

Sertifikat IDs dibuat saat Anda menjalankan `import-certificate`, satu untuk sertifikat penandatanganan, dan satu untuk sertifikat enkripsi.

```
aws transfer create-profile --as2-id MYCORP --certificate-ids c-abcdefgh123456hijk  
c-987654aaaa321bbbb
```

Contoh Respons

API Panggilan mengembalikan ID profil untuk profil baru.

```
{  
  "ProfileId": "p-11112222333344444"  
}
```

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

CreateServer

Membuat instance server virtual auto-scaling berdasarkan protokol transfer file yang dipilih di. AWS Saat Anda melakukan pembaruan ke server berkemampuan protokol transfer file atau saat Anda bekerja dengan pengguna, gunakan `ServerId` properti yang dihasilkan layanan yang ditetapkan ke server yang baru dibuat.

Sintaksis Permintaan

```
{
  "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"
    }
  ]
}
}

```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[Certificate](#)

Nama Sumber Daya Amazon (ARN) dari sertifikat AWS Certificate Manager (ACM). Diperlukan saat Protocols diatur ke FTPS.

Untuk meminta sertifikat publik baru, lihat [Meminta sertifikat publik](#) di Panduan AWS Certificate Manager Pengguna.

Untuk mengimpor sertifikat yang ada ACM, lihat [Mengimpor sertifikat ke ACM](#) dalam Panduan AWS Certificate Manager Pengguna.

Untuk meminta sertifikat pribadi untuk digunakan FTPS melalui alamat IP pribadi, lihat [Meminta sertifikat pribadi](#) di Panduan AWS Certificate Manager Pengguna.

Sertifikat dengan algoritme kriptografi dan ukuran kunci berikut didukung:

- 2048-bit RSA (_2048) RSA
- 4096-bit RSA (_4096) RSA
- Elliptic Prime Curve 256 bit (EC_prime256v1)
- Elliptic Prime Curve 384 bit (EC_secp384r1)
- Elliptic Prime Curve 521 bit (EC_secp521r1)

 Note

Sertifikat harus berupa SSL TLS sertifikat/X.509 versi 3 yang valid dengan FQDN atau alamat IP yang ditentukan dan informasi tentang penerbit.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 1600.

Wajib: Tidak

Domain

Domain dari sistem penyimpanan yang digunakan untuk transfer file. Ada dua domain yang tersedia: Amazon Simple Storage Service (Amazon S3) dan Amazon Elastic File System (Amazon). EFS Nilai defaultnya adalah S3.

 Note

Setelah server dibuat, domain tidak dapat diubah.

Tipe: String

Nilai yang Valid: S3 | EFS

Wajib: Tidak

EndpointDetails

Pengaturan titik akhir virtual private cloud (VPC) yang dikonfigurasi untuk server Anda. Ketika Anda meng-host titik akhir Anda di dalam AndaVPC, Anda dapat membuat titik akhir Anda hanya dapat diakses oleh sumber daya di dalam AndaVPC, atau Anda dapat melampirkan alamat

IP Elastic dan membuat titik akhir Anda dapat diakses oleh klien melalui internet. Grup VPC keamanan default Anda secara otomatis ditetapkan ke titik akhir Anda.

Tipe: Objek [EndpointDetails](#)

Wajib: Tidak

[EndpointType](#)

Jenis endpoint yang Anda ingin server Anda gunakan. Anda dapat memilih untuk membuat endpoint server Anda dapat diakses secara publik (PUBLIC) atau menghostingnya di dalam server Anda. VPC Dengan endpoint yang di-host di aVPC, Anda dapat membatasi akses ke server dan sumber daya hanya di dalam Anda VPC atau memilih untuk membuatnya menghadap internet dengan melampirkan alamat IP Elastis langsung ke sana.

Note

Setelah 19 Mei 2021, Anda tidak akan dapat membuat server menggunakan `EndpointType=VPC_ENDPOINT` di akun Anda Akun AWS jika akun Anda belum melakukannya sebelum 19 Mei 2021. Jika Anda telah membuat server dengan `EndpointType=VPC_ENDPOINT` di Akun AWS pada atau sebelum 19 Mei 2021, Anda tidak akan terpengaruh. Setelah tanggal ini, gunakan `EndpointType =VPC`. Untuk informasi selengkapnya, lihat [Menghentikan penggunaan _ VPC ENDPOINT](#). Direkomendasikan agar Anda menggunakan VPC sebagai `EndpointType`. Dengan tipe endpoint ini, Anda memiliki opsi untuk secara langsung mengaitkan hingga tiga IPv4 alamat Elastic (termasuk BYO IP) dengan endpoint server Anda dan menggunakan grup VPC keamanan untuk membatasi lalu lintas dengan alamat IP publik klien. Hal ini tidak mungkin terjadi jika `EndpointType` diatur ke `VPC_ENDPOINT`.

Tipe: String

Nilai yang Valid: PUBLIC | VPC | VPC_ENDPOINT

Wajib: Tidak

[HostKey](#)

KunciRSA, ECDSA, atau ED25519 pribadi yang akan digunakan untuk server SFTP yang diaktifkan. Anda dapat menambahkan beberapa kunci host, jika Anda ingin memutar tombol, atau memiliki satu set kunci aktif yang menggunakan algoritma yang berbeda.

Gunakan perintah berikut untuk menghasilkan kunci RSA 2048 bit tanpa frasa sandi:

```
ssh-keygen -t rsa -b 2048 -N "" -m PEM -f my-new-server-key.
```

Gunakan nilai minimum 2048 untuk -b opsi. Anda dapat membuat kunci yang lebih kuat dengan menggunakan 3072 atau 4096.

Gunakan perintah berikut untuk menghasilkan kunci ECDSA 256 bit tanpa frasa sandi:

```
ssh-keygen -t ecdsa -b 256 -N "" -m PEM -f my-new-server-key.
```

Nilai yang valid untuk -b opsi ECDSA adalah 256, 384, dan 521.

Gunakan perintah berikut untuk menghasilkan ED25519 kunci tanpa frasa sandi:

```
ssh-keygen -t ed25519 -N "" -f my-new-server-key.
```

Untuk semua perintah ini, Anda dapat mengganti my-new-server-key dengan string pilihan Anda.

Important

Jika Anda tidak berencana untuk memigrasikan pengguna yang sudah ada dari server SFTP berkemampuan yang sudah ada ke server baru, jangan perbarui kunci host. Mengubah kunci host server secara tidak sengaja dapat mengganggu.

Untuk informasi selengkapnya, lihat [Memperbarui kunci host untuk server yang SFTP diaktifkan](#) di Panduan AWS Transfer Family Pengguna.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 4096.

Wajib: Tidak

[IdentityProviderDetails](#)

Diperlukan saat IdentityProviderType diatur ke AWS_DIRECTORY_SERVICE, AWS_LAMBDA atau API_GATEWAY. Menerima array yang berisi semua informasi yang diperlukan untuk menggunakan direktori di AWS_DIRECTORY_SERVICE atau memanggil otentikasi API yang disediakan pelanggan, termasuk Gateway. API URL Tidak diperlukan saat IdentityProviderType diatur ke SERVICE_MANAGED.

Tipe: Objek [IdentityProviderDetails](#)

Wajib: Tidak

[IdentityProviderType](#)

Modus otentikasi untuk server. Nilai defaultnya adalah `SERVICE_MANAGED`, yang memungkinkan Anda untuk menyimpan dan mengakses kredensi pengguna dalam layanan. AWS Transfer Family

Gunakan `AWS_DIRECTORY_SERVICE` untuk menyediakan akses ke grup Direktori Aktif di AWS Directory Service for Microsoft Active Directory atau Microsoft Active Directory di lingkungan lokal Anda atau AWS menggunakan AD Connector. Opsi ini juga mengharuskan Anda untuk memberikan ID Direktori dengan menggunakan `IdentityProviderDetails` parameter.

Gunakan nilai `API_GATEWAY` untuk mengintegrasikan dengan penyedia identitas pilihan Anda. `API_GATEWAY` mengharuskan Anda untuk menyediakan endpoint Amazon API Gateway URL untuk memanggil otentikasi dengan menggunakan parameter.

`IdentityProviderDetails`

Gunakan `AWS_LAMBDA` nilai untuk langsung menggunakan AWS Lambda fungsi sebagai penyedia identitas Anda. Jika Anda memilih nilai ini, Anda harus menentukan fungsi ARN untuk Lambda dalam `Function` parameter untuk tipe `IdentityProviderDetails` data.

Tipe: String

Nilai yang Valid: `SERVICE_MANAGED` | `API_GATEWAY` | `AWS_DIRECTORY_SERVICE` | `AWS_LAMBDA`

Wajib: Tidak

[LoggingRole](#)

Nama Sumber Daya Amazon (ARN) dari peran AWS Identity and Access Management (IAM) yang memungkinkan server mengaktifkan CloudWatch pencatatan Amazon untuk Amazon S3 atau Amazon. Setelah disetel, Anda dapat melihat aktivitas pengguna di CloudWatch log Anda.

Tipe: String

Batasan Panjang: Panjang minimum sebesar 0. Panjang maksimum 2048.

Pola: `(arn:.*role/\S+)`

Wajib: Tidak

[PostAuthenticationLoginBanner](#)

Menentukan string untuk ditampilkan ketika pengguna terhubung ke server. String ini ditampilkan setelah pengguna mengautentikasi.

Note

SFTPProtokol tidak mendukung spanduk tampilan pasca-otentikasi.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 4096.

Pola: `[\x09-\x0D\x20-\x7E]*`

Wajib: Tidak

[PreAuthenticationLoginBanner](#)

Menentukan string untuk ditampilkan ketika pengguna terhubung ke server. String ini ditampilkan sebelum pengguna mengautentikasi. Misalnya, spanduk berikut menampilkan detail tentang penggunaan sistem:

```
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.
```

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 4096.

Pola: `[\x09-\x0D\x20-\x7E]*`

Wajib: Tidak

[ProtocolDetails](#)

Pengaturan protokol yang dikonfigurasi untuk server Anda.

- Untuk menunjukkan mode pasif (untuk FTP dan FTPS protokol), gunakan parameter. `PassiveIp` Masukkan alamat dotted-quad tunggal, seperti IPv4 alamat IP eksternal firewall, router, atau penyeimbang beban.

- Untuk mengabaikan kesalahan yang dihasilkan saat klien mencoba menggunakan SETSTAT perintah pada file yang Anda unggah ke bucket Amazon S3, gunakan `SetStatOption` parameter. Agar AWS Transfer Family server mengabaikan SETSTAT perintah dan mengunggah file tanpa perlu membuat perubahan apa pun pada SFTP klien Anda, tetapkan nilainya. `ENABLE_NO_OP` Jika Anda menyetel `SetStatOption` parameternya `ENABLE_NO_OP`, Transfer Family akan menghasilkan entri CloudWatch log ke Amazon Logs, sehingga Anda dapat menentukan kapan klien melakukan SETSTAT panggilan.
- Untuk menentukan apakah AWS Transfer Family server Anda melanjutkan sesi terbaru yang dinegosiasikan melalui ID sesi unik, gunakan parameter. `TlsSessionResumptionMode`
- `As2Transports` menunjukkan metode transportasi untuk AS2 pesan. Saat ini, hanya HTTP didukung.

Tipe: Objek [ProtocolDetails](#)

Wajib: Tidak

[Protocols](#)

Menentukan protokol transfer file atau protokol di mana klien protokol transfer file Anda dapat terhubung ke titik akhir server Anda. Protokol yang tersedia adalah:

- SFTP(Secure Shell (SSH) Protokol Transfer File): Transfer file SSH
- FTPS(File Transfer Protocol Secure): Transfer file dengan TLS enkripsi
- FTP (Protokol Transfer File): Transfer file tidak terenkripsi
- AS2(Pernyataan Penerapan 2): digunakan untuk mengangkut data terstruktur business-to-business

Note

- Jika Anda memilih FTPS, Anda harus memilih sertifikat yang disimpan di AWS Certificate Manager (ACM) yang digunakan untuk mengidentifikasi server Anda ketika klien terhubung dengannya FTPS.
- Jika `Protocol` termasuk salah satu FTP atau FTPS, maka `EndpointType` harus VPC dan `IdentityProviderType` harus baik `AWS_DIRECTORY_SERVICE`, `AWS_LAMBDA`, atau `API_GATEWAY`.
- Jika `Protocol` termasuk FTP, maka `AddressAllocationIds` tidak dapat dikaitkan.
- Jika `Protocol` disetel hanya ke SFTP, `EndpointType` dapat diatur ke `PUBLIC` dan `IdentityProviderType` dapat disetel salah satu jenis identitas yang

didukung:SERVICE_MANAGED,AWS_DIRECTORY_SERVICE,AWS_LAMBDA, atauAPI_GATEWAY.

- Jika Protocol termasukAS2, maka EndpointType harusVPC, dan domain harus Amazon S3.

Tipe: Array string

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 4 item.

Nilai yang Valid: SFTP | FTP | FTPS | AS2

Wajib: Tidak

S3StorageOptions

Menentukan apakah atau tidak kinerja untuk direktori Amazon S3 Anda dioptimalkan. Ini dinonaktifkan secara default.

Secara default, pemetaan direktori home memiliki TYPE file. DIRECTORY Jika Anda mengaktifkan opsi ini, Anda kemudian perlu secara eksplisit menyetel HomeDirectoryMapEntry Type ke FILE jika Anda ingin pemetaan memiliki target file.

Tipe: Objek [S3StorageOptions](#)

Wajib: Tidak

SecurityPolicyName

Menentukan nama kebijakan keamanan yang dilampirkan ke server.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 100.

Pola: TransferSecurityPolicy-.*

Wajib: Tidak

StructuredLogDestinations

Menentukan grup log yang log server Anda dikirim.

Untuk menentukan grup log, Anda harus menyediakan ARN untuk grup log yang ada. Dalam hal ini, format grup log adalah sebagai berikut:

```
arn:aws:logs:region-name:amazon-account-id:log-group:log-group-name:*
```

Sebagai contoh, `arn:aws:logs:us-east-1:111122223333:log-group:mytestgroup:*`.

Jika sebelumnya Anda telah menentukan grup log untuk server, Anda dapat menghapusnya, dan pada dasarnya mematikan logging terstruktur, dengan memberikan nilai kosong untuk parameter ini dalam `update-server` panggilan. Sebagai contoh:

```
update-server --server-id s-1234567890abcdef0 --structured-log-destinations
```

Tipe: Array string.

Anggota Array: Jumlah minimum 0 item. Jumlah maksimum 1 item.

Batasan Panjang: Panjang minimum 20. Panjang maksimum 1600.

Pola: `arn:\S+`

Wajib: Tidak

[Tags](#)

Pasangan nilai kunci yang dapat digunakan untuk grup dan mencari server.

Tipe: Array objek [Tag](#)

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 50 item.

Wajib: Tidak

[WorkflowDetails](#)

Menentukan ID alur kerja untuk alur kerja yang akan ditetapkan dan peran eksekusi yang digunakan untuk mengeksekusi alur kerja.

Selain alur kerja untuk mengeksekusi ketika file diunggah sepenuhnya, juga `WorkflowDetails` dapat berisi ID alur kerja (dan peran eksekusi) untuk alur kerja untuk mengeksekusi pada upload sebagian. Upload sebagian terjadi ketika sesi server terputus saat file masih diunggah.

Tipe: Objek [WorkflowDetails](#)

Wajib: Tidak

Sintaksis Respons

```
{  
  "ServerId": "string"  
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

ServerId

Pengenal yang ditugaskan layanan dari server yang dibuat.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s-([0-9a-f]{17})

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

AccessDeniedException

Anda tidak memiliki akses yang memadai untuk melakukan tindakan ini.

HTTPKode Status: 400

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceExistsException

Sumber daya yang diminta tidak ada, atau ada di wilayah selain yang ditentukan untuk perintah.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

ThrottlingException

Permintaan ditolak karena throttling permintaan.

HTTPKode Status: 400

Contoh

Contoh

Contoh berikut membuat server baru menggunakan fileVPC_ENDPOINT.

Permintaan Sampel

```
{
  "EndpointType": "VPC",
  "EndpointDetails": ...,
  "HostKey": "Your RSA private key",
  "IdentityProviderDetails": "IdentityProvider",
  "IdentityProviderType": "SERVICE_MANAGED",
  "LoggingRole": "CloudWatchLoggingRole",
  "Tags": [
    {
      "Key": "Name",
```

```
    "Value": "MyServer"  
  }  
]  
}
```

Contoh

Ini adalah contoh respons untuk API panggilan ini.

Contoh Respons

```
{  
  "ServerId": "s-01234567890abcdef"  
}
```

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

CreateUser

Membuat pengguna dan mengaitkannya dengan server berkemampuan protokol transfer file yang ada. Anda hanya dapat membuat dan mengaitkan pengguna dengan server yang memiliki `IdentityProviderType` atur ke `SERVICE_MANAGED`. Dengan menggunakan parameter `forCreateUser`, Anda dapat menentukan nama pengguna, mengatur direktori home, menyimpan kunci publik pengguna, dan menetapkan peran AWS Identity and Access Management (IAM) pengguna. Anda juga dapat menambahkan kebijakan sesi secara opsional, dan menetapkan metadata dengan tag yang dapat digunakan untuk mengelompokkan dan mencari pengguna.

Sintaksis Permintaan

```
{
  "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"
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[HomeDirectory](#)

Direktori arahan (folder) untuk pengguna ketika mereka masuk ke server menggunakan klien.

Contoh HomeDirectory adalah `/bucket_name/home/mydirectory`.

Note

Parameter HomeDirectory hanya digunakan jika HomeDirectoryType diatur ke PATH.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 1024.

Pola: (| / . *)

Wajib: Tidak

[HomeDirectoryMappings](#)

Pemetaan direktori logis yang menentukan jalur dan kunci Amazon S3 atau EFS Amazon apa yang harus terlihat oleh pengguna Anda dan bagaimana Anda ingin membuatnya terlihat. Anda harus menentukan Entry dan Target memasangkan, di mana Entry menunjukkan bagaimana jalur dibuat terlihat dan Target merupakan jalur Amazon S3 atau Amazon yang sebenarnya. EFS Jika Anda hanya menentukan target, itu ditampilkan apa adanya. Anda juga harus memastikan bahwa peran AWS Identity and Access Management (IAM) Anda menyediakan akses ke jalur diTarget. Nilai ini dapat diatur hanya ketika HomeDirectoryType diatur ke LOGICAL.

Berikut ini adalah contoh Entry dan Target pair.

```
[ { "Entry": "/directory1", "Target": "/bucket_name/home/mydirectory" } ]
```

Dalam kebanyakan kasus, Anda dapat menggunakan nilai ini alih-alih kebijakan sesi untuk mengunci pengguna Anda ke direktori home yang ditunjuk (`chroot`). Untuk melakukan ini, Anda

dapat mengatur Entry ke / dan mengatur Target ke nilai yang harus dilihat pengguna untuk direktori home mereka ketika mereka masuk.

Berikut ini adalah contoh Entry dan Target pair untuk chroot.

```
[ { "Entry": "/", "Target": "/bucket_name/home/mydirectory" } ]
```

Tipe: Array objek [HomeDirectoryMapEntry](#)

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 50000 item.

Wajib: Tidak

[HomeDirectoryType](#)

Jenis direktori pendaratan (folder) yang Anda inginkan direktori home pengguna Anda ketika mereka masuk ke server. Jika Anda mengaturnya PATH, pengguna akan melihat bucket Amazon S3 absolut atau EFS jalur Amazon seperti pada klien protokol transfer file mereka. Jika Anda menyetelnya LOGICAL, Anda harus menyediakan pemetaan HomeDirectoryMappings untuk bagaimana Anda ingin membuat jalur Amazon S3 atau EFS Amazon terlihat oleh pengguna Anda.

Note

Jika HomeDirectoryType ya LOGICAL, Anda harus memberikan pemetaan, menggunakan parameter. HomeDirectoryMappings Jika, di sisi lain, HomeDirectoryType adalah PATH, Anda memberikan jalur absolut menggunakan HomeDirectory parameter. Anda tidak dapat memiliki keduanya HomeDirectory dan HomeDirectoryMappings di template Anda.

Tipe: String

Nilai yang Valid: PATH | LOGICAL

Wajib: Tidak

[Policy](#)

Kebijakan sesi untuk pengguna Anda sehingga Anda dapat menggunakan peran AWS Identity and Access Management (IAM) yang sama di beberapa pengguna. Kebijakan ini mencakup akses pengguna ke sebagian bucket Amazon S3 mereka. Variabel yang dapat Anda gunakan

dalam kebijakan ini meliputi `${Transfer:UserName}`, `${Transfer:HomeDirectory}`, dan `${Transfer:HomeBucket}`.

 Note

Kebijakan ini hanya berlaku jika domainnya `ServerId` adalah Amazon S3. Amazon EFS tidak menggunakan kebijakan sesi.

Untuk kebijakan sesi, AWS Transfer Family menyimpan kebijakan sebagai JSON gumpalan, bukan Amazon Resource Name (ARN) kebijakan. Anda menyimpan kebijakan sebagai JSON gumpalan dan meneruskannya dalam `Policy` argumen.

Untuk contoh kebijakan sesi, lihat [Contoh kebijakan sesi](#).

Untuk informasi selengkapnya, lihat [AssumeRole](#) di API Referensi Layanan Token AWS Keamanan.

Tipe: String

Batasan Panjang: Panjang minimum sebesar 0. Panjang maksimum 2048.

Wajib: Tidak

[PosixProfile](#)

Menentukan POSIX identitas lengkap, termasuk ID pengguna (`Uid`), ID grup (`Gid`), dan grup sekunder apa pun IDs (`SecondaryGids`), yang mengontrol akses pengguna ke sistem EFS file Amazon Anda. POSIX izin yang ditetapkan pada file dan direktori di Amazon EFS menentukan tingkat akses yang didapat pengguna Anda saat mentransfer file masuk dan keluar dari sistem file Amazon EFS Anda.

Tipe: Objek [PosixProfile](#)

Wajib: Tidak

[Role](#)

Nama Sumber Daya Amazon (ARN) dari peran AWS Identity and Access Management (IAM) yang mengontrol akses pengguna ke bucket Amazon S3 atau sistem file EFS Amazon. Kebijakan yang dilampirkan pada peran ini menentukan tingkat akses yang ingin Anda berikan kepada pengguna saat mentransfer file masuk dan keluar dari bucket Amazon S3 atau sistem file EFS Amazon Anda. IAM Peran tersebut juga harus berisi hubungan kepercayaan yang memungkinkan server mengakses sumber daya Anda saat melayani permintaan transfer pengguna Anda.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 2048.

Pola: `arn:.*role/\S+`

Wajib: Ya

ServerId

Pengidentifikasi unik yang ditetapkan sistem untuk instans server. Ini adalah server tertentu tempat Anda menambahkan pengguna.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: `s-([0-9a-f]{17})`

Wajib: Ya

SshPublicKeyBody

Bagian publik dari kunci Secure Shell (SSH) yang digunakan untuk mengautentikasi pengguna ke server.

Tiga elemen format kunci SSH publik standar adalah `<key type>`, `<body base64>`, dan opsional `<comment>`, dengan spasi di antara setiap elemen.

AWS Transfer Family menerima RSA, ECDSA, dan ED25519 kunci.

- Untuk RSA kunci, tipe kuncinya adalah `ssh-rsa`.
- Untuk ED25519 kunci, tipe kuncinya adalah `ssh-ed25519`.
- Untuk ECDSA kunci, jenis kuncinya adalah `ecdsa-sha2-nistp256`, `ecdsa-sha2-nistp384`, atau `ecdsa-sha2-nistp521`, tergantung pada ukuran kunci yang Anda hasilkan.

Tipe: String

Batasan Panjang: Panjang minimum sebesar 0. Panjang maksimum 2048.

Wajib: Tidak

Tags

Pasangan nilai-kunci yang dapat digunakan untuk grup dan mencari pengguna. Tanda adalah metadata yang dilampirkan ke pengguna untuk tujuan apa pun.

Tipe: Array objek [Tag](#)

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 50 item.

Wajib: Tidak

[UserName](#)

String unik yang mengidentifikasi pengguna dan dikaitkan dengan file. `ServerId` Nama pengguna ini harus sepanjang minimal 3 dan maksimal 100 karakter. Berikut adalah karakter yang valid: a-z, A-Z, 0-9, garis bawah '_', tanda hubung '-', titik '.', dan tanda at '@'. Nama pengguna tidak dapat dimulai dengan tanda hubung, titik, atau tanda at.

Tipe: String

Batasan Panjang: Panjang minimum 3. Panjang maksimum 100.

Pola: `[\w][\w@.-]{2,99}`

Diperlukan: Ya

Sintaksis Respons

```
{
  "ServerId": "string",
  "UserName": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

[ServerId](#)

Pengidentifikasi server tempat pengguna dilampirkan.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: `s-([0-9a-f]{17})`

UserName

String unik yang mengidentifikasi pengguna Transfer Family.

Tipe: String

Batasan Panjang: Panjang minimum 3. Panjang maksimum 100.

Pola: `[\w][\w@.-]{2,99}`

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceExistsException

Sumber daya yang diminta tidak ada, atau ada di wilayah selain yang ditentukan untuk perintah.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Contoh

Contoh

Untuk membuat pengguna, pertama-tama Anda dapat menyimpan parameter ke dalam JSON file, misalnya `createUserParameters`, lalu jalankan perintah API `create-user`.

```
{
  "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"
}
```

Permintaan Sampel

```
aws transfer create-user --cli-input-json file://createUserParameters
```

Contoh Respons

```
{
  "ServerId": "s-1111aaaa2222bbbb3",
  "UserName": "bobusa-API"
}
```

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)

- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

CreateWorkflow

Memungkinkan Anda membuat alur kerja dengan langkah-langkah tertentu dan detail langkah yang dipanggil alur kerja setelah transfer file selesai. Setelah membuat alur kerja, Anda dapat mengaitkan alur kerja yang dibuat dengan server transfer apa pun dengan menentukan workflow-details bidang CreateServer dan operasi. UpdateServer

Sintaksis Permintaan

```
{
  "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"
  }
]
}

```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[Description](#)

Deskripsi tekstual untuk alur kerja.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 256.

Pola: `[\w- ]*`

Wajib: Tidak

[OnExceptionSteps](#)

Menentukan langkah-langkah (tindakan) untuk mengambil jika kesalahan ditemui selama pelaksanaan alur kerja.

Note

Untuk langkah-langkah khusus, fungsi Lambda perlu mengirim FAILURE ke panggilan kembali API untuk memulai langkah pengecualian. Selain itu, jika Lambda tidak mengirim SUCCESS sebelum waktu habis, langkah-langkah pengecualian dijalankan.

Tipe: Array objek [WorkflowStep](#)

Anggota Array: Jumlah minimum 0 item. Jumlah maksimum 8 item.

Wajib: Tidak

[Steps](#)

Menentukan rincian untuk langkah-langkah yang ada dalam alur kerja yang ditentukan.

TYPE Menentukan mana dari tindakan berikut yang sedang diambil untuk langkah ini.

- **COPY**- Salin file ke lokasi lain.

- **CUSTOM**- Lakukan langkah kustom dengan target AWS Lambda fungsi.
- **DECRYPT**- Dekripsi file yang dienkripsi sebelum diunggah.
- **DELETE**- Hapus file.
- **TAG**- Tambahkan tag ke file.

 Note

Saat ini, penyalinan dan penandaan hanya didukung pada S3.

Untuk lokasi file, Anda menentukan bucket dan kunci Amazon S3, atau ID dan jalur sistem EFS file Amazon.

Tipe: Array objek [WorkflowStep](#)

Anggota Array: Jumlah minimum 0 item. Jumlah maksimum 8 item.

Wajib: Ya

[Tags](#)

Pasangan nilai kunci yang dapat digunakan untuk mengelompokkan dan mencari alur kerja. Tag adalah metadata yang dilampirkan ke alur kerja untuk tujuan apa pun.

Tipe: Array objek [Tag](#)

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 50 item.

Diperlukan: Tidak

Sintaksis Respons

```
{  
  "WorkflowId": "string"  
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

WorkflowId

Pengidentifikasi unik untuk alur kerja.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: w-([a-z0-9]{17})

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

AccessDeniedException

Anda tidak memiliki akses yang memadai untuk melakukan tindakan ini.

HTTPKode Status: 400

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceExistsException

Sumber daya yang diminta tidak ada, atau ada di wilayah selain yang ditentukan untuk perintah.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

ThrottlingException

Permintaan ditolak karena throttling permintaan.

HTTPKode Status: 400

Contoh

Contoh

Anda dapat menyimpan informasi langkah alur kerja ke dalam file teks, dan kemudian menggunakan file itu untuk membuat alur kerja, seperti pada contoh berikut. Contoh berikut mengasumsikan Anda telah menyimpan langkah-langkah alur kerja Anda ke *example-file.json* (dalam folder yang sama dari tempat Anda menjalankan perintah), dan bahwa Anda ingin membuat alur kerja di wilayah Virginia N. (us-east-1).

```
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}"
    }
}
]
```

Contoh

CreateWorkflowPanggilan mengembalikan ID alur kerja untuk alur kerja baru.

Contoh Respons

```
{
  "WorkflowId": "w-1234abcd5678efghi"
}
```

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

DeleteAccess

Memungkinkan Anda menghapus akses yang ditentukan dalam ExternalID parameter ServerID dan.

Sintaksis Permintaan

```
{  
  "ExternalId": "string",  
  "ServerId": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

ExternalId

Pengidentifikasi unik yang diperlukan untuk mengidentifikasi grup tertentu dalam direktori Anda. Pengguna grup yang Anda asosiasikan memiliki akses ke EFS sumber daya Amazon S3 atau Amazon Anda melalui protokol yang diaktifkan. AWS Transfer Family Jika Anda tahu nama grup, Anda dapat melihat SID nilai dengan menjalankan perintah berikut menggunakan Windows PowerShell.

```
Get-ADGroup -Filter {samAccountName -like "YourGroupName*"} -Properties  
* | Select SamAccountName,ObjectSid
```

Dalam perintah itu, ganti YourGroupName dengan nama grup Active Directory Anda.

Ekspresi reguler yang digunakan untuk memvalidasi parameter ini adalah string karakter yang terdiri dari huruf besar dan huruf kecil karakter alfanumerik tanpa spasi. Anda juga dapat menyertakan garis bawah atau salah satu karakter berikut: =, . @: /-

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 256.

Pola: S-1-[\d-]+

Wajib: Ya

ServerId

Pengidentifikasi unik yang ditetapkan sistem untuk server yang telah ditetapkan pengguna ini.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s-([0-9a-f]{17})

Diperlukan: Ya

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200 dengan HTTP tubuh kosong.

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

DeleteAgreement

Hapus perjanjian yang ditentukan dalam yang disediakan `AgreementId`.

Sintaksis Permintaan

```
{  
  "AgreementId": "string",  
  "ServerId": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

AgreementId

Pengidentifikasi unik untuk perjanjian. Pengenal ini dikembalikan saat Anda membuat perjanjian.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: a-([0-9a-f]{17})

Wajib: Ya

ServerId

Pengenal server yang terkait dengan perjanjian yang Anda hapus.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s-([0-9a-f]{17})

Diperlukan: Ya

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200 dengan HTTP tubuh kosong.

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)

- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

DeleteCertificate

Menghapus sertifikat yang ditentukan dalam `CertificateId` parameter.

Sintaksis Permintaan

```
{  
  "CertificateId": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

CertificateId

Pengidentifikasi objek sertifikat yang Anda hapus.

Tipe: String

Kendala Panjang: Panjang tetap 22.

Pola: `cert-([0-9a-f]{17})`

Diperlukan: Ya

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200 dengan HTTP tubuh kosong.

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

DeleteConnector

Menghapus konektor yang ditentukan dalam yang disediakan `ConnectorId`.

Sintaksis Permintaan

```
{  
  "ConnectorId": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

ConnectorId

Pengidentifikasi unik untuk konektor.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: `c-([0-9a-f]{17})`

Diperlukan: Ya

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200 dengan HTTP tubuh kosong.

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

DeleteHostKey

Menghapus kunci host yang ditentukan dalam HostKeyId parameter.

Sintaksis Permintaan

```
{  
  "HostKeyId": "string",  
  "ServerId": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[HostKeyId](#)

Pengidentifikasi kunci host yang Anda hapus.

Tipe: String

Kendala Panjang: Panjang tetap 25.

Pola: hostkey-[0-9a-f]{17}

Wajib: Ya

[ServerId](#)

Pengidentifikasi server yang berisi kunci host yang Anda hapus.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s-([0-9a-f]{17})

Diperlukan: Ya

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200 dengan HTTP tubuh kosong.

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

ThrottlingException

Permintaan ditolak karena throttling permintaan.

HTTPKode Status: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

DeleteProfile

Menghapus profil yang ditentukan dalam `ProfileId` parameter.

Sintaksis Permintaan

```
{  
  "ProfileId": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

ProfileId

Pengidentifikasi profil yang Anda hapus.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: `p-([0-9a-f]{17})`

Diperlukan: Ya

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200 dengan HTTP tubuh kosong.

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

DeleteServer

Menghapus server berkemampuan protokol transfer file yang Anda tentukan.

Tidak ada tanggapan yang kembali dari operasi ini.

Sintaksis Permintaan

```
{  
  "ServerId": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[ServerId](#)

Pengidentifikasi unik yang ditetapkan sistem untuk instance server.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s-([0-9a-f]{17})

Diperlukan: Ya

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200 dengan HTTP tubuh kosong.

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

AccessDeniedException

Anda tidak memiliki akses yang memadai untuk melakukan tindakan ini.

HTTPKode Status: 400

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Contoh

Contoh

Contoh berikut menghapus server.

Permintaan Sampel

```
{  
  "ServerId": "s-01234567890abcdef"  
}
```

Contoh

Jika berhasil, tidak ada yang dikembalikan.

Contoh Respons

```
{  
}
```

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

DeleteSshPublicKey

Menghapus kunci publik Secure Shell (SSH) pengguna.

Sintaksis Permintaan

```
{
  "ServerId": "string",
  "SshPublicKeyId": "string",
  "UserName": "string"
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[ServerId](#)

Pengidentifikasi unik yang ditetapkan sistem untuk instance server berkemampuan protokol transfer file yang memiliki pengguna yang ditugaskan padanya.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s-([0-9a-f]{17})

Wajib: Ya

[SshPublicKeyId](#)

Pengenal unik yang digunakan untuk mereferensikan SSH kunci spesifik pengguna Anda.

Tipe: String

Kendala Panjang: Panjang tetap 21.

Pola: key-[0-9a-f]{17}

Wajib: Ya

UserName

String unik yang mengidentifikasi pengguna yang kunci publiknya sedang dihapus.

Tipe: String

Batasan Panjang: Panjang minimum 3. Panjang maksimum 100.

Pola: `[\w][\w@.-]{2,99}`

Diperlukan: Ya

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200 dengan HTTP tubuh kosong.

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

ThrottlingException

Permintaan ditolak karena throttling permintaan.

HTTPKode Status: 400

Contoh

Contoh

Contoh berikut menghapus kunci SSH publik pengguna.

Permintaan Sampel

```
{
  "ServerId": "s-01234567890abcdef",
  "SshPublicKeyId": "MyPublicKey",
  "UserName": "my_user"
}
```

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

DeleteUser

Menghapus pengguna milik server berkemampuan protokol transfer file yang Anda tentukan.

Tidak ada tanggapan yang kembali dari operasi ini.

Note

Ketika Anda menghapus pengguna dari server, informasi pengguna hilang.

Sintaksis Permintaan

```
{  
  "ServerId": "string",  
  "UserName": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

ServerId

Pengidentifikasi unik yang ditetapkan sistem untuk instance server yang memiliki pengguna yang ditugaskan padanya.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s-([0-9a-f]{17})

Wajib: Ya

UserName

String unik yang mengidentifikasi pengguna yang sedang dihapus dari server.

Tipe: String

Batasan Panjang: Panjang minimum 3. Panjang maksimum 100.

Pola: `[\w][\w@.-]{2,99}`

Diperlukan: Ya

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200 dengan HTTP tubuh kosong.

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Contoh

Contoh

Contoh berikut menghapus pengguna Transfer Family.

Permintaan Sampel

```
{
  "ServerId": "s-01234567890abcdef",
  "UserNames": "my_user"
}
```

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

DeleteWorkflow

Menghapus alur kerja yang ditentukan.

Sintaksis Permintaan

```
{  
  "WorkflowId": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[WorkflowId](#)

Pengidentifikasi unik untuk alur kerja.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: w-([a-z0-9]{17})

Diperlukan: Ya

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200 dengan HTTP tubuh kosong.

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

AccessDeniedException

Anda tidak memiliki akses yang memadai untuk melakukan tindakan ini.

HTTPKode Status: 400

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

DescribeAccess

Menjelaskan akses yang ditetapkan ke server berkemampuan protokol transfer file tertentu, seperti yang diidentifikasi oleh `ServerId` propertinya dan miliknya. `ExternalId`

Respons dari panggilan ini mengembalikan properti akses yang terkait dengan `ServerId` nilai yang ditentukan.

Sintaksis Permintaan

```
{
  "ExternalId": "string",
  "ServerId": "string"
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

ExternalId

Pengidentifikasi unik yang diperlukan untuk mengidentifikasi grup tertentu dalam direktori Anda. Pengguna grup yang Anda kaitkan memiliki akses ke EFS sumber daya Amazon S3 atau Amazon Anda melalui protokol yang diaktifkan. AWS Transfer Family Jika Anda tahu nama grup, Anda dapat melihat SID nilai dengan menjalankan perintah berikut menggunakan Windows PowerShell.

```
Get-ADGroup -Filter {samAccountName -like "YourGroupName*"} -Properties
* | Select SamAccountName, ObjectSid
```

Dalam perintah itu, ganti `YourGroupName` dengan nama grup Active Directory Anda.

Ekspresi reguler yang digunakan untuk memvalidasi parameter ini adalah string karakter yang terdiri dari huruf besar dan huruf kecil karakter alfanumerik tanpa spasi. Anda juga dapat menyertakan garis bawah atau salah satu karakter berikut: `=`, `.`, `@`: `/-`

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 256.

Pola: `S-1-[\d-]+`

Wajib: Ya

ServerId

Pengidentifikasi unik yang ditetapkan sistem untuk server yang memiliki akses ini ditetapkan.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s-([0-9a-f]{17})

Diperlukan: Ya

Sintaksis Respons

```
{
  "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"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

Access

Pengidentifikasi eksternal server tempat akses dilampirkan.

Tipe: Objek [DescribedAccess](#)

ServerId

Pengidentifikasi unik yang ditetapkan sistem untuk server yang memiliki akses ini ditetapkan.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s - ([0-9a-f]{17})

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

DescribeAgreement

Menjelaskan perjanjian yang diidentifikasi oleh `AgreementId`.

Sintaksis Permintaan

```
{  
  "AgreementId": "string",  
  "ServerId": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[AgreementId](#)

Pengidentifikasi unik untuk perjanjian. Pengenal ini dikembalikan saat Anda membuat perjanjian.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: a-([0-9a-f]{17})

Wajib: Ya

[ServerId](#)

Pengidentifikasi server yang terkait dengan perjanjian.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s-([0-9a-f]{17})

Diperlukan: Ya

Sintaksis Respons

```
{
```

```

"Agreement": {
  "AccessRole": "string",
  "AgreementId": "string",
  "Arn": "string",
  "BaseDirectory": "string",
  "Description": "string",
  "LocalProfileId": "string",
  "PartnerProfileId": "string",
  "ServerId": "string",
  "Status": "string",
  "Tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ]
}

```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

Agreement

Rincian untuk perjanjian yang ditentukan, dikembalikan sebagai `DescribedAgreement` objek.

Tipe: Objek [DescribedAgreement](#)

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

DescribeCertificate

Menjelaskan sertifikat yang diidentifikasi oleh `CertificateId`.

Sintaksis Permintaan

```
{  
  "CertificateId": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

CertificateId

Array pengidentifikasi untuk sertifikat yang diimpor. Anda menggunakan pengenal ini untuk bekerja dengan profil dan profil mitra.

Tipe: String

Kendala Panjang: Panjang tetap 22.

Pola: cert-([0-9a-f]{17})

Diperlukan: Ya

Sintaksis Respons

```
{  
  "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"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

Certificate

Rincian untuk sertifikat yang ditentukan, dikembalikan sebagai objek.

Tipe: Objek [DescribedCertificate](#)

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

DescribeConnector

Menjelaskan konektor yang diidentifikasi oleh ConnectorId.

Sintaksis Permintaan

```
{  
  "ConnectorId": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

ConnectorId

Pengidentifikasi unik untuk konektor.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: c-([0-9a-f]{17})

Diperlukan: Ya

Sintaksis Respons

```
{  
  "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",
  "ServiceManagedEgressIpAddresses": [ "string" ],
  "SftpConfig": {
    "TrustedHostKeys": [ "string" ],
    "UserSecretId": "string"
  },
  "Tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ],
  "Url": "string"
}

```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

Connector

Struktur yang berisi rincian konektor.

Tipe: Objek [DescribedConnector](#)

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

DescribeExecution

Anda dapat menggunakan DescribeExecution untuk memeriksa rincian eksekusi alur kerja yang ditentukan.

Note

API Panggilan ini hanya menampilkan detail untuk alur kerja yang sedang berlangsung. Jika Anda memberikan ID untuk eksekusi yang tidak sedang berlangsung, atau jika eksekusi tidak cocok dengan ID alur kerja yang ditentukan, Anda akan menerima ResourceNotFound pengecualian.

Sintaksis Permintaan

```
{  
  "ExecutionId": "string",  
  "WorkflowId": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

ExecutionId

Pengidentifikasi unik untuk eksekusi alur kerja.

Tipe: String

Batas Panjang: Panjang tetap 36.

Pola: `[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}`

Wajib: Ya

WorkflowId

Pengidentifikasi unik untuk alur kerja.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: w-([a-z0-9]{17})

Diperlukan: Ya

Sintaksis Respons

```
{
  "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"
}

```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

Execution

Struktur yang berisi rincian eksekusi alur kerja.

Tipe: Objek [DescribedExecution](#)

WorkflowId

Pengidentifikasi unik untuk alur kerja.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: w-([a-z0-9]{17})

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)

- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

DescribeHostKey

Mengembalikan rincian kunci host yang ditentukan oleh `HostKeyId` dan `ServerId`.

Sintaksis Permintaan

```
{  
  "HostKeyId": "string",  
  "ServerId": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[HostKeyId](#)

Pengidentifikasi kunci host yang ingin Anda jelaskan.

Tipe: String

Kendala Panjang: Panjang tetap 25.

Pola: `hostkey-[0-9a-f]{17}`

Wajib: Ya

[ServerId](#)

Pengidentifikasi server yang berisi kunci host yang ingin Anda jelaskan.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: `s-([0-9a-f]{17})`

Diperlukan: Ya

Sintaksis Respons

```
{
```

```
"HostKey": {
  "Arn": "string",
  "DateImported": number,
  "Description": "string",
  "HostKeyFingerprint": "string",
  "HostKeyId": "string",
  "Tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ],
  "Type": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

[HostKey](#)

Mengembalikan rincian untuk kunci host yang ditentukan.

Tipe: Objek [DescribedHostKey](#)

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPCode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPCode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

DescribeProfile

Mengembalikan rincian profil yang ditentukan oleh `ProfileId`.

Sintaksis Permintaan

```
{  
  "ProfileId": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

ProfileId

Pengidentifikasi profil yang ingin Anda jelaskan.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: `p-([0-9a-f]{17})`

Diperlukan: Ya

Sintaksis Respons

```
{  
  "Profile": {  
    "Arn": "string",  
    "As2Id": "string",  
    "CertificateIds": [ "string" ],  
    "ProfileId": "string",  
    "ProfileType": "string",  
    "Tags": [  
      {  
        "Key": "string",  
        "Value": "string"  
      }  
    ]  
  }  
}
```

```
]
}
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

[Profile](#)

Rincian profil yang ditentukan, dikembalikan sebagai objek.

Tipe: Objek [DescribedProfile](#)

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

DescribeSecurityPolicy

Menjelaskan kebijakan keamanan yang dilampirkan ke server berkemampuan protokol transfer file Anda. Respons berisi deskripsi properti kebijakan keamanan. Untuk informasi selengkapnya tentang kebijakan keamanan, lihat [Bekerja dengan kebijakan keamanan](#).

Sintaksis Permintaan

```
{  
  "SecurityPolicyName": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[SecurityPolicyName](#)

Menentukan nama kebijakan keamanan yang dilampirkan ke server.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 100.

Pola: TransferSecurityPolicy-.*

Diperlukan: Ya

Sintaksis Respons

```
{  
  "SecurityPolicy": {  
    "Fips": boolean,  
    "SecurityPolicyName": "string",  
    "SshCiphers": [ "string" ],  
    "SshKexs": [ "string" ],  
    "SshMacs": [ "string" ],  
    "TlsCiphers": [ "string" ]  
  }  
}
```

```
}  
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

[SecurityPolicy](#)

Array yang berisi properti kebijakan keamanan.

Tipe: Objek [DescribedSecurityPolicy](#)

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Contoh

Contoh

Perintah contoh berikut mengambil nama kebijakan keamanan sebagai argumen, dan mengembalikan algoritme untuk kebijakan keamanan yang ditentukan.

Permintaan Sampel

```
aws transfer describe-security-policy --security-policy-name "TransferSecurityPolicy-FIPS-2023-05"
```

Contoh Respons

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

```
}  
}
```

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

DescribeServer

Menjelaskan server berkemampuan protokol transfer file yang Anda tentukan dengan meneruskan parameter. `ServerId`

Respons berisi deskripsi properti server. Ketika Anda mengatur `EndpointType` ke VPC, respon akan berisi `EndpointDetails`.

Sintaksis Permintaan

```
{
  "ServerId": "string"
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

ServerId

Pengidentifikasi unik yang ditetapkan sistem untuk server.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: `s-([0-9a-f]{17})`

Diperlukan: Ya

Sintaksis Respons

```
{
  "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"
    }
  ]
}
}
}

```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

Server

Array yang berisi properti server dengan yang `ServerID` Anda tentukan.

Tipe: Objek [DescribedServer](#)

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPCode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPCode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Contoh

Contoh

Contoh berikut mengembalikan properti yang ditugaskan ke server.

Permintaan Sampel

```
{
  "ServerId": "s-01234567890abcdef"
}
```

Contoh

Contoh ini menggambarkan salah satu penggunaan. DescribeServer

Contoh Respons

```
{
  "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  
  }  
}
```

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

DescribeUser

Menjelaskan pengguna yang ditugaskan ke server berkemampuan protokol transfer file tertentu, seperti yang diidentifikasi oleh properti `ServerId`.

Respons dari panggilan ini mengembalikan properti pengguna yang terkait dengan `ServerId` nilai yang ditentukan.

Sintaksis Permintaan

```
{  
  "ServerId": "string",  
  "UserName": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

ServerId

Pengidentifikasi unik yang ditetapkan sistem untuk server yang telah ditetapkan pengguna ini.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: `s-([0-9a-f]{17})`

Wajib: Ya

UserName

Nama pengguna yang ditetapkan ke satu atau lebih server. Nama pengguna adalah bagian dari kredensi masuk untuk menggunakan AWS Transfer Family layanan dan melakukan tugas transfer file.

Tipe: String

Batasan Panjang: Panjang minimum 3. Panjang maksimum 100.

Pola: `[\w][\w@.-]{2,99}`

Diperlukan: Ya

Sintaksis Respons

```
{
  "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"
  }
}
```

```
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

ServerId

Pengidentifikasi unik yang ditetapkan sistem untuk server yang telah ditetapkan pengguna ini.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s-([0-9a-f]){17})

User

Array yang berisi properti pengguna Transfer Family untuk ServerID nilai yang Anda tentukan.

Tipe: Objek [DescribedUser](#)

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Contoh

Contoh

Contoh berikut menunjukkan rincian untuk pengguna yang ada.

Permintaan Sampel

```
aws transfer describe-user --server-id s-1111aaaa2222bbbb3 --user-name bob-test
```

Contoh Respons

```
{
  "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"
  }
}
```

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

DescribeWorkflow

Menjelaskan alur kerja yang ditentukan.

Sintaksis Permintaan

```
{  
  "WorkflowId": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[WorkflowId](#)

Pengidentifikasi unik untuk alur kerja.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: w-([a-z0-9]{17})

Diperlukan: Ya

Sintaksis Respons

```
{  
  "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"
}
}

```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

Workflow

Struktur yang berisi rincian alur kerja.

Tipe: Objek [DescribedWorkflow](#)

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServiceError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

ImportCertificate

Mengimpor sertifikat penandatanganan dan enkripsi yang Anda perlukan untuk membuat profil lokal (AS2) dan profil mitra.

Sintaksis Permintaan

```
{
  "ActiveDate": number,
  "Certificate": "string",
  "CertificateChain": "string",
  "Description": "string",
  "InactiveDate": number,
  "PrivateKey": "string",
  "Tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ],
  "Usage": "string"
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[ActiveDate](#)

Tanggal opsional yang menentukan kapan sertifikat menjadi aktif.

Tipe: Timestamp

Wajib: Tidak

[Certificate](#)

- Untuk ituCLI, berikan jalur file untuk sertifikat dalam URI format. Misalnya, `--certificate file://encryption-cert.pem`. Atau, Anda dapat memberikan konten mentah.
- Untuk SDK, tentukan konten mentah dari file sertifikat. Misalnya, `--certificate "`cat encryption-cert.pem`"`.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 16384.

Pola: `[\u0009\u000A\u000D\u0020-\u00FF]*`

Wajib: Ya

[CertificateChain](#)

Daftar opsional sertifikat yang membentuk rantai untuk sertifikat yang sedang diimpor.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 2097152.

Pola: `[\u0009\u000A\u000D\u0020-\u00FF]*`

Wajib: Tidak

[Description](#)

Deskripsi singkat yang membantu mengidentifikasi sertifikat.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 200.

Pola: `[\p{Graph}]+`

Wajib: Tidak

[InactiveDate](#)

Tanggal opsional yang menentukan kapan sertifikat menjadi tidak aktif.

Tipe: Timestamp

Wajib: Tidak

[PrivateKey](#)

- UntukCLI, berikan jalur file untuk kunci pribadi dalam URI format.Misalnya, `--private-key file://encryption-key.pem` Atau, Anda dapat memberikan konten mentah dari file kunci pribadi.
- UntukSDK, tentukan konten mentah dari file kunci pribadi. Sebagai contoh, `--private-key "`cat encryption-key.pem`"`.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 16384.

Pola: `[\u0009\u000A\u000D\u0020-\u00FF]*`

Wajib: Tidak

Tags

Pasangan nilai kunci yang dapat digunakan untuk mengelompokkan dan mencari sertifikat.

Tipe: Array objek [Tag](#)

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 50 item.

Wajib: Tidak

Usage

Menentukan apakah sertifikat ini digunakan untuk penandatanganan atau enkripsi.

Tipe: String

Nilai yang Valid: SIGNING | ENCRYPTION

Wajib: Ya

Sintaksis Respons

```
{  
  "CertificateId": "string"  
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

CertificateId

Array pengidentifikasi untuk sertifikat yang diimpor. Anda menggunakan pengenal ini untuk bekerja dengan profil dan profil mitra.

Tipe: String

Kendala Panjang: Panjang tetap 22.

Pola: cert-([0-9a-f]{17})

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Contoh

Contoh

Contoh berikut mengimpor sertifikat yang akan digunakan untuk enkripsi. Pada perintah pertama, kami menyediakan konten file sertifikat dan rantai sertifikat. Gunakan format ini untuk SDK perintah.

```
aws transfer import-certificate --usage ENCRYPTION --certificate "`cat encryption-  
cert.pem`" \
```

```
--private-key "`cat encryption-key.pem`" --certificate-chain "`cat root-ca.pem`"
```

Contoh

Contoh berikut identik dengan perintah sebelumnya, kecuali bahwa kami menyediakan lokasi file untuk kunci pribadi, sertifikat, dan file rantai sertifikat. Versi perintah ini tidak berfungsi jika Anda menggunakan fileSDK.

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

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

ImportHostKey

Menambahkan kunci host ke server yang ditentukan oleh `ServerId` parameter.

Sintaksis Permintaan

```
{
  "Description": "string",
  "HostKeyBody": "string",
  "ServerId": "string",
  "Tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ]
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[Description](#)

Deskripsi teks yang mengidentifikasi kunci host ini.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 200.

Pola: `[\p{Print}]*`

Wajib: Tidak

[HostKeyBody](#)

Bagian kunci pribadi dari sebuah SSH key pair.

AWS Transfer Family menerima RSA, ECDSA, dan ED25519 kunci.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 4096.

Wajib: Ya

[ServerId](#)

Pengidentifikasi server yang berisi kunci host yang Anda impor.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s - ([0-9a-f]{17})

Wajib: Ya

[Tags](#)

Pasangan kunci-nilai yang dapat digunakan untuk mengelompokkan dan mencari kunci host.

Tipe: Array objek [Tag](#)

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 50 item.

Diperlukan: Tidak

Sintaksis Respons

```
{
  "HostKeyId": "string",
  "ServerId": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

[HostKeyId](#)

Mengembalikan pengenalan kunci host untuk kunci yang diimpor.

Tipe: String

Kendala Panjang: Panjang tetap 25.

Pola: hostkey-[0-9a-f]{17}

ServerId

Mengembalikan pengenalan server yang berisi kunci yang diimpor.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s-([0-9a-f]{17})

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceExistsException

Sumber daya yang diminta tidak ada, atau ada di wilayah selain yang ditentukan untuk perintah.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

ThrottlingException

Permintaan ditolak karena throttling permintaan.

HTTPKode Status: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

ImportSshPublicKey

Menambahkan kunci publik Secure Shell (SSH) ke pengguna Transfer Family yang diidentifikasi oleh `UserName` nilai yang ditetapkan ke server berkemampuan protokol transfer file tertentu, yang diidentifikasi oleh `ServerId`.

Respons mengembalikan `UserName` nilai, `ServerId` nilai, dan `namaSshPublicKeyId`.

Sintaksis Permintaan

```
{
  "ServerId": "string",
  "SshPublicKeyBody": "string",
  "UserName": "string"
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[ServerId](#)

Pengidentifikasi unik yang ditetapkan sistem untuk server.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s-([0-9a-f]{17})

Wajib: Ya

[SshPublicKeyBody](#)

Bagian kunci publik dari sebuah SSH key pair.

AWS Transfer Family menerima RSA, ECDSA, dan ED25519 kunci.

Tipe: String

Batasan Panjang: Panjang minimum sebesar 0. Panjang maksimum 2048.

Wajib: Ya

UserName

Nama pengguna Transfer Family yang ditetapkan ke satu atau beberapa server.

Tipe: String

Batasan Panjang: Panjang minimum 3. Panjang maksimum 100.

Pola: `[\w][\w@.-]{2,99}`

Diperlukan: Ya

Sintaksis Respons

```
{
  "ServerId": "string",
  "SshPublicKeyId": "string",
  "UserName": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

ServerId

Pengidentifikasi unik yang ditetapkan sistem untuk server.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: `s-([0-9a-f]{17})`

SshPublicKeyId

Nama yang diberikan untuk kunci publik oleh sistem yang diimpor.

Tipe: String

Kendala Panjang: Panjang tetap 21.

Pola: key-[0-9a-f]{17}

UserName

Nama pengguna yang ditetapkan ke ServerID nilai yang Anda tentukan.

Tipe: String

Batasan Panjang: Panjang minimum 3. Panjang maksimum 100.

Pola: [\w][\w@.-]{2,99}

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServiceError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceExistsException

Sumber daya yang diminta tidak ada, atau ada di wilayah selain yang ditentukan untuk perintah.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

ThrottlingException

Permintaan ditolak karena throttling permintaan.

HTTPKode Status: 400

Contoh

Contoh

Perintah ini mengimpor ECDSA kunci yang disimpan dalam `id_ecdsa.pub` file.

```
aws transfer import-ssh-public-key --server-id s-021345abcdef6789 --ssh-public-key-body
file://id_ecdsa.pub --user-name jane-doe
```

Contoh

Jika Anda menjalankan perintah sebelumnya, sistem mengembalikan informasi berikut.

```
{
  "ServerId": "s-021345abcdef6789",
  "SshPublicKeyId": "key-1234567890abcdef0",
  "UserName": "jane-doe"
}
```

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)

- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

ListAccesses

Daftar detail untuk semua akses yang Anda miliki di server Anda.

Sintaksis Permintaan

```
{  
  "MaxResults": number,  
  "NextToken": "string",  
  "ServerId": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[MaxResults](#)

Menentukan jumlah maksimum akses SIDs untuk kembali.

Jenis: Integer

Rentang yang Valid: Nilai minimum 1. Nilai maksimum 1000.

Wajib: Tidak

[NextToken](#)

Ketika Anda bisa mendapatkan hasil tambahan dari `ListAccesses` panggilan, `NextToken` parameter dikembalikan dalam output. Anda kemudian dapat meneruskan perintah berikutnya ke `NextToken` parameter untuk melanjutkan daftar akses tambahan.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 6144.

Wajib: Tidak

[ServerId](#)

Pengidentifikasi unik yang ditetapkan sistem untuk server yang memiliki pengguna yang ditugaskan untuk itu.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s-([0-9a-f]{17})

Diperlukan: Ya

Sintaksis Respons

```
{
  "Accesses": [
    {
      "ExternalId": "string",
      "HomeDirectory": "string",
      "HomeDirectoryType": "string",
      "Role": "string"
    }
  ],
  "NextToken": "string",
  "ServerId": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

[Accesses](#)

Mengembalikan akses dan properti mereka untuk `ServerId` nilai yang Anda tentukan.

Tipe: Array objek [ListedAccess](#)

[NextToken](#)

Ketika Anda bisa mendapatkan hasil tambahan dari `ListAccesses` panggilan, `NextToken` parameter dikembalikan dalam output. Anda kemudian dapat meneruskan perintah berikutnya ke `NextToken` parameter untuk melanjutkan daftar akses tambahan.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 6144.

ServerId

Pengidentifikasi unik yang ditetapkan sistem untuk server yang memiliki pengguna yang ditugaskan untuk itu.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s - ([0-9a-f]{17})

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidNextTokenException

NextTokenParameter yang dilewatkan tidak valid.

HTTPKode Status: 400

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

ListAgreements

Mengembalikan daftar perjanjian untuk server yang diidentifikasi oleh `ServerId` yang Anda berikan. Jika Anda ingin membatasi hasil ke angka tertentu, berikan nilai untuk `MaxResults` parameter tersebut. Jika Anda menjalankan perintah sebelumnya dan menerima nilai untuk `NextToken`, Anda dapat memberikan nilai tersebut untuk melanjutkan daftar perjanjian dari tempat Anda tinggalkan.

Sintaksis Permintaan

```
{  
  "MaxResults": number,  
  "NextToken": "string",  
  "ServerId": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[MaxResults](#)

Jumlah maksimum perjanjian untuk dikembalikan.

Jenis: Integer

Rentang yang Valid: Nilai minimum 1. Nilai maksimum 1000.

Wajib: Tidak

[NextToken](#)

Ketika Anda bisa mendapatkan hasil tambahan dari `ListAgreements` panggilan, `NextToken` parameter dikembalikan dalam output. Anda kemudian dapat meneruskan perintah berikutnya ke `NextToken` parameter untuk melanjutkan daftar perjanjian tambahan.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 6144.

Wajib: Tidak

ServerId

Pengidentifikasi server yang Anda inginkan daftar perjanjian.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s-([0-9a-f]{17})

Diperlukan: Ya

Sintaksis Respons

```
{
  "Agreements": [
    {
      "AgreementId": "string",
      "Arn": "string",
      "Description": "string",
      "LocalProfileId": "string",
      "PartnerProfileId": "string",
      "ServerId": "string",
      "Status": "string"
    }
  ],
  "NextToken": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

Agreements

Mengembalikan array, di mana setiap item berisi rincian perjanjian.

Tipe: Array objek ListedAgreement

NextToken

Mengembalikan token yang dapat Anda gunakan untuk menelepon `ListAgreements` lagi dan menerima hasil tambahan, jika ada.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 6144.

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

`InternalServerError`

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

`InvalidNextTokenException`

`NextTokenParameter` yang dilewatkan tidak valid.

HTTPKode Status: 400

`InvalidRequestException`

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

`ResourceNotFoundException`

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

`ServiceUnavailableException`

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

ListCertificates

Mengembalikan daftar sertifikat saat ini yang telah diimpor ke AWS Transfer Family. Jika Anda ingin membatasi hasil ke angka tertentu, berikan nilai untuk `MaxResults` parameter tersebut. Jika Anda menjalankan perintah sebelumnya dan menerima nilai untuk `NextToken` parameter, Anda dapat memberikan nilai tersebut untuk melanjutkan daftar sertifikat dari tempat Anda tinggalkan.

Sintaksis Permintaan

```
{  
  "MaxResults": number,  
  "NextToken": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[MaxResults](#)

Jumlah maksimum sertifikat untuk dikembalikan.

Jenis: Integer

Rentang yang Valid: Nilai minimum 1. Nilai maksimum 1000.

Wajib: Tidak

[NextToken](#)

Ketika Anda bisa mendapatkan hasil tambahan dari `ListCertificates` panggilan, `NextToken` parameter dikembalikan dalam output. Anda kemudian dapat meneruskan perintah berikutnya ke `NextToken` parameter untuk melanjutkan daftar sertifikat tambahan.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 6144.

Wajib: Tidak

Sintaksis Respons

```
{
  "Certificates": [
    {
      "ActiveDate": number,
      "Arn": "string",
      "CertificateId": "string",
      "Description": "string",
      "InactiveDate": number,
      "Status": "string",
      "Type": "string",
      "Usage": "string"
    }
  ],
  "NextToken": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

[Certificates](#)

Mengembalikan array sertifikat yang ditentukan dalam `ListCertificates` panggilan.

Tipe: Array objek [ListedCertificate](#)

[NextToken](#)

Mengembalikan token berikutnya, yang dapat Anda gunakan untuk daftar sertifikat berikutnya.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 6144.

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServiceError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidNextTokenException

NextTokenParameter yang dilewatkan tidak valid.

HTTPKode Status: 400

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)

- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

ListConnectors

Daftar konektor untuk Wilayah yang ditentukan.

Sintaksis Permintaan

```
{  
  "MaxResults": number,  
  "NextToken": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[MaxResults](#)

Jumlah maksimum konektor untuk kembali.

Jenis: Integer

Rentang yang Valid: Nilai minimum 1. Nilai maksimum 1000.

Wajib: Tidak

[NextToken](#)

Ketika Anda bisa mendapatkan hasil tambahan dari `ListConnectors` panggilan, `NextToken` parameter dikembalikan dalam output. Anda kemudian dapat meneruskan perintah berikutnya ke `NextToken` parameter untuk melanjutkan daftar konektor tambahan.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 6144.

Wajib: Tidak

Sintaksis Respons

```
{
```

```
"Connectors": [  
  {  
    "Arn": "string",  
    "ConnectorId": "string",  
    "Url": "string"  
  }  
],  
"NextToken": "string"  
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

Connectors

Mengembalikan array, di mana setiap item berisi rincian konektor.

Tipe: Array objek [ListedConnector](#)

NextToken

Mengembalikan token yang dapat Anda gunakan untuk menelepon ListConnectors lagi dan menerima hasil tambahan, jika ada.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 6144.

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidNextTokenException

NextTokenParameter yang dilewatkan tidak valid.

HTTPKode Status: 400

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

ListExecutions

Daftar semua eksekusi yang sedang berlangsung untuk alur kerja yang ditentukan.

Note

Jika ID alur kerja yang ditentukan tidak dapat ditemukan, `ListExecutions` mengembalikan `ResourceNotFound` pengecualian.

Sintaksis Permintaan

```
{  
  "MaxResults": number,  
  "NextToken": "string",  
  "WorkflowId": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[MaxResults](#)

Menentukan jumlah maksimum eksekusi untuk kembali.

Jenis: Integer

Rentang yang Valid: Nilai minimum 1. Nilai maksimum 1000.

Wajib: Tidak

[NextToken](#)

`ListExecutions` mengembalikan `NextToken` parameter dalam output. Anda kemudian dapat meneruskan `NextToken` parameter dalam perintah berikutnya untuk melanjutkan daftar eksekusi tambahan.

Ini berguna untuk pagination, misalnya. Jika Anda memiliki 100 eksekusi untuk alur kerja, Anda mungkin hanya ingin daftar pertama 10. Jika demikian, hubungi API dengan menentukan: `max-results`

```
aws transfer list-executions --max-results 10
```

Ini mengembalikan rincian untuk 10 eksekusi pertama, serta pointer (NextToken) ke eksekusi kesebelas. Anda sekarang dapat menelepon API lagi, memberikan NextToken nilai yang Anda terima:

```
aws transfer list-executions --max-results 10 --next-token
$somePointerReturnedFromPreviousListResult
```

Panggilan ini mengembalikan 10 eksekusi berikutnya, yang ke-11 hingga ke-20. Anda kemudian dapat mengulangi panggilan sampai rincian untuk semua 100 eksekusi telah dikembalikan.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 6144.

Wajib: Tidak

[WorkflowId](#)

Pengidentifikasi unik untuk alur kerja.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: w-([a-z0-9]{17})

Diperlukan: Ya

Sintaksis Respons

```
{
  "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"
}

```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

Executions

Mengembalikan rincian untuk setiap eksekusi, dalam `ListedExecution` array.

Tipe: Array objek [ListedExecution](#)

NextToken

`ListExecutions` mengembalikan `NextToken` parameter dalam output. Anda kemudian dapat meneruskan `NextToken` parameter dalam perintah berikutnya untuk melanjutkan daftar eksekusi tambahan.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 6144.

WorkflowId

Pengidentifikasi unik untuk alur kerja.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: w-([a-z0-9]{17})

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidNextTokenException

NextTokenParameter yang dilewatkan tidak valid.

HTTPKode Status: 400

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

ListHostKeys

Mengembalikan daftar kunci host untuk server yang ditentukan oleh `ServerId` parameter.

Sintaksis Permintaan

```
{  
  "MaxResults": number,  
  "NextToken": "string",  
  "ServerId": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[MaxResults](#)

Jumlah maksimum kunci host untuk kembali.

Jenis: Integer

Rentang yang Valid: Nilai minimum 1. Nilai maksimum 1000.

Wajib: Tidak

[NextToken](#)

Ketika ada hasil tambahan yang tidak dikembalikan, `NextToken` parameter dikembalikan. Anda dapat menggunakan nilai tersebut untuk panggilan berikutnya `ListHostKeys` untuk melanjutkan hasil daftar.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 6144.

Wajib: Tidak

[ServerId](#)

Pengidentifikasi server yang berisi kunci host yang ingin Anda lihat.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s-([0-9a-f]{17})

Diperlukan: Ya

Sintaksis Respons

```
{
  "HostKeys": [
    {
      "Arn": "string",
      "DateImported": number,
      "Description": "string",
      "Fingerprint": "string",
      "HostKeyId": "string",
      "Type": "string"
    }
  ],
  "NextToken": "string",
  "ServerId": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

[HostKeys](#)

Mengembalikan array, di mana setiap item berisi rincian kunci host.

Tipe: Array objek [ListedHostKey](#)

[NextToken](#)

Mengembalikan token yang dapat Anda gunakan untuk menelepon ListHostKeys lagi dan menerima hasil tambahan, jika ada.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 6144.

ServerId

Mengembalikan pengenalan server yang berisi kunci host yang terdaftar.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s-([0-9a-f]{17})

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidNextTokenException

NextTokenParameter yang dilewatkan tidak valid.

HTTPKode Status: 400

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

ListProfiles

Mengembalikan daftar profil untuk sistem Anda. Jika Anda ingin membatasi hasil ke angka tertentu, berikan nilai untuk `MaxResults` parameter tersebut. Jika Anda menjalankan perintah sebelumnya dan menerima nilai untuk `NextToken`, Anda dapat memberikan nilai itu untuk melanjutkan daftar profil dari tempat Anda tinggalkan.

Sintaksis Permintaan

```
{  
  "MaxResults": number,  
  "NextToken": "string",  
  "ProfileType": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[MaxResults](#)

Jumlah maksimum profil untuk kembali.

Jenis: Integer

Rentang yang Valid: Nilai minimum 1. Nilai maksimum 1000.

Wajib: Tidak

[NextToken](#)

Ketika ada hasil tambahan yang tidak dikembalikan, `NextToken` parameter dikembalikan. Anda dapat menggunakan nilai tersebut untuk panggilan berikutnya `ListProfiles` untuk melanjutkan hasil daftar.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 6144.

Wajib: Tidak

ProfileType

Menunjukkan apakah hanya akan mencantumkan profil LOCAL tipe atau hanya PARTNER mengetik profil. Jika tidak disediakan dalam permintaan, perintah mencantumkan semua jenis profil.

Tipe: String

Nilai yang Valid: LOCAL | PARTNER

Wajib: Tidak

Sintaksis Respons

```
{
  "NextToken": "string",
  "Profiles": [
    {
      "Arn": "string",
      "As2Id": "string",
      "ProfileId": "string",
      "ProfileType": "string"
    }
  ]
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

NextToken

Mengembalikan token yang dapat Anda gunakan untuk menelepon ListProfiles lagi dan menerima hasil tambahan, jika ada.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 6144.

Profiles

Mengembalikan array, di mana setiap item berisi rincian profil.

Tipe: Array objek [ListedProfile](#)

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidNextTokenException

NextTokenParameter yang dilewatkan tidak valid.

HTTPKode Status: 400

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

ListSecurityPolicies

Daftar kebijakan keamanan yang dilampirkan ke server berkemampuan protokol transfer file Anda.

Sintaksis Permintaan

```
{  
  "MaxResults": number,  
  "NextToken": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[MaxResults](#)

Menentukan jumlah kebijakan keamanan untuk kembali sebagai respon terhadap ListSecurityPolicies query.

Jenis: Integer

Rentang yang Valid: Nilai minimum 1. Nilai maksimum 1000.

Wajib: Tidak

[NextToken](#)

Ketika hasil tambahan diperoleh dari ListSecurityPolicies perintah, NextToken parameter dikembalikan dalam output. Anda kemudian dapat meneruskan NextToken parameter dalam perintah berikutnya untuk melanjutkan daftar kebijakan keamanan tambahan.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 6144.

Wajib: Tidak

Sintaksis Respons

```
{
```

```
"NextToken": "string",  
"SecurityPolicyNames": [ "string" ]  
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

[NextToken](#)

Ketika Anda bisa mendapatkan hasil tambahan dari `ListSecurityPolicies` operasi, `NextToken` parameter dikembalikan dalam output. Dalam perintah berikut, Anda dapat meneruskan `NextToken` parameter untuk melanjutkan daftar kebijakan keamanan.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 6144.

[SecurityPolicyNames](#)

Berbagai kebijakan keamanan yang terdaftar.

Tipe: Array string

Batasan Panjang: Panjang minimum sebesar 0. Panjang maksimum 100.

Pola: `TransferSecurityPolicy-.+`

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTP Kode Status: 500

InvalidNextTokenException

`NextTokenParameter` yang dilewatkan tidak valid.

HTTPKode Status: 400

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Contoh

Contoh

Contoh berikut mencantumkan nama untuk semua kebijakan keamanan yang tersedia.

Permintaan Sampel

```
aws transfer list-security-policies
```

Contoh Respons

```
{
  "SecurityPolicyNames": [
    "TransferSecurityPolicy-2022-03",
    "TransferSecurityPolicy-FIPS-2020-06",
    "TransferSecurityPolicy-2018-11",
    "TransferSecurityPolicy-2023-05",
    "TransferSecurityPolicy-FIPS-2023-05",
    "TransferSecurityPolicy-2020-06"
  ]
}
```

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)

- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

ListServers

Daftar server berkemampuan protokol transfer file yang terkait dengan akun Anda. AWS

Sintaksis Permintaan

```
{  
  "MaxResults": number,  
  "NextToken": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[MaxResults](#)

Menentukan jumlah server untuk kembali sebagai respon terhadap ListServers query.

Jenis: Integer

Rentang yang Valid: Nilai minimum 1. Nilai maksimum 1000.

Wajib: Tidak

[NextToken](#)

Ketika hasil tambahan diperoleh dari ListServers perintah, NextToken parameter dikembalikan dalam output. Anda kemudian dapat meneruskan NextToken parameter dalam perintah berikutnya untuk melanjutkan daftar server tambahan.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 6144.

Wajib: Tidak

Sintaksis Respons

```
{  
  "NextToken": "string",  
}
```

```
"Servers": [  
  {  
    "Arn": "string",  
    "Domain": "string",  
    "EndpointType": "string",  
    "IdentityProviderType": "string",  
    "LoggingRole": "string",  
    "ServerId": "string",  
    "State": "string",  
    "UserCount": number  
  }  
]
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

[NextToken](#)

Ketika Anda bisa mendapatkan hasil tambahan dari `ListServers` operasi, `NextToken` parameter dikembalikan dalam output. Dalam perintah berikut, Anda dapat meneruskan `NextToken` parameter untuk melanjutkan daftar server tambahan.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 6144.

[Servers](#)

Array server yang terdaftar.

Tipe: Array objek [ListedServer](#)

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidNextTokenException

NextTokenParameter yang dilewatkan tidak valid.

HTTPKode Status: 400

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Contoh

Contoh

Contoh berikut mencantumkan server yang ada di Anda Akun AWS.

Perhatikan bahwa NextToken nilai contoh tidak nyata: mereka dimaksudkan untuk menunjukkan cara menggunakan parameter.

Permintaan Sampel

```
{
  "MaxResults": 1,
  "NextToken": "token-from-previous-API-call"
}
```

Contoh Respons

```
{
  "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
  }
]
```

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

ListTagsForResource

Daftar semua tag yang terkait dengan Amazon Resource Name (ARN) yang Anda tentukan. Sumber daya dapat berupa pengguna, server, atau peran.

Sintaksis Permintaan

```
{  
  "Arn": "string",  
  "MaxResults": number,  
  "NextToken": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

Arn

Meminta tag yang terkait dengan Amazon Resource Name (ARN) tertentu. An ARN adalah pengenal untuk AWS sumber daya tertentu, seperti server, pengguna, atau peran.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 1600.

Pola: `arn:\S+`

Wajib: Ya

MaxResults

Menentukan jumlah tag untuk kembali sebagai respon terhadap `ListTagsForResource` permintaan.

Jenis: Integer

Rentang yang Valid: Nilai minimum 1. Nilai maksimum 1000.

Wajib: Tidak

NextToken

Ketika Anda meminta hasil tambahan dari `ListTagsForResource` operasi, `NextToken` parameter dikembalikan dalam input. Anda kemudian dapat meneruskan perintah berikutnya ke `NextToken` parameter untuk melanjutkan daftar tag tambahan.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 6144.

Wajib: Tidak

Sintaksis Respons

```
{
  "Arn": "string",
  "NextToken": "string",
  "Tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ]
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

Arn

Yang ARN Anda tentukan untuk daftar tag.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 1600.

Pola: `arn:\S+`

NextToken

Ketika Anda bisa mendapatkan hasil tambahan dari `ListTagsForResource` panggilan, `NextToken` parameter dikembalikan dalam output. Anda kemudian dapat meneruskan perintah berikutnya ke `NextToken` parameter untuk melanjutkan daftar tag tambahan.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 6144.

Tags

Pasangan kunci-nilai yang ditugaskan ke sumber daya, biasanya untuk tujuan pengelompokan dan pencarian item. Tag adalah metadata yang Anda tentukan.

Tipe: Array objek [Tag](#)

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 50 item.

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

`InternalServerError`

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

`InvalidNextTokenException`

`NextTokenParameter` yang dilewatkan tidak valid.

HTTPKode Status: 400

`InvalidRequestException`

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

`ServiceUnavailableException`

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Contoh

Contoh

Contoh berikut mencantumkan tag untuk sumber daya dengan yang ARN Anda tentukan.

Permintaan Sampel

```
{
  "Arn": "arn:aws:transfer:us-east-1:176354371281:server/s-01234567890abcdef"
}
```

Contoh

Contoh ini menggambarkan salah satu penggunaan. ListTagsForResource

Contoh Respons

```
{
  "Tags": [
    {
      "Key": "Name",
      "Value": "MyServer"
    }
  ]
}
```

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)

- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

ListUsers

Daftar pengguna untuk server berkemampuan protokol transfer file yang Anda tentukan dengan meneruskan parameter. `ServerId`

Sintaksis Permintaan

```
{  
  "MaxResults": number,  
  "NextToken": "string",  
  "ServerId": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[MaxResults](#)

Menentukan jumlah pengguna untuk kembali sebagai respon terhadap `ListUsers` permintaan.

Jenis: Integer

Rentang yang Valid: Nilai minimum 1. Nilai maksimum 1000.

Wajib: Tidak

[NextToken](#)

Jika ada hasil tambahan dari `ListUsers` panggilan, `NextToken` parameter dikembalikan dalam output. Anda kemudian dapat meneruskan `NextToken` ke `ListUsers` perintah berikutnya, untuk melanjutkan daftar pengguna tambahan.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 6144.

Wajib: Tidak

[ServerId](#)

Pengidentifikasi unik yang ditetapkan sistem untuk server yang memiliki pengguna yang ditugaskan untuk itu.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s-([0-9a-f]{17})

Diperlukan: Ya

Sintaksis Respons

```
{
  "NextToken": "string",
  "ServerId": "string",
  "Users": [
    {
      "Arn": "string",
      "HomeDirectory": "string",
      "HomeDirectoryType": "string",
      "Role": "string",
      "SshPublicKeyCount": number,
      "UserName": "string"
    }
  ]
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

[NextToken](#)

Ketika Anda bisa mendapatkan hasil tambahan dari `ListUsers` panggilan, `NextToken` parameter dikembalikan dalam output. Anda kemudian dapat meneruskan perintah berikutnya ke `NextToken` parameter untuk melanjutkan daftar pengguna tambahan.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 6144.

[ServerId](#)

Pengidentifikasi unik yang ditetapkan sistem untuk server tempat pengguna ditugaskan.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s - ([0-9a-f]{17})

Users

Mengembalikan pengguna Transfer Family dan propertinya untuk `ServerId` nilai yang Anda tentukan.

Tipe: Array objek [ListedUser](#)

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidNextTokenException

`NextTokenParameter` yang dilewatkan tidak valid.

HTTPKode Status: 400

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPCode Status: 500

Contoh

Contoh

ListUsersAPI Panggilan mengembalikan daftar pengguna yang terkait dengan server yang Anda tentukan.

Permintaan Sampel

```
{
  "MaxResults": 100,
  "NextToken": "eyJNYXJrZXIiOiBudWxsLCAiYm90b1X0cnVuU2F0ZV9hbW91bnQiOiAyfQ==",
  "ServerId": "s-01234567890abcdef"
}
```

Contoh

Ini adalah contoh respons untuk API panggilan ini.

Contoh Respons

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

```
}  
]  
}
```

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

ListWorkflows

Daftar semua alur kerja yang terkait dengan wilayah Anda saat ini. Akun AWS

Sintaksis Permintaan

```
{  
  "MaxResults": number,  
  "NextToken": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[MaxResults](#)

Menentukan jumlah maksimum alur kerja untuk kembali.

Jenis: Integer

Rentang yang Valid: Nilai minimum 1. Nilai maksimum 1000.

Wajib: Tidak

[NextToken](#)

ListWorkflows mengembalikan NextToken parameter dalam output. Anda kemudian dapat meneruskan NextToken parameter dalam perintah berikutnya untuk melanjutkan daftar alur kerja tambahan.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 6144.

Wajib: Tidak

Sintaksis Respons

```
{  
  "NextToken": "string",  
}
```

```
"Workflows": [  
  {  
    "Arn": "string",  
    "Description": "string",  
    "WorkflowId": "string"  
  }  
]
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

[NextToken](#)

ListWorkflows mengembalikan NextToken parameter dalam output. Anda kemudian dapat meneruskan NextToken parameter dalam perintah berikutnya untuk melanjutkan daftar alur kerja tambahan.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 6144.

[Workflows](#)

Mengembalikan Arn, WorkflowId, dan Description untuk setiap alur kerja.

Tipe: Array objek [ListedWorkflow](#)

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServiceError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTP Kode Status: 500

InvalidNextTokenException

NextTokenParameter yang dilewatkan tidak valid.

HTTPKode Status: 400

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

SendWorkflowStepState

Mengirim callback untuk langkah-langkah kustom asinkron.

TheExecutionId, WorkflowId, dan Token diteruskan ke sumber daya target selama eksekusi langkah kustom alur kerja. Anda harus menyertakan mereka yang memiliki panggilan balik mereka serta memberikan status.

Sintaksis Permintaan

```
{
  "ExecutionId": "string",
  "Status": "string",
  "Token": "string",
  "WorkflowId": "string"
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

ExecutionId

Pengidentifikasi unik untuk eksekusi alur kerja.

Tipe: String

Batas Panjang: Panjang tetap 36.

Pola: `[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}`

Wajib: Ya

Status

Menunjukkan apakah langkah yang ditentukan berhasil atau gagal.

Tipe: String

Nilai yang Valid: SUCCESS | FAILURE

Wajib: Ya

Token

Digunakan untuk membedakan antara beberapa callback untuk beberapa langkah Lambda dalam eksekusi yang sama.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum adalah 64.

Pola: \w+

Wajib: Ya

WorkflowId

Pengidentifikasi unik untuk alur kerja.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: w-([a-z0-9]{17})

Diperlukan: Ya

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200 dengan HTTP tubuh kosong.

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

AccessDeniedException

Anda tidak memiliki akses yang memadai untuk melakukan tindakan ini.

HTTPKode Status: 400

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

ThrottlingException

Permintaan ditolak karena throttling permintaan.

HTTPKode Status: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

StartFileTransfer

Memulai transfer file antara AWS penyimpanan lokal dan remote AS2 atau SFTP server.

- Untuk AS2 konektor, Anda menentukan `ConnectorId` dan satu atau lebih `SendFilePaths` untuk mengidentifikasi file yang ingin Anda transfer.
- Untuk SFTP konektor, transfer file dapat berupa outbound atau inbound. Dalam kedua kasus, Anda menentukan `ConnectorId`. Tergantung pada arah transfer, Anda juga menentukan item berikut:
 - Jika Anda mentransfer file dari SFTP server mitra ke penyimpanan Amazon Web Services, Anda menentukan satu atau beberapa `RetrieveFilePaths` untuk mengidentifikasi file yang ingin ditransfer, dan `LocalDirectoryPath` untuk menentukan folder tujuan.
 - Jika Anda mentransfer file ke SFTP server mitra dari AWS penyimpanan, Anda menentukan satu atau lebih `SendFilePaths` untuk mengidentifikasi file yang ingin Anda transfer, dan `RemoteDirectoryPath` untuk menentukan folder tujuan.

Sintaksis Permintaan

```
{  
  "ConnectorId": "string",  
  "LocalDirectoryPath": "string",  
  "RemoteDirectoryPath": "string",  
  "RetrieveFilePaths": [ "string" ],  
  "SendFilePaths": [ "string" ]  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

ConnectorId

Pengidentifikasi unik untuk konektor.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: `c-([0-9a-f]{17})`

Wajib: Ya

LocalDirectoryPath

Untuk transfer masuk, LocalDirectoryPath menentukan tujuan untuk satu atau lebih file yang ditransfer dari server mitraSFTP.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 1024.

Pola: (.)+

Wajib: Tidak

RemoteDirectoryPath

Untuk transfer keluar, RemoteDirectoryPath menentukan tujuan untuk satu atau lebih file yang ditransfer ke server mitraSFTP. Jika Anda tidak menentukanRemoteDirectoryPath, tujuan untuk file yang ditransfer adalah direktori home SFTP pengguna.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 1024.

Pola: (.)+

Wajib: Tidak

RetrieveFilePaths

Satu atau lebih jalur sumber untuk SFTP server mitra. Setiap string mewakili jalur file sumber untuk satu transfer file masuk.

Tipe: Array string

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 10 item.

Batasan Panjang: Panjang minimum 1. Panjang maksimum 1024.

Pola: (.)+

Wajib: Tidak

SendFilePaths

Satu atau lebih jalur sumber untuk penyimpanan Amazon S3. Setiap string mewakili jalur file sumber untuk satu transfer file keluar. Misalnya, `DOC-EXAMPLE-BUCKET/myfile.txt` .

Note

Ganti `DOC-EXAMPLE-BUCKET` dengan salah satu ember Anda yang sebenarnya.

Tipe: Array string

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 10 item.

Batasan Panjang: Panjang minimum 1. Panjang maksimum 1024.

Pola: `(. )+`

Diperlukan: Tidak

Sintaksis Respons

```
{
  "TransferId": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

TransferId

Mengembalikan identifier unik untuk transfer file.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 512.

Pola: `[0-9a-zA-Z./- ]+`

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

ThrottlingException

Permintaan ditolak karena throttling permintaan.

HTTPKode Status: 400

Contoh

Contoh

Contoh berikut memulai transfer AS2 file dari server Transfer Family ke titik akhir mitra dagang jarak jauh. Ganti `DOC-EXAMPLE-BUCKET` dengan salah satu ember Anda yang sebenarnya.

Permintaan Sampel

```
{
  "ConnectorId": "c-AAAA1111BBBB2222C",
```

```
"SendFilePaths": [  
    "/DOC-EXAMPLE-BUCKET/myfile-1.txt",  
    "/DOC-EXAMPLE-BUCKET/myfile-2.txt",  
    "/DOC-EXAMPLE-BUCKET/myfile-3.txt"  
]  
}
```

Contoh Respons

```
{  
  "TransferId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"  
}
```

Contoh

Contoh berikut memulai transfer file dari AWS penyimpanan lokal ke SFTP server jauh.

Permintaan Sampel

```
{  
  "ConnectorId": "c-01234567890abcdef",  
  "SendFilePaths": [  
    "/DOC-EXAMPLE-BUCKET/myfile-1.txt",  
    "/DOC-EXAMPLE-BUCKET/myfile-2.txt",  
    "/DOC-EXAMPLE-BUCKET/myfile-3.txt"  
  ],  
  "RemoteDirectoryPath": "/MySFTPRootFolder/fromTransferFamilyServer"  
}
```

Contoh Respons

```
{  
  "TransferId": "a1b2c3d4-5678-90ab-cdef-EXAMPLE22222"  
}
```

Contoh

Contoh berikut memulai transfer file dari SFTP server jauh ke AWS penyimpanan lokal.

Permintaan Sampel

```
{
```

```
"ConnectorId": "c-111122223333AAAAA",
"RetrieveFilePaths": [
  "/MySFTPFolder/toTransferFamily/myfile-1.txt",
  "/MySFTPFolder/toTransferFamily/myfile-2.txt",
  "/MySFTPFolder/toTransferFamily/myfile-3.txt"
],
"LocalDirectoryPath": "/DOC-EXAMPLE-BUCKET/mySourceFiles"
}
```

Contoh Respons

```
{
  "TransferId": "a1b2c3d4-5678-90ab-cdef-EXAMPLEaaaaa"
}
```

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

StartServer

Mengubah status server berkemampuan protokol transfer file dari ke. OFFLINE ONLINE Ini tidak berdampak pada server yang sudah ada ONLINE. ONLINE Server dapat menerima dan memproses pekerjaan transfer file.

Status STARTING menunjukkan bahwa server berada dalam keadaan perantara, baik tidak sepenuhnya dapat merespons, atau tidak sepenuhnya online. Nilai START_FAILED dapat menunjukkan kondisi kesalahan.

Tidak ada tanggapan yang dikembalikan dari panggilan ini.

Sintaksis Permintaan

```
{  
  "ServerId": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[ServerId](#)

Pengidentifikasi unik yang ditetapkan sistem untuk server yang Anda mulai.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s-([0-9a-f]{17})

Diperlukan: Ya

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200 dengan HTTP tubuh kosong.

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

ThrottlingException

Permintaan ditolak karena throttling permintaan.

HTTPKode Status: 400

Contoh

Contoh

Contoh berikut memulai server.

Permintaan Sampel

```
{
  "ServerId": "s-01234567890abcdef"
```

```
}
```

Contoh

Ini adalah contoh respons untuk API panggilan ini.

Contoh Respons

```
{  
  "ServerId": "s-01234567890abcdef"  
}
```

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

StopServer

Mengubah status server berkemampuan protokol transfer file dari ke. ONLINE OFFLINE OFFLINE Server tidak dapat menerima dan memproses pekerjaan transfer file. Informasi yang terkait dengan server Anda, seperti server dan properti pengguna, tidak terpengaruh dengan menghentikan server Anda.

Note

Menghentikan server tidak mengurangi atau memengaruhi penagihan titik akhir protokol transfer file Anda; Anda harus menghapus server untuk berhenti ditagih.

Status STOPPING menunjukkan bahwa server berada dalam keadaan perantara, baik tidak sepenuhnya dapat merespons, atau tidak sepenuhnya offline. Nilai STOP_FAILED dapat menunjukkan kondisi kesalahan.

Tidak ada tanggapan yang dikembalikan dari panggilan ini.

Sintaksis Permintaan

```
{  
  "ServerId": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

ServerId

Pengidentifikasi unik yang ditetapkan sistem untuk server yang Anda hentikan.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s-([0-9a-f]{17})

Diperlukan: Ya

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200 dengan HTTP tubuh kosong.

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

ThrottlingException

Permintaan ditolak karena throttling permintaan.

HTTPKode Status: 400

Contoh

Contoh

Contoh berikut menghentikan server.

Permintaan Sampel

```
{  
  "ServerId": "s-01234567890abcdef"  
}
```

Contoh

Ini adalah contoh respons untuk API panggilan ini.

Contoh Respons

```
{  
  "ServerId": "s-01234567890abcdef"  
}
```

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

TagResource

Melampirkan pasangan kunci-nilai ke sumber daya, seperti yang diidentifikasi oleh Amazon Resource Name (). ARN Sumber daya adalah pengguna, server, peran, dan entitas lainnya.

Tidak ada jawaban yang dikembalikan dari panggilan ini.

Sintaksis Permintaan

```
{
  "Arn": "string",
  "Tags": [
    {
      "Key": "string",
      "Value": "string"
    }
  ]
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[Arn](#)

Amazon Resource Name (ARN) untuk AWS sumber daya tertentu, seperti server, pengguna, atau peran.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 1600.

Pola: `arn:\S+`

Wajib: Ya

[Tags](#)

Pasangan nilai kunci yang ditetapkan untuk ARNs yang dapat Anda gunakan untuk mengelompokkan dan mencari sumber daya berdasarkan jenis. Anda dapat melampirkan

metadata ini ke sumber daya (server, pengguna, alur kerja, dan sebagainya) untuk tujuan apa pun.

Tipe: Array objek [Tag](#)

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 50 item.

Diperlukan: Ya

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200 dengan HTTP tubuh kosong.

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Contoh

Contoh

Contoh berikut menambahkan tag ke server berkemampuan protokol transfer file.

Permintaan Sampel

```
{
  "Arn": "arn:aws:transfer:us-east-1:176354371281:server/s-01234567890abcdef",
  "Tags": [
    {
      "Key": "Group",
      "Value": "Europe"
    }
  ]
}
```

Contoh

Contoh ini menggambarkan salah satu penggunaan. TagResource

Contoh Respons

HTTP 200 response with an empty HTTP body.

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)

- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

TestConnection

Menguji apakah SFTP konektor Anda berhasil diatur. Kami sangat menyarankan Anda memanggil operasi ini untuk menguji kemampuan Anda mentransfer file antara AWS penyimpanan lokal dan SFTP server mitra dagang.

Sintaksis Permintaan

```
{  
  "ConnectorId": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

ConnectorId

Pengidentifikasi unik untuk konektor.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: c-([0-9a-f]{17})

Diperlukan: Ya

Sintaksis Respons

```
{  
  "ConnectorId": "string",  
  "Status": "string",  
  "StatusMessage": "string"  
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

ConnectorId

Mengembalikan identifier dari objek konektor yang Anda uji.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: c-([0-9a-f]{17})

Status

Pengembalian OK untuk tes yang berhasil, atau ERROR jika tes gagal.

Tipe: String

StatusMessage

Kembali Connection succeeded jika tes berhasil. Atau, mengembalikan pesan kesalahan deskriptif jika tes gagal. Daftar berikut memberikan rincian pemecahan masalah, tergantung pada pesan kesalahan yang Anda terima.

- Verifikasi bahwa nama rahasia Anda sejajar dengan yang ada di izin Peran Transfer.
- Verifikasi server URL dalam konfigurasi konektor, dan verifikasi bahwa kredensial login berhasil bekerja di luar konektor.
- Verifikasi bahwa rahasia itu ada dan diformat dengan benar.
- Verifikasi bahwa kunci host tepercaya dalam konfigurasi konektor cocok dengan ssh-keyscan output.

Tipe: String

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServiceError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Contoh

Contoh

Contoh berikut menguji koneksi ke server jarak jauh.

```
aws transfer test-connection --connector-id c-abcd1234567890fff
```

Contoh Respons

Jika berhasil, API panggilan mengembalikan detail berikut.

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

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

TestIdentityProvider

Jika server `IdentityProviderType` berkemampuan protokol transfer file `AWS_DIRECTORY_SERVICE` atau `API_Gateway`, uji apakah penyedia identitas Anda berhasil disiapkan. Kami sangat menyarankan Anda memanggil operasi ini untuk menguji metode otentikasi Anda segera setelah Anda membuat server Anda. Dengan demikian, Anda dapat memecahkan masalah dengan integrasi penyedia identitas untuk memastikan bahwa pengguna Anda dapat berhasil menggunakan layanan.

Parameter `ServerId` dan `UserName` diperlukan. `ItsServerProtocol`, `SourceIp`, dan `UserPassword` semuanya opsional.

Perhatikan hal berikut:

- Anda tidak dapat menggunakan `TestIdentityProvider` jika server Anda `SERVICE_MANAGED`. `IdentityProviderType`
- `TestIdentityProvider` tidak berfungsi dengan kunci: hanya menerima kata sandi.
- `TestIdentityProvider` dapat menguji operasi kata sandi untuk Penyedia Identitas khusus yang menangani kunci dan kata sandi.
- Jika Anda memberikan nilai yang salah untuk parameter apa pun, Response bidang kosong.
- Jika Anda memberikan ID server untuk server yang menggunakan pengguna yang dikelola layanan, Anda mendapatkan kesalahan:

```
An error occurred (InvalidRequestException) when calling the
TestIdentityProvider operation: s-server-ID not configured for external
auth
```

- Jika Anda memasukkan ID Server untuk `--server-id` parameter yang tidak mengidentifikasi server Transfer yang sebenarnya, Anda menerima kesalahan berikut:

```
An error occurred (ResourceNotFoundException) when calling the
TestIdentityProvider operation: Unknown server.
```

Mungkin saja server Anda berada di wilayah yang berbeda. Anda dapat menentukan wilayah dengan menambahkan yang berikut: `--region region-code`, seperti `--region us-east-2` untuk menentukan server di AS Timur (Ohio).

Sintaksis Permintaan

```
{  
  "ServerId": "string",  
  "ServerProtocol": "string",  
  "SourceIp": "string",  
  "UserName": "string",  
  "UserPassword": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[ServerId](#)

Pengidentifikasi yang ditetapkan sistem untuk server tertentu. Metode otentikasi pengguna server itu diuji dengan nama pengguna dan kata sandi.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s-([0-9a-f]{17})

Wajib: Ya

[ServerProtocol](#)

Jenis protokol transfer file yang akan diuji.

Protokol yang tersedia adalah:

- Secure Shell (SSH) Protokol Transfer File (SFTP)
- Protokol Transfer File Aman (FTPS)
- Protokol Transfer File (FTP)
- Pernyataan Penerapan 2 () AS2

Tipe: String

Nilai yang Valid: SFTP | FTP | FTPS | AS2

Wajib: Tidak

[SourceIp](#)

Alamat IP sumber akun yang akan diuji.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum sebesar 32.

Pola: \d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3}

Wajib: Tidak

[UserName](#)

Nama akun yang akan diuji.

Tipe: String

Batasan Panjang: Panjang minimum 3. Panjang maksimum 100.

Pola: [\w][\w@.-]{2,99}

Wajib: Ya

[UserPassword](#)

Kata sandi akun yang akan diuji.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 1024.

Wajib: Tidak

Sintaksis Respons

```
{
  "Message": "string",
  "Response": "string",
  "StatusCode": number,
  "Url": "string"
```

```
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

Message

Pesan yang menunjukkan apakah tes berhasil atau tidak.

Note

Jika string kosong dikembalikan, penyebab yang paling mungkin adalah otentikasi gagal karena nama pengguna atau kata sandi yang salah.

Tipe: String

Response

Respons yang dikembalikan dari API Gateway Anda atau fungsi Lambda Anda.

Tipe: String

StatusCode

Kode HTTP status yang merupakan respons dari API Gateway Anda atau fungsi Lambda Anda.

Jenis: Integer

Url

Titik akhir layanan yang digunakan untuk mengautentikasi pengguna.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum sebesar 255.

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Contoh

Contoh

Permintaan berikut mengembalikan pesan dari penyedia identitas bahwa kombinasi nama pengguna dan kata sandi adalah identitas yang valid untuk digunakan AWS Transfer Family.

Permintaan Sampel

```
{
  "ServerID": "s-01234567890abcdef",
  "UserName": "my_user",
  "UserPassword": "MyPassword-1"
}
```

Contoh

Respons berikut menunjukkan respons sampel untuk tes yang berhasil.

Contoh Respons

```
"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\": \"\"
```

Contoh

Respons berikut menunjukkan bahwa pengguna yang ditentukan milik lebih dari satu grup yang memiliki akses.

```
"Response": "",
"StatusCode": 200,
"Message": "More than one associated access found for user's groups."
```

Contoh

Jika Anda telah membuat dan mengonfigurasi penyedia identitas kustom dengan menggunakan API Gateway, Anda dapat memasukkan perintah berikut untuk menguji pengguna Anda:

```
aws transfer test-identity-provider --server-id s-0123456789abcdefg --user-
name myuser
```

di mana s-0123456789abcdefg adalah server transfer Anda, dan myuser adalah nama pengguna untuk pengguna kustom Anda.

Jika perintah berhasil, respons Anda mirip dengan yang berikut ini, di mana:

- Akun AWS ID adalah 012345678901
- Peran pengguna adalah user-role-api-gateway
- Direktori home adalah myuser-bucket
- Kunci publik adalah kunci publik

- Doa URL adalah doa- 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\"
}
```

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

UntagResource

Melepaskan pasangan kunci-nilai dari sumber daya, seperti yang diidentifikasi oleh Amazon Resource Name (). ARN Sumber daya adalah pengguna, server, peran, dan entitas lainnya.

Tidak ada tanggapan yang dikembalikan dari panggilan ini.

Sintaksis Permintaan

```
{
  "Arn": "string",
  "TagKeys": [ "string" ]
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[Arn](#)

Nilai sumber daya yang akan menghapus tag. Amazon Resource Name (ARN) adalah pengidentifikasi untuk AWS sumber daya tertentu, seperti server, pengguna, atau peran.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 1600.

Pola: `arn:\S+`

Wajib: Ya

[TagKeys](#)

TagKeys adalah pasangan nilai kunci yang ditetapkan untuk ARNs yang dapat digunakan untuk mengelompokkan dan mencari sumber daya berdasarkan jenis. Metadata ini dapat dilampirkan ke sumber daya untuk tujuan apa pun.

Tipe: Array string

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 50 item.

Batasan Panjang: Panjang minimum sebesar 0. Panjang maksimum 128.

Wajib: Ya

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200 dengan HTTP tubuh kosong.

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

Contoh

Contoh

Contoh berikut menghapus tag server berkemampuan protokol transfer file.

Permintaan Sampel

```
{
  "Arn": "arn:aws:transfer:us-east-1:176354371281:server/s-01234567890abcdef",
  "TagKeys": "Europe" ]
}
```

Contoh

Contoh ini menggambarkan satu penggunaan. UntagResource

Contoh Respons

HTTP 200 response with an empty HTTP body.

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

UpdateAccess

Memungkinkan Anda memperbarui parameter untuk akses yang ditentukan dalam ExternalID parameter ServerID dan.

Sintaksis Permintaan

```
{
  "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"
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

ExternalId

Pengidentifikasi unik yang diperlukan untuk mengidentifikasi grup tertentu dalam direktori Anda. Pengguna grup yang Anda asosiasikan memiliki akses ke EFS sumber daya Amazon S3 atau Amazon Anda melalui protokol yang diaktifkan. AWS Transfer Family Jika Anda tahu nama grup, Anda dapat melihat SID nilai dengan menjalankan perintah berikut menggunakan Windows PowerShell.

```
Get-ADGroup -Filter {samAccountName -like "YourGroupName*"} -Properties  
* | Select SamAccountName,ObjectSid
```

Dalam perintah itu, ganti YourGroupName dengan nama grup Active Directory Anda.

Ekspresi reguler yang digunakan untuk memvalidasi parameter ini adalah string karakter yang terdiri dari huruf besar dan huruf kecil karakter alfanumerik tanpa spasi. Anda juga dapat menyertakan garis bawah atau salah satu karakter berikut: =, . @: /-

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 256.

Pola: S-1-[\d-]+

Wajib: Ya

[HomeDirectory](#)

Direktori arahan (folder) untuk pengguna ketika mereka masuk ke server menggunakan klien.

Contoh HomeDirectory adalah /bucket_name/home/mydirectory.

Note

Parameter HomeDirectory hanya digunakan jika HomeDirectoryType diatur ke PATH.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 1024.

Pola: (| / . *)

Wajib: Tidak

[HomeDirectoryMappings](#)

Pemetaan direktori logis yang menentukan jalur dan kunci Amazon S3 atau EFS Amazon apa yang harus terlihat oleh pengguna Anda dan bagaimana Anda ingin membuatnya terlihat. Anda harus menentukan Entry dan Target memasangkan, di mana Entry menunjukkan bagaimana jalur dibuat terlihat dan Target merupakan jalur Amazon S3 atau Amazon yang sebenarnya. EFS

Jika Anda hanya menentukan target, itu ditampilkan apa adanya. Anda juga harus memastikan bahwa peran AWS Identity and Access Management (IAM) Anda menyediakan akses ke jalur diTarget. Nilai ini dapat diatur hanya ketika HomeDirectoryType diatur ke LOGICAL.

Berikut ini adalah contoh Entry dan Target pair.

```
[ { "Entry": "/directory1", "Target": "/bucket_name/home/mydirectory" } ]
```

Dalam kebanyakan kasus, Anda dapat menggunakan nilai ini alih-alih kebijakan sesi untuk mengunci pengguna Anda ke direktori home yang ditunjuk (chroot). Untuk melakukan ini, Anda dapat mengatur Entry ke / dan mengatur Target ke nilai HomeDirectory parameter.

Berikut ini adalah contoh Entry dan Target pair untukchroot.

```
[ { "Entry": "/", "Target": "/bucket_name/home/mydirectory" } ]
```

Tipe: Array objek [HomeDirectoryMapEntry](#)

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 50000 item.

Wajib: Tidak

[HomeDirectoryType](#)

Jenis direktori pendaratan (folder) yang Anda inginkan direktori home pengguna Anda ketika mereka masuk ke server. Jika Anda mengaturnyaPATH, pengguna akan melihat bucket Amazon S3 absolut atau EFS jalur Amazon seperti pada klien protokol transfer file mereka. Jika Anda menyetelnyaLOGICAL, Anda harus menyediakan pemetaan HomeDirectoryMappings untuk bagaimana Anda ingin membuat jalur Amazon S3 atau EFS Amazon terlihat oleh pengguna Anda.

Note

Jika HomeDirectoryType yaLOGICAL, Anda harus memberikan pemetaan, menggunakan parameter. HomeDirectoryMappings Jika, di sisi lain, HomeDirectoryType adalahPATH, Anda memberikan jalur absolut menggunakan HomeDirectory parameter. Anda tidak dapat memiliki keduanya HomeDirectory dan HomeDirectoryMappings di template Anda.

Tipe: String

Nilai yang Valid: PATH | LOGICAL

Wajib: Tidak

Policy

Kebijakan sesi untuk pengguna Anda sehingga Anda dapat menggunakan peran AWS Identity and Access Management (IAM) yang sama di beberapa pengguna. Kebijakan ini mencakup akses pengguna ke sebagian bucket Amazon S3 mereka. Variabel yang dapat Anda gunakan dalam kebijakan ini meliputi `${Transfer:UserName}`, `${Transfer:HomeDirectory}`, dan `${Transfer:HomeBucket}`.

Note

Kebijakan ini hanya berlaku jika domainnya `ServerId` adalah Amazon S3. Amazon EFS tidak menggunakan kebijakan sesi.

Untuk kebijakan sesi, AWS Transfer Family menyimpan kebijakan sebagai JSON gumpalan, bukan Amazon Resource Name (ARN) kebijakan. Anda menyimpan kebijakan sebagai JSON gumpalan dan meneruskannya dalam `Policy` argumen.

Untuk contoh kebijakan sesi, lihat [Contoh kebijakan sesi](#).

Untuk informasi selengkapnya, lihat [AssumeRole](#) di API Referensi Layanan Token AWS Keamanan.

Tipe: String

Batasan Panjang: Panjang minimum sebesar 0. Panjang maksimum 2048.

Wajib: Tidak

PosixProfile

POSIXIdentitas lengkap, termasuk ID pengguna (`Uid`), ID grup (`Gid`), dan grup sekunder IDs (`SecondaryGids`), yang mengontrol akses pengguna ke sistem EFS file Amazon Anda. POSIXIzin yang ditetapkan pada file dan direktori di sistem file Anda menentukan tingkat akses yang didapat pengguna saat mentransfer file masuk dan keluar dari sistem file Amazon EFS Anda.

Tipe: Objek [PosixProfile](#)

Wajib: Tidak

Role

Nama Sumber Daya Amazon (ARN) dari peran AWS Identity and Access Management (IAM) yang mengontrol akses pengguna ke bucket Amazon S3 atau sistem file EFS Amazon Anda. Kebijakan yang dilampirkan pada peran ini menentukan tingkat akses yang ingin Anda berikan kepada pengguna saat mentransfer file masuk dan keluar dari bucket Amazon S3 atau sistem file EFS Amazon. IAM Peran tersebut juga harus berisi hubungan kepercayaan yang memungkinkan server mengakses sumber daya Anda saat melayani permintaan transfer pengguna Anda.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 2048.

Pola: `arn:.*role/\S+`

Wajib: Tidak

ServerId

Pengidentifikasi unik yang ditetapkan sistem untuk instans server. Ini adalah server tertentu tempat Anda menambahkan pengguna.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: `s-([0-9a-f]{17})`

Diperlukan: Ya

Sintaksis Respons

```
{
  "ExternalId": "string",
  "ServerId": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

ExternalId

Pengenal eksternal grup yang penggunanya memiliki akses ke sumber daya Amazon S3 atau EFS Amazon Anda melalui protokol yang diaktifkan menggunakan AWS Transfer Family.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 256.

Pola: S-1-[\d-]+

ServerId

Pengidentifikasi server tempat pengguna dilampirkan.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s-([0-9a-f]{17})

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServiceError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceExistsException

Sumber daya yang diminta tidak ada, atau ada di wilayah selain yang ditentukan untuk perintah.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

ThrottlingException

Permintaan ditolak karena throttling permintaan.

HTTPKode Status: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

UpdateAgreement

Memperbarui beberapa parameter untuk perjanjian yang ada. Berikan `AgreementId` dan `ServerId` untuk perjanjian yang ingin Anda perbarui, bersama dengan nilai baru untuk parameter yang akan diperbarui.

Sintaksis Permintaan

```
{
  "AccessRole": "string",
  "AgreementId": "string",
  "BaseDirectory": "string",
  "Description": "string",
  "LocalProfileId": "string",
  "PartnerProfileId": "string",
  "ServerId": "string",
  "Status": "string"
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[AccessRole](#)

Konektor digunakan untuk mengirim file menggunakan SFTP protokol AS2 atau. Untuk peran akses, berikan Amazon Resource Name (ARN) AWS Identity and Access Management peran yang akan digunakan.

Untuk AS2 konektor

Dengan AS2, Anda dapat mengirim file dengan memanggil `StartFileTransfer` dan menentukan jalur file dalam parameter permintaan, `SendFilePaths`. Kami menggunakan direktori induk file (misalnya, untuk `--send-file-paths /bucket/dir/file.txt`, direktori induk `/bucket/dir/`) untuk sementara menyimpan file AS2 pesan yang diproses, menyimpan MDN ketika kami menerimanya dari mitra, dan menulis JSON file akhir yang berisi metadata transmisi yang relevan. Jadi, `AccessRole` kebutuhan untuk menyediakan akses baca dan tulis ke direktori induk dari lokasi file yang digunakan dalam `StartFileTransfer` permintaan. Selain

itu, Anda perlu menyediakan akses baca dan tulis ke direktori induk dari file yang ingin Anda kirimStartFileTransfer.

Jika Anda menggunakan otentikasi Dasar untuk AS2 konektor Anda, peran akses memerlukan secretsmanager:GetSecretValue izin untuk rahasia tersebut. Jika rahasia dienkripsi menggunakan kunci yang dikelola pelanggan alih-alih kunci yang dikelola di AWS Secrets Manager, maka peran tersebut juga memerlukan kms:Decrypt izin untuk kunci tersebut.

Untuk SFTP konektor

Pastikan bahwa peran akses menyediakan akses baca dan tulis ke direktori induk dari lokasi file yang digunakan dalam StartFileTransfer permintaan. Selain itu, pastikan bahwa peran tersebut memberikan secretsmanager:GetSecretValue izin untuk AWS Secrets Manager.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 2048.

Pola: arn:.*role/\S+

Wajib: Tidak

AgreementId

Pengidentifikasi unik untuk perjanjian. Pengenal ini dikembalikan saat Anda membuat perjanjian.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: a-([0-9a-f]{17})

Wajib: Ya

BaseDirectory

Untuk mengubah direktori pendaratan (folder) untuk file yang ditransfer, berikan folder bucket yang ingin Anda gunakan; misalnya, */DOC-EXAMPLE-BUCKET/home/mydirectory* .

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 1024.

Pola: (|/.*)

Wajib: Tidak

Description

Untuk mengganti deskripsi yang ada, berikan deskripsi singkat untuk perjanjian tersebut.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 200.

Pola: $[\backslash p\{Graph\}]^+$

Wajib: Tidak

LocalProfileId

Pengenal unik untuk profil AS2 lokal.

Untuk mengubah pengenal profil lokal, berikan nilai baru di sini.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: $p - ([0-9a-f]\{17\})$

Wajib: Tidak

PartnerProfileId

Pengenal unik untuk profil mitra. Untuk mengubah pengenal profil mitra, berikan nilai baru di sini.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: $p - ([0-9a-f]\{17\})$

Wajib: Tidak

ServerId

Pengidentifikasi unik yang ditetapkan sistem untuk instans server. Ini adalah server khusus yang digunakan perjanjian.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s-([0-9a-f]{17})

Wajib: Ya

[Status](#)

Anda dapat memperbarui status perjanjian, baik mengaktifkan perjanjian yang tidak aktif atau sebaliknya.

Tipe: String

Nilai yang Valid: ACTIVE | INACTIVE

Wajib: Tidak

Sintaksis Respons

```
{  
  "AgreementId": "string"  
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

[AgreementId](#)

Pengidentifikasi unik untuk perjanjian. Pengenal ini dikembalikan saat Anda membuat perjanjian.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: a-([0-9a-f]{17})

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceExistsException

Sumber daya yang diminta tidak ada, atau ada di wilayah selain yang ditentukan untuk perintah.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

ThrottlingException

Permintaan ditolak karena throttling permintaan.

HTTPKode Status: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)

- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

UpdateCertificate

Memperbarui tanggal aktif dan tidak aktif untuk sertifikat.

Sintaksis Permintaan

```
{  
  "ActiveDate": number,  
  "CertificateId": "string",  
  "Description": "string",  
  "InactiveDate": number  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[ActiveDate](#)

Tanggal opsional yang menentukan kapan sertifikat menjadi aktif.

Tipe: Timestamp

Wajib: Tidak

[CertificateId](#)

Pengidentifikasi objek sertifikat yang Anda perbarui.

Tipe: String

Kendala Panjang: Panjang tetap 22.

Pola: cert-([0-9a-f]{17})

Wajib: Ya

[Description](#)

Deskripsi singkat untuk membantu mengidentifikasi sertifikat.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 200.

Pola: `[\p{Graph}]`+

Wajib: Tidak

InactiveDate

Tanggal opsional yang menentukan kapan sertifikat menjadi tidak aktif.

Tipe: Timestamp

Wajib: Tidak

Sintaksis Respons

```
{  
  "CertificateId": "string"  
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

CertificateId

Mengembalikan identifier dari objek sertifikat yang Anda perbarui.

Tipe: String

Kendala Panjang: Panjang tetap 22.

Pola: `cert-([0-9a-f]{17})`

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServiceError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

ThrottlingException

Permintaan ditolak karena throttling permintaan.

HTTPKode Status: 400

Contoh

Contoh

Contoh berikut memperbarui tanggal aktif sertifikat, menyetel tanggal aktif menjadi 16 Januari 2022 pukul 16:12:07 UTC -5 jam.

Permintaan Sampel

```
aws transfer update-certificate --certificate-id c-abcdefgh123456hijk --active-date
2022-01-16T16:12:07-05:00
```

Contoh

Berikut ini adalah contoh respons untuk API panggilan ini.

Contoh Respons

```
"CertificateId": "c-abcdefg123456hijk"
```

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

UpdateConnector

Memperbarui beberapa parameter untuk konektor yang ada. Berikan konektor `ConnectorId` yang ingin Anda perbarui, bersama dengan nilai baru untuk parameter yang akan diperbarui.

Sintaksis Permintaan

```
{
  "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",
  "SftpConfig": {
    "TrustedHostKeys": [ "string" ],
    "UserSecretId": "string"
  },
  "Url": "string"
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[AccessRole](#)

Konektor digunakan untuk mengirim file menggunakan SFTP protokol AS2 atau. Untuk peran akses, berikan Amazon Resource Name (ARN) AWS Identity and Access Management peran yang akan digunakan.

Untuk AS2 konektor

Dengan AS2, Anda dapat mengirim file dengan memanggil `StartFileTransfer` dan menentukan jalur file dalam parameter permintaan, `SendFilePaths`. Kami menggunakan direktori induk file (misalnya, untuk `--send-file-paths /bucket/dir/file.txt`, direktori induk `/bucket/dir/`) untuk sementara menyimpan file AS2 pesan yang diproses, menyimpan MDN ketika kami menerimanya dari mitra, dan menulis JSON file akhir yang berisi metadata transmisi yang relevan. Jadi, `AccessRole` kebutuhan untuk menyediakan akses baca dan tulis ke direktori induk dari lokasi file yang digunakan dalam `StartFileTransfer` permintaan. Selain itu, Anda perlu menyediakan akses baca dan tulis ke direktori induk dari file yang ingin Anda kirim `StartFileTransfer`.

Jika Anda menggunakan otentikasi Dasar untuk AS2 konektor Anda, peran akses memerlukan `secretsmanager:GetSecretValue` izin untuk rahasia tersebut. Jika rahasia dienkripsi menggunakan kunci yang dikelola pelanggan alih-alih kunci yang dikelola di AWS Secrets Manager, maka peran tersebut juga memerlukan `kms:Decrypt` izin untuk kunci tersebut.

Untuk SFTP konektor

Pastikan bahwa peran akses menyediakan akses baca dan tulis ke direktori induk dari lokasi file yang digunakan dalam `StartFileTransfer` permintaan. Selain itu, pastikan bahwa peran tersebut memberikan `secretsmanager:GetSecretValue` izin untuk AWS Secrets Manager.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 2048.

Pola: `arn:.*role/\S+`

Wajib: Tidak

[As2Config](#)

Struktur yang berisi parameter untuk objek AS2 konektor.

Tipe: Objek [As2ConnectorConfig](#)

Wajib: Tidak

[ConnectorId](#)

Pengidentifikasi unik untuk konektor.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: c-([0-9a-f]{17})

Wajib: Ya

[LoggingRole](#)

Amazon Resource Name (ARN) dari peran AWS Identity and Access Management (IAM) yang memungkinkan konektor mengaktifkan CloudWatch logging untuk peristiwa Amazon S3. Saat disetel, Anda dapat melihat aktivitas konektor di CloudWatch log Anda.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 2048.

Pola: arn:.*role/\S+

Wajib: Tidak

[SftpConfig](#)

Struktur yang berisi parameter untuk objek SFTP konektor.

Tipe: Objek [SftpConnectorConfig](#)

Wajib: Tidak

[Url](#)

URL dari pasangan AS2 atau SFTP titik akhir.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum sebesar 255.

Wajib: Tidak

Sintaksis Respons

```
{
  "ConnectorId": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

ConnectorId

Mengembalikan identifier dari objek konektor yang Anda memperbarui.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: c - ([0-9a-f]{17})

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceExistsException

Sumber daya yang diminta tidak ada, atau ada di wilayah selain yang ditentukan untuk perintah.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

ThrottlingException

Permintaan ditolak karena throttling permintaan.

HTTPKode Status: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

UpdateHostKey

Memperbarui deskripsi untuk kunci host yang ditentukan oleh HostKeyId parameter ServerId dan.

Sintaksis Permintaan

```
{
  "Description": "string",
  "HostKeyId": "string",
  "ServerId": "string"
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[Description](#)

Deskripsi yang diperbarui untuk kunci host.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 200.

Pola: [\p{Print}]*

Wajib: Ya

[HostKeyId](#)

Pengidentifikasi kunci host yang Anda perbarui.

Tipe: String

Kendala Panjang: Panjang tetap 25.

Pola: hostkey-[0-9a-f]{17}

Wajib: Ya

[ServerId](#)

Pengidentifikasi server yang berisi kunci host yang Anda perbarui.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s-([0-9a-f]){17}

Diperlukan: Ya

Sintaksis Respons

```
{
  "HostKeyId": "string",
  "ServerId": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

[HostKeyId](#)

Mengembalikan pengenalan kunci host untuk kunci host diperbarui.

Tipe: String

Kendala Panjang: Panjang tetap 25.

Pola: hostkey-[0-9a-f]{17}

[ServerId](#)

Mengembalikan pengenalan server untuk server yang berisi kunci host diperbarui.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s - ([0-9a-f]{17})

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

ThrottlingException

Permintaan ditolak karena throttling permintaan.

HTTPKode Status: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

UpdateProfile

Memperbarui beberapa parameter untuk profil yang ada. Berikan profil `ProfileId` yang ingin Anda perbarui, bersama dengan nilai baru untuk parameter yang akan diperbarui.

Sintaksis Permintaan

```
{  
  "CertificateIds": [ "string" ],  
  "ProfileId": "string"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

CertificateIds

Array pengidentifikasi untuk sertifikat yang diimpor. Anda menggunakan pengenalan ini untuk bekerja dengan profil dan profil mitra.

Tipe: Array string

Kendala Panjang: Panjang tetap 22.

Pola: cert-([0-9a-f]{17})

Wajib: Tidak

ProfileId

Pengidentifikasi objek profil yang Anda perbarui.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: p-([0-9a-f]{17})

Diperlukan: Ya

Sintaksis Respons

```
{  
  "ProfileId": "string"  
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

ProfileId

Mengembalikan pengenalan untuk profil yang sedang diperbarui.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: p-([0-9a-f]{17})

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPCode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPCode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

ThrottlingException

Permintaan ditolak karena throttling permintaan.

HTTPKode Status: 400

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

UpdateServer

Memperbarui properti server berkemampuan protokol transfer file setelah server tersebut dibuat.

UpdateServerPanggilan mengembalikan server ServerId yang Anda perbarui.

Sintaksis Permintaan

```
{
  "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"
      }
    ]
  }
}

```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[Certificate](#)

Amazon Resource Name (ARN) dari AWS sertifikat Certificate Manager (ACM). Diperlukan saat Protocols diatur ke FTPS.

Untuk meminta sertifikat publik baru, lihat [Meminta sertifikat publik](#) di Panduan Pengguna AWS Certificate Manager.

Untuk mengimpor sertifikat yang ada ACM, lihat [Mengimpor sertifikat ke ACM dalam](#) Panduan Pengguna AWS Certificate Manager.

Untuk meminta sertifikat pribadi untuk digunakan FTPS melalui alamat IP pribadi, lihat [Meminta sertifikat pribadi](#) di Panduan Pengguna AWS Certificate Manager.

Sertifikat dengan algoritme kriptografi dan ukuran kunci berikut didukung:

- 2048-bit RSA (_2048) RSA
- 4096-bit RSA (_4096) RSA
- Elliptic Prime Curve 256 bit (EC_prime256v1)
- Elliptic Prime Curve 384 bit (EC_secp384r1)

- Elliptic Prime Curve 521 bit (EC_secp521r1)

 Note

Sertifikat harus berupa SSL TLS sertifikat/X.509 versi 3 yang valid dengan FQDN atau alamat IP yang ditentukan dan informasi tentang penerbit.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 1600.

Wajib: Tidak

EndpointDetails

Pengaturan titik akhir virtual private cloud (VPC) yang dikonfigurasi untuk server Anda. Ketika Anda meng-host titik akhir Anda di dalam AndaVPC, Anda dapat membuat titik akhir Anda hanya dapat diakses oleh sumber daya di dalam AndaVPC, atau Anda dapat melampirkan alamat IP Elastic dan membuat titik akhir Anda dapat diakses oleh klien melalui internet. Grup VPC keamanan default Anda secara otomatis ditetapkan ke titik akhir Anda.

Tipe: Objek [EndpointDetails](#)

Wajib: Tidak

EndpointType

Jenis endpoint yang Anda ingin server Anda gunakan. Anda dapat memilih untuk membuat endpoint server Anda dapat diakses secara publik (PUBLIC) atau menghostingnya di dalam server Anda. VPC Dengan endpoint yang di-host di aVPC, Anda dapat membatasi akses ke server dan sumber daya hanya di dalam Anda VPC atau memilih untuk membuatnya menghadap internet dengan melampirkan alamat IP Elastis langsung ke sana.

 Note

Setelah 19 Mei 2021, Anda tidak akan dapat membuat server menggunakan EndpointType=VPC_ENDPOINT di AWS akun Anda jika akun Anda belum melakukannya sebelum 19 Mei 2021. Jika Anda telah membuat server dengan EndpointType=VPC_ENDPOINT di AWS akun Anda pada atau sebelum 19 Mei 2021, Anda tidak akan terpengaruh. Setelah tanggal ini, gunakan EndpointType =VPC. Untuk informasi selengkapnya, lihat [Menghentikan penggunaan _ VPC ENDPOINT](#).

Direkomendasikan agar Anda menggunakan VPC sebagai EndpointType. Dengan tipe endpoint ini, Anda memiliki opsi untuk secara langsung mengaitkan hingga tiga IPv4 alamat Elastic (termasuk BYO IP) dengan endpoint server Anda dan menggunakan grup VPC keamanan untuk membatasi lalu lintas dengan alamat IP publik klien. Hal ini tidak mungkin terjadi jika EndpointType diatur ke VPC_ENDPOINT.

Tipe: String

Nilai yang Valid: PUBLIC | VPC | VPC_ENDPOINT

Wajib: Tidak

HostKey

KunciRSA, ECDSA, atau ED25519 pribadi yang akan digunakan untuk server SFTP yang diaktifkan. Anda dapat menambahkan beberapa kunci host, jika Anda ingin memutar tombol, atau memiliki satu set kunci aktif yang menggunakan algoritma yang berbeda.

Gunakan perintah berikut untuk menghasilkan kunci RSA 2048 bit tanpa frasa sandi:

```
ssh-keygen -t rsa -b 2048 -N "" -m PEM -f my-new-server-key.
```

Gunakan nilai minimum 2048 untuk -b opsi. Anda dapat membuat kunci yang lebih kuat dengan menggunakan 3072 atau 4096.

Gunakan perintah berikut untuk menghasilkan kunci ECDSA 256 bit tanpa frasa sandi:

```
ssh-keygen -t ecdsa -b 256 -N "" -m PEM -f my-new-server-key.
```

Nilai yang valid untuk -b opsi ECDSA adalah 256, 384, dan 521.

Gunakan perintah berikut untuk menghasilkan ED25519 kunci tanpa frasa sandi:

```
ssh-keygen -t ed25519 -N "" -f my-new-server-key.
```

Untuk semua perintah ini, Anda dapat mengganti my-new-server-key dengan string pilihan Anda.

Important

Jika Anda tidak berencana untuk memigrasikan pengguna yang sudah ada dari server SFTP berkemampuan yang sudah ada ke server baru, jangan perbarui kunci host. Mengubah kunci host server secara tidak sengaja dapat mengganggu.

Untuk informasi selengkapnya, lihat [Memperbarui kunci host untuk server yang SFTP diaktifkan](#) di Panduan AWS Transfer Family Pengguna.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 4096.

Wajib: Tidak

[IdentityProviderDetails](#)

Array yang berisi semua informasi yang diperlukan untuk memanggil API metode otentikasi pelanggan.

Tipe: Objek [IdentityProviderDetails](#)

Wajib: Tidak

[LoggingRole](#)

Nama Sumber Daya Amazon (ARN) dari peran AWS Identity and Access Management (IAM) yang memungkinkan server mengaktifkan CloudWatch pencatatan Amazon untuk Amazon S3 atau Amazon. EFSevents Saat disetel, Anda dapat melihat aktivitas pengguna di CloudWatch log Anda.

Tipe: String

Batasan Panjang: Panjang minimum sebesar 0. Panjang maksimum 2048.

Pola: (|arn:.*role/\S+)

Wajib: Tidak

[PostAuthenticationLoginBanner](#)

Menentukan string untuk ditampilkan ketika pengguna terhubung ke server. String ini ditampilkan setelah pengguna mengautentikasi.

Note

SFTPProtokol tidak mendukung spanduk tampilan pasca-otentikasi.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 4096.

Pola: `[\x09-\x0D\x20-\x7E]*`

Wajib: Tidak

[PreAuthenticationLoginBanner](#)

Menentukan string untuk ditampilkan ketika pengguna terhubung ke server. String ini ditampilkan sebelum pengguna mengautentikasi. Misalnya, spanduk berikut menampilkan detail tentang penggunaan sistem:

```
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.
```

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 4096.

Pola: `[\x09-\x0D\x20-\x7E]*`

Wajib: Tidak

[ProtocolDetails](#)

Pengaturan protokol yang dikonfigurasi untuk server Anda.

- Untuk menunjukkan mode pasif (untuk FTP dan FTPS protokol), gunakan parameter. `PassiveIp` Masukkan satu alamat dotted-quad, seperti IPv4 alamat IP eksternal firewall, router, atau penyeimbang beban.
- Untuk mengabaikan kesalahan yang dihasilkan saat klien mencoba menggunakan SETSTAT perintah pada file yang Anda unggah ke bucket Amazon S3, gunakan `SetStatOption` parameter. Agar AWS Transfer Family server mengabaikan SETSTAT perintah dan mengunggah file tanpa perlu membuat perubahan apa pun pada SFTP klien Anda, tetapkan nilainya. `ENABLE_NO_OP` Jika Anda menyetel `SetStatOption` parameter ke `ENABLE_NO_OP`, Transfer Family akan menghasilkan entri CloudWatch log ke Amazon Logs, sehingga Anda dapat menentukan kapan klien melakukan SETSTAT panggilan.
- Untuk menentukan apakah AWS Transfer Family server Anda melanjutkan sesi terbaru yang dinegosiasikan melalui ID sesi unik, gunakan parameter. `TlsSessionResumptionMode`

- `As2Transports` menunjukkan metode transportasi untuk AS2 pesan. Saat ini, hanya HTTP didukung.

Tipe: Objek [ProtocolDetails](#)

Wajib: Tidak

[Protocols](#)

Menentukan protokol transfer file atau protokol di mana klien protokol transfer file Anda dapat terhubung ke titik akhir server Anda. Protokol yang tersedia adalah:

- SFTP(Secure Shell (SSH) Protokol Transfer File): Transfer file SSH
- FTPS(File Transfer Protocol Secure): Transfer file dengan TLS enkripsi
- FTP (Protokol Transfer File): Transfer file tidak terenkripsi
- AS2(Pernyataan Penerapan 2): digunakan untuk mengangkut data terstruktur business-to-business

Note

- Jika Anda memilih FTPS, Anda harus memilih sertifikat yang disimpan di AWS Certificate Manager (ACM) yang digunakan untuk mengidentifikasi server Anda ketika klien terhubung dengannya FTPS.
- Jika `Protocol` termasuk salah satu FTP atau FTPS, maka `EndpointType` harus VPC dan `IdentityProviderType` harus baik `AWS_DIRECTORY_SERVICE`, `AWS_LAMBDA`, atau `API_GATEWAY`.
- Jika `Protocol` termasuk FTP, maka `AddressAllocationIds` tidak dapat dikaitkan.
- Jika `Protocol` disetel hanya ke SFTP, `EndpointType` dapat diatur ke PUBLIC dan `IdentityProviderType` dapat disetel salah satu jenis identitas yang didukung: `SERVICE_MANAGED`, `AWS_DIRECTORY_SERVICE`, `AWS_LAMBDA`, atau `API_GATEWAY`.
- Jika `Protocol` termasuk AS2, maka `EndpointType` harus VPC, dan domain harus Amazon S3.

Tipe: Array string

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 4 item.

Nilai yang Valid: SFTP | FTP | FTPS | AS2

Wajib: Tidak

[S3StorageOptions](#)

Menentukan apakah atau tidak kinerja untuk direktori Amazon S3 Anda dioptimalkan. Ini dinonaktifkan secara default.

Secara default, pemetaan direktori home memiliki TYPE file. DIRECTORY Jika Anda mengaktifkan opsi ini, Anda kemudian perlu secara eksplisit menyetel HomeDirectoryMapEntry Type ke FILE jika Anda ingin pemetaan memiliki target file.

Tipe: Objek [S3StorageOptions](#)

Wajib: Tidak

[SecurityPolicyName](#)

Menentukan nama kebijakan keamanan yang dilampirkan ke server.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 100.

Pola: TransferSecurityPolicy-.*

Wajib: Tidak

[ServerId](#)

Pengidentifikasi unik yang ditetapkan sistem untuk instance server yang ditetapkan oleh pengguna Transfer Family.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s-([0-9a-f]{17})

Wajib: Ya

[StructuredLogDestinations](#)

Menentukan grup log yang log server Anda dikirim.

Untuk menentukan grup log, Anda harus menyediakan ARN untuk grup log yang ada. Dalam hal ini, format grup log adalah sebagai berikut:

```
arn:aws:logs:region-name:amazon-account-id:log-group:log-group-name:*
```

Sebagai contoh, `arn:aws:logs:us-east-1:111122223333:log-group:mytestgroup:*`.

Jika sebelumnya Anda telah menentukan grup log untuk server, Anda dapat menghapusnya, dan pada dasarnya mematikan logging terstruktur, dengan memberikan nilai kosong untuk parameter ini dalam `update-server` panggilan. Sebagai contoh:

```
update-server --server-id s-1234567890abcdef0 --structured-log-destinations
```

Tipe: Array string.

Anggota Array: Jumlah minimum 0 item. Jumlah maksimum 1 item.

Batasan Panjang: Panjang minimum 20. Panjang maksimum 1600.

Pola: `arn:\S+`

Wajib: Tidak

[WorkflowDetails](#)

Menentukan ID alur kerja untuk alur kerja yang akan ditetapkan dan peran eksekusi yang digunakan untuk mengeksekusi alur kerja.

Selain alur kerja untuk mengeksekusi ketika file diunggah sepenuhnya, juga `WorkflowDetails` dapat berisi ID alur kerja (dan peran eksekusi) untuk alur kerja untuk mengeksekusi pada upload sebagian. Upload sebagian terjadi ketika sesi server terputus saat file masih diunggah.

Untuk menghapus alur kerja terkait dari server, Anda dapat memberikan `OnUpload` objek kosong, seperti pada contoh berikut.

```
aws transfer update-server --server-id s-01234567890abcdef --workflow-details '{"OnUpload":[]}'
```

Tipe: Objek [WorkflowDetails](#)

Wajib: Tidak

Sintaksis Respons

```
{  
  "ServerId": "string"  
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

[ServerId](#)

Pengidentifikasi unik yang ditetapkan sistem untuk server yang ditetapkan oleh pengguna Transfer Family.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s-([0-9a-f]{17})

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

AccessDeniedException

Anda tidak memiliki akses yang memadai untuk melakukan tindakan ini.

HTTPKode Status: 400

ConflictException

Pengecualian ini dilemparkan ketika UpdateServer dipanggil untuk server berkemampuan protokol transfer file yang memiliki VPC tipe titik akhir dan server tidak dalam VpcEndpointID keadaan tersedia.

HTTPKode Status: 400

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceExistsException

Sumber daya yang diminta tidak ada, atau ada di wilayah selain yang ditentukan untuk perintah.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

ThrottlingException

Permintaan ditolak karena throttling permintaan.

HTTPKode Status: 400

Contoh

Contoh

Contoh berikut memperbarui peran server.

Permintaan Sampel

```
{
  "EndpointDetails": {
    "VpcEndpointId": "vpce-01234f056f3g13",
```



```
"LoggingRole": "CloudWatchS3Events",  
"ServerId": "s-01234567890abcdef"  
}  
}
```

Contoh

Contoh berikut menghapus alur kerja terkait dari server.

Permintaan Sampel

```
aws transfer update-server --server-id s-01234567890abcdef --workflow-details  
'{"OnUpload":[]}'
```

Contoh

Ini adalah contoh respons untuk API panggilan ini.

Contoh Respons

```
{  
  "ServerId": "s-01234567890abcdef"  
}
```

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

UpdateUser

Menetapkan properti baru untuk pengguna. Parameter yang Anda lewati mengubah salah satu atau semua hal berikut: direktori home, peran, dan kebijakan untuk UserName dan yang ServerId Anda tentukan.

Respons mengembalikan ServerId dan UserName untuk pengguna yang diperbarui.

Di konsol, Anda dapat memilih Dibatasi saat membuat atau memperbarui pengguna. Ini memastikan bahwa pengguna tidak dapat mengakses apa pun di luar direktori home mereka. Cara terprogram untuk mengonfigurasi perilaku ini adalah dengan memperbarui pengguna. Setel HomeDirectoryType ke LOGICAL, dan tentukan HomeDirectoryMappings dengan Entry as root (/) dan Target sebagai direktori home mereka.

Misalnya, jika direktori home pengguna/test/admin-user, perintah berikut akan memperbarui pengguna sehingga konfigurasi mereka di konsol menunjukkan flag Restricted seperti yang dipilih.

```
aws transfer update-user --server-id <server-id> --user-name admin-user --home-directory-type LOGICAL --home-directory-mappings "[{\"Entry\":\"/\", \"Target\":\"/test/admin-user\"}]"
```

Sintaksis Permintaan

```
{
  "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"  
}
```

Parameter Permintaan

Untuk informasi tentang parameter yang umum untuk semua tindakan, lihat [Parameter Umum](#).

Permintaan menerima data berikut dalam JSON format.

[HomeDirectory](#)

Direktori arahan (folder) untuk pengguna ketika mereka masuk ke server menggunakan klien.

Contoh HomeDirectory adalah /bucket_name/home/mydirectory.

Note

Parameter HomeDirectory hanya digunakan jika HomeDirectoryType diatur ke PATH.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 1024.

Pola: (| / . *)

Wajib: Tidak

[HomeDirectoryMappings](#)

Pemetaan direktori logis yang menentukan jalur dan kunci Amazon S3 atau EFS Amazon apa yang harus terlihat oleh pengguna Anda dan bagaimana Anda ingin membuatnya terlihat. Anda harus menentukan Entry dan Target memasangkan, di mana Entry menunjukkan bagaimana jalur dibuat terlihat dan Target merupakan jalur Amazon S3 atau Amazon yang sebenarnya. EFS Jika Anda hanya menentukan target, itu ditampilkan apa adanya. Anda juga harus memastikan bahwa peran AWS Identity and Access Management (IAM) Anda menyediakan akses ke jalur diTarget. Nilai ini dapat diatur hanya ketika HomeDirectoryType diatur ke LOGICAL.

Berikut ini adalah contoh Entry dan Target pair.

```
[ { "Entry": "/directory1", "Target": "/bucket_name/home/  
mydirectory" } ]
```

Dalam kebanyakan kasus, Anda dapat menggunakan nilai ini alih-alih kebijakan sesi untuk mengunci pengguna Anda ke direktori home yang ditunjuk (`chroot`). Untuk melakukan ini, Anda dapat mengatur `Entry` ke `'/'` dan mengatur `Target` ke nilai `HomeDirectory` parameter.

Berikut ini adalah contoh `Entry` dan `Target` pair untuk `chroot`.

```
[ { "Entry": "/", "Target": "/bucket_name/home/mydirectory" } ]
```

Tipe: Array objek [HomeDirectoryMapEntry](#)

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 50000 item.

Wajib: Tidak

[HomeDirectoryType](#)

Jenis direktori pendaratan (folder) yang Anda inginkan direktori home pengguna Anda ketika mereka masuk ke server. Jika Anda mengaturnya `PATH`, pengguna akan melihat bucket Amazon S3 absolut atau EFS jalur Amazon seperti pada klien protokol transfer file mereka. Jika Anda menyetelnya `LOGICAL`, Anda harus menyediakan pemetaan `HomeDirectoryMappings` untuk bagaimana Anda ingin membuat jalur Amazon S3 atau EFS Amazon terlihat oleh pengguna Anda.

Note

Jika `HomeDirectoryType` ya `LOGICAL`, Anda harus memberikan pemetaan, menggunakan parameter `HomeDirectoryMappings`. Jika, di sisi lain, `HomeDirectoryType` adalah `PATH`, Anda memberikan jalur absolut menggunakan `HomeDirectory` parameter. Anda tidak dapat memiliki keduanya `HomeDirectory` dan `HomeDirectoryMappings` di template Anda.

Tipe: String

Nilai yang Valid: `PATH` | `LOGICAL`

Wajib: Tidak

[Policy](#)

Kebijakan sesi untuk pengguna Anda sehingga Anda dapat menggunakan peran AWS Identity and Access Management (IAM) yang sama di beberapa pengguna. Kebijakan ini mencakup akses pengguna ke sebagian bucket Amazon S3 mereka. Variabel yang dapat Anda gunakan

dalam kebijakan ini meliputi `${Transfer:UserName}`, `${Transfer:HomeDirectory}`, dan `${Transfer:HomeBucket}`.

 Note

Kebijakan ini hanya berlaku jika domainnya `ServerId` adalah Amazon S3. Amazon EFS tidak menggunakan kebijakan sesi.

Untuk kebijakan sesi, AWS Transfer Family menyimpan kebijakan sebagai JSON gumpalan, bukan Amazon Resource Name (ARN) kebijakan. Anda menyimpan kebijakan sebagai JSON gumpalan dan meneruskannya dalam `Policy` argumen.

Untuk contoh kebijakan sesi, lihat [Contoh kebijakan sesi](#).

Untuk informasi selengkapnya, lihat [AssumeRole](#) di API Referensi Layanan Token AWS Keamanan.

Tipe: String

Batasan Panjang: Panjang minimum sebesar 0. Panjang maksimum 2048.

Wajib: Tidak

[PosixProfile](#)

Menentukan POSIX identitas lengkap, termasuk ID pengguna (`Uid`), ID grup (`Gid`), dan grup sekunder IDs (`SecondaryGids`), yang mengontrol akses pengguna ke Amazon Elastic File Systems (AmazonEFS). POSIXIzin yang ditetapkan pada file dan direktori di sistem file Anda menentukan tingkat akses yang didapat pengguna saat mentransfer file masuk dan keluar dari sistem file Amazon EFS Anda.

Tipe: Objek [PosixProfile](#)

Wajib: Tidak

[Role](#)

Nama Sumber Daya Amazon (ARN) dari peran AWS Identity and Access Management (IAM) yang mengontrol akses pengguna ke bucket Amazon S3 atau sistem file EFS Amazon Anda. Kebijakan yang dilampirkan pada peran ini menentukan tingkat akses yang ingin Anda berikan kepada pengguna saat mentransfer file masuk dan keluar dari bucket Amazon S3 atau sistem file EFS Amazon. IAMPeran tersebut juga harus berisi hubungan kepercayaan yang memungkinkan server mengakses sumber daya Anda saat melayani permintaan transfer pengguna Anda.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 2048.

Pola: `arn:.*role/\S+`

Wajib: Tidak

ServerId

Pengidentifikasi unik yang ditetapkan sistem untuk instance server Transfer Family yang ditetapkan oleh pengguna.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: `s-([0-9a-f]{17})`

Wajib: Ya

UserName

String unik yang mengidentifikasi pengguna dan dikaitkan dengan server seperti yang ditentukan oleh `ServerId`. Nama pengguna ini harus sepanjang minimal 3 dan maksimal 100 karakter. Berikut adalah karakter yang valid: a-z, A-Z, 0-9, garis bawah '_', tanda hubung '-', titik '.', dan tanda at '@'. Nama pengguna tidak dapat dimulai dengan tanda hubung, titik, atau tanda at.

Tipe: String

Batasan Panjang: Panjang minimum 3. Panjang maksimum 100.

Pola: `[\w][\w@.-]{2,99}`

Diperlukan: Ya

Sintaksis Respons

```
{
  "ServerId": "string",
  "UserName": "string"
}
```

Elemen Respons

Jika tindakan berhasil, layanan mengirimkan kembali respons HTTP 200.

Data berikut dikembalikan dalam JSON format oleh layanan.

ServerId

Pengidentifikasi unik yang ditetapkan sistem untuk instance server Transfer Family tempat akun ditetapkan.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s - ([0-9a-f]{17})

UserName

Pengidentifikasi unik untuk pengguna yang ditugaskan ke instance server yang ditentukan dalam permintaan.

Tipe: String

Batasan Panjang: Panjang minimum 3. Panjang maksimum 100.

Pola: [\w][\w@.-]{2,99}

Kesalahan

Untuk informasi tentang kesalahan yang umum untuk semua tindakan, lihat [Kesalahan Umum](#).

InternalServerError

Pengecualian ini dilemparkan ketika kesalahan terjadi dalam AWS Transfer Family layanan.

HTTPKode Status: 500

InvalidRequestException

Pengecualian ini dilontarkan ketika klien mengirimkan permintaan yang salah format.

HTTPKode Status: 400

ResourceNotFoundException

Pengecualian ini dilemparkan ketika sumber daya tidak ditemukan oleh layanan AWS Transfer Family.

HTTPKode Status: 400

ServiceUnavailableException

Permintaan gagal karena layanan AWS Transfer Family tidak tersedia.

HTTPKode Status: 500

ThrottlingException

Permintaan ditolak karena throttling permintaan.

HTTPKode Status: 400

Contoh

Contoh

Contoh berikut memperbarui pengguna Transfer Family.

Permintaan Sampel

```
{
  "HomeDirectory": "/bucket2/documentation",
  "HomeDirectoryMappings": [
    {
      "Entry": "/directory1",
      "Target": "/bucket_name/home/mydirectory"
    }
  ],
  "HomeDirectoryType": "PATH",
  "Role": "AssumeRole",
  "ServerId": "s-01234567890abcdef",
  "UserName": "my_user"
}
```

Contoh

Ini adalah contoh respons untuk API panggilan ini.

Contoh Respons

```
{
  "ServerId": "s-01234567890abcdef",
  "UserName": "my_user"
}
```

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS Antarmuka Baris Perintah](#)
- [AWS SDK untuk .NET](#)
- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk JavaScript V3](#)
- [AWS SDK untuk PHP V3](#)
- [AWS SDK untuk Python](#)
- [AWS SDK untuk Ruby V3](#)

Tipe Data

tipe data berikut didukung:

- [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

Berisi rincian untuk objek AS2 konektor. Objek konektor digunakan untuk proses AS2 keluar, untuk menghubungkan AWS Transfer Family pelanggan dengan mitra dagang.

Daftar Isi

BasicAuthSecretId

Memberikan dukungan otentikasi Dasar ke AS2 KonektorAPI. Untuk menggunakan otentikasi Dasar, Anda harus memberikan nama atau Amazon Resource Name (ARN) rahasia di AWS Secrets Manager.

Nilai default untuk parameter ini adalah `null`, yang menunjukkan bahwa otentikasi Dasar tidak diaktifkan untuk konektor.

Jika konektor harus menggunakan otentikasi Dasar, rahasianya harus dalam format berikut:

```
{ "Username": "user-name", "Password": "user-password" }
```

Ganti `user-name` dan `user-password` dengan kredensi untuk pengguna sebenarnya yang sedang diautentikasi.

Perhatikan hal berikut:

- Anda menyimpan kredensi ini di Secrets Manager, tidak meneruskannya langsung ke ini. API
- Jika Anda menggunakan API, SDKs, atau CloudFormation untuk mengkonfigurasi konektor Anda, maka Anda harus membuat rahasia sebelum Anda dapat mengaktifkan otentikasi Dasar. Namun, jika Anda menggunakan konsol AWS manajemen, Anda dapat meminta sistem membuat rahasia untuk Anda.

Jika sebelumnya Anda telah mengaktifkan otentikasi dasar untuk konektor, Anda dapat menonaktifkannya dengan menggunakan `UpdateConnector` API panggilan. Misalnya, jika Anda menggunakan CLI, Anda dapat menjalankan perintah berikut untuk menghapus otentikasi Dasar:

```
update-connector --connector-id my-connector-id --as2-config  
'BasicAuthSecretId=""'
```

Tipe: String

Batasan Panjang: Panjang minimum sebesar 0. Panjang maksimum 2048.

Wajib: Tidak

Compression

Menentukan apakah AS2 file dikompresi.

Tipe: String

Nilai yang Valid: ZLIB | DISABLED

Wajib: Tidak

EncryptionAlgorithm

Algoritma yang digunakan untuk mengenkripsi file.

Perhatikan hal berikut:

- Jangan gunakan DES_EDE3_CBC algoritma kecuali Anda harus mendukung klien lama yang membutuhkannya, karena ini adalah algoritma enkripsi yang lemah.
- Anda hanya dapat menentukan NONE apakah URL untuk konektor Anda menggunakan HTTPS. Menggunakan HTTPS memastikan bahwa tidak ada lalu lintas yang dikirim dalam teks yang jelas.

Tipe: String

Nilai yang Valid: AES128_CBC | AES192_CBC | AES256_CBC | DES_EDE3_CBC | NONE

Wajib: Tidak

LocalProfileId

Pengenal unik untuk profil AS2 lokal.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: p-([0-9a-f]{17})

Wajib: Tidak

MdnResponse

Digunakan untuk permintaan keluar (dari AWS Transfer Family server ke AS2 server mitra) untuk menentukan apakah respons mitra untuk transfer sinkron atau asinkron. Tentukan salah satu dari nilai berikut:

- SYNC: Sistem mengharapkan MDN respons sinkron, mengkonfirmasi bahwa file berhasil ditransfer (atau tidak).
- NONE: Menentukan bahwa tidak ada MDN respon yang diperlukan.

Tipe: String

Nilai yang Valid: SYNC | NONE

Wajib: Tidak

MdnSigningAlgorithm

Algoritma penandatanganan untuk MDN respons.

Note

Jika disetel ke DEFAULT (atau tidak disetel sama sekali), nilai untuk SigningAlgorithm digunakan.

Tipe: String

Nilai yang Valid: SHA256 | SHA384 | SHA512 | SHA1 | NONE | DEFAULT

Wajib: Tidak

MessageSubject

Digunakan sebagai atribut Subject HTTP header dalam AS2 pesan yang sedang dikirim dengan konektor.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 1024.

Pola: `[\p{Print}\p{Blank}]+`

Wajib: Tidak

PartnerProfileId

Pengenal unik untuk profil mitra untuk konektor.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: `p-([0-9a-f]{17})`

Wajib: Tidak

SigningAlgorithm

Algoritma yang digunakan untuk menandatangani AS2 pesan yang dikirim dengan konektor.

Tipe: String

Nilai yang Valid: SHA256 | SHA384 | SHA512 | SHA1 | NONE

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

CopyStepDetails

Setiap tipe langkah memiliki StepDetails strukturnya sendiri.

Daftar Isi

DestinationFileLocation

Menentukan lokasi untuk file yang disalin. Gunakan `${Transfer:UserName}` atau `${Transfer:UploadDate}` di bidang ini untuk membuat parameter awalan tujuan berdasarkan nama pengguna atau tanggal upload.

- Tetapkan nilai `DestinationFileLocation` `${Transfer:UserName}` to untuk menyalin file yang diunggah ke bucket Amazon S3 yang diawali dengan nama pengguna Transfer Family yang mengunggah file tersebut.
- Tetapkan nilai `DestinationFileLocation` `${Transfer:UploadDate}` to untuk menyalin file yang diunggah ke bucket Amazon S3 yang diawali dengan tanggal unggahan.



Note

Sistem menyelesaikan `UploadDate` ke format tanggal YYYY-MM-DD, berdasarkan tanggal file diunggah. UTC

Tipe: Objek [InputFileLocation](#)

Wajib: Tidak

Name

Nama langkah, digunakan sebagai pengenalan.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 30.

Pola: `[\w-]*`

Wajib: Tidak

OverwriteExisting

Bendera yang menunjukkan apakah akan menimpa file yang ada dengan nama yang sama. Default-nya adalah FALSE.

Jika alur kerja memproses file yang memiliki nama yang sama dengan file yang ada, perilakunya adalah sebagai berikut:

- Jika `OverwriteExisting` ya `TRUE`, file yang ada diganti dengan file yang sedang diproses.
- Jika `OverwriteExisting` ya `FALSE`, tidak ada yang terjadi, dan pemrosesan alur kerja berhenti.

Tipe: String

Nilai yang Valid: `TRUE` | `FALSE`

Wajib: Tidak

SourceFileLocation

Menentukan file mana yang akan digunakan sebagai masukan ke langkah alur kerja: baik output dari langkah sebelumnya, atau file yang awalnya diunggah untuk alur kerja.

- Untuk menggunakan file sebelumnya sebagai input, masukkan `${previous.file}`. Dalam hal ini, langkah alur kerja ini menggunakan file output dari langkah alur kerja sebelumnya sebagai input. Ini adalah nilai default.
- Untuk menggunakan lokasi file yang awalnya diunggah sebagai masukan untuk langkah ini, masukkan `${original.file}`.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 256.

Pola: `\$\{(\w+.\w+)\}`

Diperlukan: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

CustomStepDetails

Setiap tipe langkah memiliki StepDetails strukturnya sendiri.

Daftar Isi

Name

Nama langkah, digunakan sebagai pengenalan.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 30.

Pola: `[\w-]*`

Wajib: Tidak

SourceFileLocation

Menentukan file mana yang akan digunakan sebagai masukan ke langkah alur kerja: baik output dari langkah sebelumnya, atau file yang awalnya diunggah untuk alur kerja.

- Untuk menggunakan file sebelumnya sebagai input, masukkan `${previous.file}`. Dalam hal ini, langkah alur kerja ini menggunakan file output dari langkah alur kerja sebelumnya sebagai input. Ini adalah nilai default.
- Untuk menggunakan lokasi file yang awalnya diunggah sebagai masukan untuk langkah ini, masukkan `${original.file}`.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 256.

Pola: `\$\{(\w+.\w+)\}`

Wajib: Tidak

Target

Fungsi ARN untuk Lambda yang sedang dipanggil.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 170.

Pola: `arn:[a-z-]+:lambda:.*`

Wajib: Tidak

TimeoutSeconds

Batas waktu, dalam hitungan detik, untuk langkah tersebut.

Jenis: Integer

Rentang yang Valid: Nilai minimum 1. Nilai maksimum 1800.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

DecryptStepDetails

Setiap tipe langkah memiliki StepDetails strukturnya sendiri.

Daftar Isi

DestinationFileLocation

Menentukan lokasi untuk file yang didekripsi. Gunakan `${Transfer:UserName}` atau `${Transfer:UploadDate}` di bidang ini untuk membuat parameter awalan tujuan berdasarkan nama pengguna atau tanggal upload.

- Tetapkan nilai DestinationFileLocation `${Transfer:UserName}` to untuk mendekripsi file yang diunggah ke bucket Amazon S3 yang diawali dengan nama pengguna Transfer Family yang mengunggah file tersebut.
- Tetapkan nilai DestinationFileLocation `${Transfer:UploadDate}` to untuk mendekripsi file yang diunggah ke bucket Amazon S3 yang diawali dengan tanggal unggahan.



Note

Sistem menyelesaikan UploadDate ke format tanggal YYYY-MM-DD, berdasarkan tanggal file diunggah. UTC

Tipe: Objek [InputFileLocation](#)

Wajib: Ya

Type

Jenis enkripsi yang digunakan. Saat ini, nilai ini harusPGP.

Tipe: String

Nilai yang Valid: PGP

Wajib: Ya

Name

Nama langkah, digunakan sebagai pengenalan.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 30.

Pola: `[\w-]*`

Wajib: Tidak

OverwriteExisting

Bendera yang menunjukkan apakah akan menimpa file yang ada dengan nama yang sama. Default-nya adalah FALSE.

Jika alur kerja memproses file yang memiliki nama yang sama dengan file yang ada, perilakunya adalah sebagai berikut:

- Jika `OverwriteExisting` ya `TRUE`, file yang ada diganti dengan file yang sedang diproses.
- Jika `OverwriteExisting` ya `FALSE`, tidak ada yang terjadi, dan pemrosesan alur kerja berhenti.

Tipe: String

Nilai yang Valid: `TRUE` | `FALSE`

Wajib: Tidak

SourceFileLocation

Menentukan file mana yang akan digunakan sebagai masukan ke langkah alur kerja: baik output dari langkah sebelumnya, atau file yang awalnya diunggah untuk alur kerja.

- Untuk menggunakan file sebelumnya sebagai input, masukkan `${previous.file}`. Dalam hal ini, langkah alur kerja ini menggunakan file output dari langkah alur kerja sebelumnya sebagai input. Ini adalah nilai default.
- Untuk menggunakan lokasi file yang awalnya diunggah sebagai masukan untuk langkah ini, masukkan `${original.file}`.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 256.

Pola: `\$\{(\w+.\w+)\}`

Diperlukan: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

DeleteStepDetails

Nama langkah, digunakan untuk mengidentifikasi langkah hapus.

Daftar Isi

Name

Nama langkah, digunakan sebagai pengidentifikasi.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 30.

Pola: `[\w-]*`

Wajib: Tidak

SourceFileLocation

Menentukan file mana yang akan digunakan sebagai input ke langkah alur kerja: baik output dari langkah sebelumnya, atau file yang awalnya diunggah untuk alur kerja.

- Untuk menggunakan file sebelumnya sebagai input, masukkan `${previous.file}`. Dalam hal ini, langkah alur kerja ini menggunakan file output dari langkah alur kerja sebelumnya sebagai input. Ini adalah nilai default.
- Untuk menggunakan lokasi file yang awalnya diunggah sebagai masukan untuk langkah ini, masukkan `${original.file}`.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 256.

Pola: `\$\{(\w+.\w+)\}`

Diperlukan: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C++](#)

- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

DescribedAccess

Menjelaskan properti akses yang ditentukan.

Daftar Isi

ExternalId

Pengidentifikasi unik yang diperlukan untuk mengidentifikasi grup tertentu dalam direktori Anda. Pengguna grup yang Anda asosiasikan memiliki akses ke EFS sumber daya Amazon S3 atau Amazon Anda melalui protokol yang diaktifkan. AWS Transfer Family Jika Anda tahu nama grup, Anda dapat melihat SID nilai dengan menjalankan perintah berikut menggunakan Windows PowerShell.

```
Get-ADGroup -Filter {samAccountName -like "YourGroupName*"} -Properties
* | Select SamAccountName, ObjectSid
```

Dalam perintah itu, ganti YourGroupName dengan nama grup Active Directory Anda.

Ekspresi reguler yang digunakan untuk memvalidasi parameter ini adalah string karakter yang terdiri dari huruf besar dan huruf kecil karakter alfanumerik tanpa spasi. Anda juga dapat menyertakan garis bawah atau salah satu karakter berikut: =, . @: /-

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 256.

Pola: S-1-[\d-]+

Wajib: Tidak

HomeDirectory

Direktori arahan (folder) untuk pengguna ketika mereka masuk ke server menggunakan klien.

Contoh HomeDirectory adalah /bucket_name/home/mydirectory.

Note

Parameter HomeDirectory hanya digunakan jika HomeDirectoryType diatur ke PATH.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 1024.

Pola: (| / . *)

Wajib: Tidak

HomeDirectoryMappings

Pemetaan direktori logis yang menentukan jalur dan kunci Amazon S3 atau EFS Amazon apa yang harus terlihat oleh pengguna Anda dan bagaimana Anda ingin membuatnya terlihat. Anda harus menentukan Entry dan Target memasangkan, di mana Entry menunjukkan bagaimana jalur dibuat terlihat dan Target merupakan jalur Amazon S3 atau Amazon yang sebenarnya. EFS Jika Anda hanya menentukan target, itu ditampilkan apa adanya. Anda juga harus memastikan bahwa peran AWS Identity and Access Management (IAM) Anda menyediakan akses ke jalur diTarget. Nilai ini dapat diatur hanya ketika HomeDirectoryType diatur ke LOGICAL.

Dalam kebanyakan kasus, Anda dapat menggunakan nilai ini alih-alih kebijakan sesi untuk mengunci akses terkait ke direktori home yang ditunjuk ("chroot"). Untuk melakukan ini, Anda dapat mengatur Entry ke '/' dan mengatur Target ke nilai HomeDirectory parameter.

Tipe: Array objek [HomeDirectoryMapEntry](#)

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 50000 item.

Wajib: Tidak

HomeDirectoryType

Jenis direktori pendaratan (folder) yang Anda inginkan direktori home pengguna Anda ketika mereka masuk ke server. Jika Anda mengaturnyaPATH, pengguna akan melihat bucket Amazon S3 absolut atau EFS jalur Amazon seperti pada klien protokol transfer file mereka. Jika Anda menyetelnyaLOGICAL, Anda harus menyediakan pemetaan HomeDirectoryMappings untuk bagaimana Anda ingin membuat jalur Amazon S3 atau EFS Amazon terlihat oleh pengguna Anda.

Note

Jika HomeDirectoryType yaLOGICAL, Anda harus memberikan pemetaan, menggunakan parameter. HomeDirectoryMappings Jika, di sisi lain, HomeDirectoryType adalahPATH, Anda memberikan jalur absolut menggunakan HomeDirectory parameter. Anda tidak dapat memiliki keduanya HomeDirectory dan HomeDirectoryMappings di template Anda.

Tipe: String

Nilai yang Valid: PATH | LOGICAL

Wajib: Tidak

Policy

Kebijakan sesi untuk pengguna Anda sehingga Anda dapat menggunakan peran AWS Identity and Access Management (IAM) yang sama di beberapa pengguna. Kebijakan ini mencakup akses pengguna ke sebagian bucket Amazon S3 mereka. Variabel yang dapat Anda gunakan dalam kebijakan ini meliputi `${Transfer:UserName}`, `${Transfer:HomeDirectory}`, dan `${Transfer:HomeBucket}`.

Tipe: String

Batasan Panjang: Panjang minimum sebesar 0. Panjang maksimum 2048.

Wajib: Tidak

PosixProfile

POSIXIdentitas lengkap, termasuk ID pengguna (Uid), ID grup (Gid), dan grup sekunder IDs (SecondaryGids) apa pun, yang mengontrol akses pengguna ke sistem EFS file Amazon Anda. POSIXIzin yang ditetapkan pada file dan direktori di sistem file Anda menentukan tingkat akses yang didapat pengguna saat mentransfer file masuk dan keluar dari sistem file Amazon EFS Anda.

Tipe: Objek [PosixProfile](#)

Wajib: Tidak

Role

Nama Sumber Daya Amazon (ARN) dari peran AWS Identity and Access Management (IAM) yang mengontrol akses pengguna ke bucket Amazon S3 atau sistem file EFS Amazon. Kebijakan yang dilampirkan pada peran ini menentukan tingkat akses yang ingin Anda berikan kepada pengguna saat mentransfer file masuk dan keluar dari bucket Amazon S3 atau sistem file EFS Amazon. IAMPeran tersebut juga harus berisi hubungan kepercayaan yang memungkinkan server mengakses sumber daya Anda saat melayani permintaan transfer pengguna Anda.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 2048.

Pola: `arn:.*role/\S+`

Diperlukan: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

DescribedAgreement

Menjelaskan sifat-sifat suatu perjanjian.

Daftar Isi

Arn

Nama Sumber Daya Amazon (ARN) yang unik untuk perjanjian.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 1600.

Pola: `arn:\S+`

Wajib: Ya

AccessRole

Konektor digunakan untuk mengirim file menggunakan SFTP protokol AS2 atau. Untuk peran akses, berikan Amazon Resource Name (ARN) AWS Identity and Access Management peran yang akan digunakan.

Untuk AS2 konektor

Dengan AS2, Anda dapat mengirim file dengan memanggil `StartFileTransfer` dan menentukan jalur file dalam parameter permintaan, `SendFilePaths`. Kami menggunakan direktori induk file (misalnya, untuk `--send-file-paths /bucket/dir/file.txt`, direktori induk `/bucket/dir/`) untuk sementara menyimpan file AS2 pesan yang diproses, menyimpan MDN ketika kami menerimanya dari mitra, dan menulis JSON file akhir yang berisi metadata transmisi yang relevan. Jadi, `AccessRole` kebutuhan untuk menyediakan akses baca dan tulis ke direktori induk dari lokasi file yang digunakan dalam `StartFileTransfer` permintaan. Selain itu, Anda perlu menyediakan akses baca dan tulis ke direktori induk dari file yang ingin Anda kirim `StartFileTransfer`.

Jika Anda menggunakan otentikasi Dasar untuk AS2 konektor Anda, peran akses memerlukan `secretsmanager:GetSecretValue` izin untuk rahasia tersebut. Jika rahasia dienkripsi menggunakan kunci yang dikelola pelanggan alih-alih kunci terkelola di AWS Secrets Manager, maka peran tersebut juga memerlukan `kms:Decrypt` izin untuk kunci tersebut.

Untuk SFTP konektor

Pastikan bahwa peran akses menyediakan akses baca dan tulis ke direktori induk dari lokasi file yang digunakan dalam `StartFileTransfer` permintaan. Selain itu, pastikan bahwa peran tersebut memberikan `secretsmanager:GetSecretValue` izin untuk AWS Secrets Manager.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 2048.

Pola: `arn:.*role/\S+`

Wajib: Tidak

AgreementId

Pengidentifikasi unik untuk perjanjian. Pengenal ini dikembalikan saat Anda membuat perjanjian.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: `a-([0-9a-f]{17})`

Wajib: Tidak

BaseDirectory

Direktori pendaratan (folder) untuk file yang ditransfer dengan menggunakan AS2 protokol.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 1024.

Pola: `(|/.*)`

Wajib: Tidak

Description

Nama atau deskripsi singkat yang digunakan untuk mengidentifikasi perjanjian.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 200.

Pola: `[\p{Graph}]+`

Wajib: Tidak

LocalProfileId

Pengenal unik untuk profil AS2 lokal.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: p-([0-9a-f]{17})

Wajib: Tidak

PartnerProfileId

Pengenal unik untuk profil mitra yang digunakan dalam perjanjian.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: p-([0-9a-f]{17})

Wajib: Tidak

ServerId

Pengidentifikasi unik yang ditetapkan sistem untuk instans server. Pengenal ini menunjukkan server spesifik yang digunakan perjanjian.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s-([0-9a-f]{17})

Wajib: Tidak

Status

Status perjanjian saat ini, baik ACTIVE atau INACTIVE.

Tipe: String

Nilai yang Valid: ACTIVE | INACTIVE

Wajib: Tidak

Tags

Pasangan nilai kunci yang dapat digunakan untuk mengelompokkan dan mencari perjanjian.

Tipe: Array objek [Tag](#)

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 50 item.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

DescribedCertificate

Menjelaskan properti sertifikat.

Daftar Isi

Arn

Nama Sumber Daya Amazon (ARN) yang unik untuk sertifikat.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 1600.

Pola: `arn:\S+`

Wajib: Ya

ActiveDate

Tanggal opsional yang menentukan kapan sertifikat menjadi aktif.

Tipe: Timestamp

Wajib: Tidak

Certificate

Nama file untuk sertifikat.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 16384.

Pola: `[\u0009\u000A\u000D\u0020-\u00FF]*`

Wajib: Tidak

CertificateChain

Daftar sertifikat yang membentuk rantai untuk sertifikat.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 2097152.

Pola: `[\u0009\u000A\u000D\u0020-\u00FF]*`

Wajib: Tidak

CertificateId

Array pengidentifikasi untuk sertifikat yang diimpor. Anda menggunakan pengenalan ini untuk bekerja dengan profil dan profil mitra.

Tipe: String

Kendala Panjang: Panjang tetap 22.

Pola: `cert-([0-9a-f]{17})`

Wajib: Tidak

Description

Nama atau deskripsi yang digunakan untuk mengidentifikasi sertifikat.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 200.

Pola: `[\p{Graph}]+`

Wajib: Tidak

InactiveDate

Tanggal opsional yang menentukan kapan sertifikat menjadi tidak aktif.

Tipe: Timestamp

Wajib: Tidak

NotAfterDate

Tanggal akhir sertifikat itu valid.

Tipe: Timestamp

Wajib: Tidak

NotBeforeDate

Tanggal paling awal bahwa sertifikat itu valid.

Tipe: Timestamp

Wajib: Tidak

Serial

Nomor seri untuk sertifikat.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 48.

Pola: `[\p{XDigit}{2}:?]*`

Wajib: Tidak

Status

Sertifikat dapat berupa `ACTIVE`, `PENDING_ROTATION`, atau `INACTIVE`.

`PENDING_ROTATION` berarti bahwa sertifikat ini akan menggantikan sertifikat saat ini ketika kedaluwarsa.

Tipe: String

Nilai yang Valid: `ACTIVE` | `PENDING_ROTATION` | `INACTIVE`

Wajib: Tidak

Tags

Pasangan nilai kunci yang dapat digunakan untuk mengelompokkan dan mencari sertifikat.

Tipe: Array objek [Tag](#)

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 50 item.

Wajib: Tidak

Type

Jika kunci pribadi telah ditentukan untuk sertifikat, jenisnya adalah `CERTIFICATE_WITH_PRIVATE_KEY`. Jika tidak ada kunci pribadi, tipenya adalah `CERTIFICATE`.

Tipe: String

Nilai yang Valid: CERTIFICATE | CERTIFICATE_WITH_PRIVATE_KEY

Wajib: Tidak

Usage

Menentukan apakah sertifikat ini digunakan untuk penandatanganan atau enkripsi.

Tipe: String

Nilai yang Valid: SIGNING | ENCRYPTION

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

DescribedConnector

Menjelaskan parameter untuk konektor, seperti yang diidentifikasi oleh `ConnectorId`.

Daftar Isi

Arn

Nama Sumber Daya Amazon (ARN) yang unik untuk konektor.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 1600.

Pola: `arn:\S+`

Wajib: Ya

AccessRole

Konektor digunakan untuk mengirim file menggunakan SFTP protokol AS2 atau. Untuk peran akses, berikan Amazon Resource Name (ARN) AWS Identity and Access Management peran yang akan digunakan.

Untuk AS2 konektor

Dengan AS2, Anda dapat mengirim file dengan memanggil `StartFileTransfer` dan menentukan jalur file dalam parameter permintaan, `SendFilePaths`. Kami menggunakan direktori induk file (misalnya, untuk `--send-file-paths /bucket/dir/file.txt`, direktori induk `/bucket/dir/`) untuk sementara menyimpan file AS2 pesan yang diproses, menyimpan MDN ketika kami menerimanya dari mitra, dan menulis JSON file akhir yang berisi metadata transmisi yang relevan. Jadi, `AccessRole` kebutuhan untuk menyediakan akses baca dan tulis ke direktori induk dari lokasi file yang digunakan dalam `StartFileTransfer` permintaan. Selain itu, Anda perlu menyediakan akses baca dan tulis ke direktori induk dari file yang ingin Anda kirim `StartFileTransfer`.

Jika Anda menggunakan otentikasi Dasar untuk AS2 konektor Anda, peran akses memerlukan `secretsmanager:GetSecretValue` izin untuk rahasia tersebut. Jika rahasia dienkripsi menggunakan kunci yang dikelola pelanggan alih-alih kunci yang dikelola di AWS Secrets Manager, maka peran tersebut juga memerlukan `kms:Decrypt` izin untuk kunci tersebut.

Untuk SFTP konektor

Pastikan bahwa peran akses menyediakan akses baca dan tulis ke direktori induk lokasi file yang digunakan dalam `StartFileTransfer` permintaan. Selain itu, pastikan bahwa peran tersebut memberikan `secretsmanager:GetSecretValue` izin untuk AWS Secrets Manager.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 2048.

Pola: `arn:.*role/\S+`

Wajib: Tidak

As2Config

Struktur yang berisi parameter untuk objek AS2 konektor.

Tipe: Objek [As2ConnectorConfig](#)

Wajib: Tidak

ConnectorId

Pengidentifikasi unik untuk konektor.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: `c-([0-9a-f]{17})`

Wajib: Tidak

LoggingRole

Amazon Resource Name (ARN) dari peran AWS Identity and Access Management (IAM) yang memungkinkan konektor mengaktifkan CloudWatch logging untuk peristiwa Amazon S3. Saat disetel, Anda dapat melihat aktivitas konektor di CloudWatch log Anda.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 2048.

Pola: `arn:.*role/\S+`

Wajib: Tidak

ServiceManagedEgressIpAddresses

Daftar alamat IP jalan keluar dari konektor ini. Alamat IP ini ditetapkan secara otomatis saat Anda membuat konektor.

Tipe: Array string

Pola: \d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3}

Wajib: Tidak

SftpConfig

Struktur yang berisi parameter untuk objek SFTP konektor.

Tipe: Objek [SftpConnectorConfig](#)

Wajib: Tidak

Tags

Pasangan nilai kunci yang dapat digunakan untuk mengelompokkan dan mencari konektor.

Tipe: Array objek [Tag](#)

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 50 item.

Wajib: Tidak

Url

URLDari pasangan AS2 atau SFTP titik akhir.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum sebesar 255.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)

- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

DescribedExecution

Detail untuk objek eksekusi.

Daftar Isi

ExecutionId

Pengidentifikasi unik untuk eksekusi alur kerja.

Tipe: String

Batas Panjang: Panjang tetap 36.

Pola: `[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}`

Wajib: Tidak

ExecutionRole

IAMPeran yang terkait dengan eksekusi.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 2048.

Pola: `arn:.*role/\S+`

Wajib: Tidak

InitialFileLocation

Struktur yang menggambarkan Amazon S3 atau lokasi EFS file. Ini adalah lokasi file ketika eksekusi dimulai: jika file sedang disalin, ini adalah lokasi file awal (sebagai lawan dari tujuan).

Tipe: Objek [FileLocation](#)

Wajib: Tidak

LoggingConfiguration

Peran IAM logging yang terkait dengan eksekusi.

Tipe: Objek [LoggingConfiguration](#)

Wajib: Tidak

PosixProfile

POSIXIdentitas lengkap, termasuk ID pengguna (Uid), ID grup (Gid), dan grup sekunder apa pun IDs (SecondaryGids), yang mengontrol akses pengguna ke sistem EFS file Amazon Anda. POSIXIzin yang ditetapkan pada file dan direktori di sistem file Anda menentukan tingkat akses yang didapat pengguna saat mentransfer file masuk dan keluar dari sistem file Amazon EFS Anda.

Tipe: Objek [PosixProfile](#)

Wajib: Tidak

Results

Struktur yang menggambarkan hasil eksekusi. Ini termasuk daftar langkah-langkah bersama dengan rincian setiap langkah, jenis kesalahan dan pesan (jika ada), dan OnExceptionSteps struktur.

Tipe: Objek [ExecutionResults](#)

Wajib: Tidak

ServiceMetadata

Objek kontainer untuk detail sesi yang terkait dengan alur kerja.

Tipe: Objek [ServiceMetadata](#)

Wajib: Tidak

Status

Status adalah salah satu eksekusi. Dapat dalam proses, selesai, pengecualian ditemui, atau menangani pengecualian.

Tipe: String

Nilai yang Valid: IN_PROGRESS | COMPLETED | EXCEPTION | HANDLING_EXCEPTION

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

DescribedHostKey

Detail untuk kunci host server.

Daftar Isi

Arn

Amazon Resource Name (ARN) yang unik untuk kunci host.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 1600.

Pola: `arn:\S+`

Wajib: Ya

DateImported

Tanggal di mana kunci host ditambahkan ke server.

Tipe: Timestamp

Wajib: Tidak

Description

Deskripsi teks untuk kunci host ini.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 200.

Pola: `[\p{Print}]*`

Wajib: Tidak

HostKeyFingerprint

Sidik jari kunci publik, yang merupakan urutan pendek byte yang digunakan untuk mengidentifikasi kunci publik yang lebih panjang.

Tipe: String

Wajib: Tidak

HostKeyId

Pengenal unik untuk kunci host.

Tipe: String

Kendala Panjang: Panjang tetap 25.

Pola: `hostkey-[0-9a-f]{17}`

Wajib: Tidak

Tags

Pasangan kunci-nilai yang dapat digunakan untuk mengelompokkan dan mencari kunci host.

Tipe: Array objek [Tag](#)

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 50 item.

Wajib: Tidak

Type

Algoritma enkripsi yang digunakan untuk kunci host. TypeParameter ditentukan dengan menggunakan salah satu nilai berikut:

- `ssh-rsa`
- `ssh-ed25519`
- `ecdsa-sha2-nistp256`
- `ecdsa-sha2-nistp384`
- `ecdsa-sha2-nistp521`

Tipe: String

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

DescribedProfile

Detail untuk AS2 profil lokal atau mitra.

Daftar Isi

Arn

Nama Sumber Daya Amazon (ARN) yang unik untuk profil.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 1600.

Pola: `arn:\S+`

Wajib: Ya

As2Id

As2Id itu adalah AS2-nama, seperti yang didefinisikan dalam [RFC4130](#). Untuk transfer masuk, ini adalah AS2-From header untuk AS2 pesan yang dikirim dari mitra. Untuk konektor keluar, ini adalah AS2-To header untuk AS2 pesan yang dikirim ke mitra menggunakan `StartFileTransfer` API operasi. ID ini tidak dapat menyertakan spasi.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 128.

Pola: `[\p{Print}\s]*`

Wajib: Tidak

CertificateIds

Array pengidentifikasi untuk sertifikat yang diimpor. Anda menggunakan pengenalan ini untuk bekerja dengan profil dan profil mitra.

Tipe: Array string

Kendala Panjang: Panjang tetap 22.

Pola: `cert-([0-9a-f]{17})`

Wajib: Tidak

ProfileId

Pengenal unik untuk AS2 profil lokal atau mitra.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: p-([0-9a-f]{17})

Wajib: Tidak

ProfileType

Menunjukkan apakah hanya akan mencantumkan profil LOCAL tipe atau hanya PARTNER mengetik profil. Jika tidak disediakan dalam permintaan, perintah mencantumkan semua jenis profil.

Tipe: String

Nilai yang Valid: LOCAL | PARTNER

Wajib: Tidak

Tags

Pasangan nilai kunci yang dapat digunakan untuk mengelompokkan dan mencari profil.

Tipe: Array objek [Tag](#)

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 50 item.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)

- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

DescribedSecurityPolicy

Menjelaskan properti kebijakan keamanan yang ditentukan. Untuk informasi selengkapnya tentang kebijakan keamanan, lihat [Bekerja dengan kebijakan keamanan](#).

Daftar Isi

SecurityPolicyName

Menentukan nama kebijakan keamanan yang dilampirkan ke server.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 100.

Pola: TransferSecurityPolicy-.*+

Wajib: Ya

Fips

Menentukan apakah kebijakan ini memungkinkan Federal Information Processing Standards (FIPS).

Tipe: Boolean

Wajib: Tidak

SshCiphers

Menentukan algoritma enkripsi sandi Secure Shell (SSH) yang diaktifkan dalam kebijakan keamanan yang dilampirkan ke server.

Tipe: Array string

Batasan Panjang: Panjang minimum sebesar 0. Panjang maksimum 50.

Wajib: Tidak

SshKexs

Menentukan algoritma enkripsi pertukaran SSH kunci (KEX) yang diaktifkan dalam kebijakan keamanan yang dilampirkan ke server.

Tipe: Array string

Batasan Panjang: Panjang minimum sebesar 0. Panjang maksimum 50.

Wajib: Tidak

SshMacs

Menentukan kode otentikasi SSH pesan (MAC) algoritma enkripsi diaktifkan dalam kebijakan keamanan yang dilampirkan ke server.

Tipe: Array string

Batasan Panjang: Panjang minimum sebesar 0. Panjang maksimum 50.

Wajib: Tidak

TlsCiphers

Menentukan algoritma enkripsi sandi Transport Layer Security (TLS) yang diaktifkan dalam kebijakan keamanan yang dilampirkan ke server.

Tipe: Array string

Batasan Panjang: Panjang minimum sebesar 0. Panjang maksimum 50.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

DescribedServer

Menjelaskan properti server berkemampuan protokol transfer file yang ditentukan.

Daftar Isi

Arn

Menentukan Amazon Resource Name (ARN) unik dari server.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 1600.

Pola: `arn:\S+`

Wajib: Ya

As2ServiceManagedEgressIpAddresses

Daftar alamat IP jalan keluar dari server ini. Alamat IP ini hanya relevan untuk server yang menggunakan AS2 protokol. Mereka digunakan untuk mengirim asinkronMDNs.

Alamat IP ini ditetapkan secara otomatis saat Anda membuat AS2 server. Selain itu, jika Anda memperbarui server yang ada dan menambahkan AS2 protokol, alamat IP statis juga ditetapkan.

Tipe: Array string

Pola: `\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3}`

Wajib: Tidak

Certificate

Menentukan AWS sertifikat ARN Certificate Manager (ACM). Diperlukan saat Protocols diatur ke FTPS.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 1600.

Wajib: Tidak

Domain

Menentukan domain sistem penyimpanan yang digunakan untuk transfer file.

Tipe: String

Nilai yang Valid: S3 | EFS

Wajib: Tidak

EndpointDetails

Pengaturan titik akhir virtual private cloud (VPC) yang dikonfigurasi untuk server Anda. Ketika Anda meng-host titik akhir Anda di dalam AndaVPC, Anda dapat membuat titik akhir Anda hanya dapat diakses oleh sumber daya di dalam AndaVPC, atau Anda dapat melampirkan alamat IP Elastic dan membuat titik akhir Anda dapat diakses oleh klien melalui internet. Grup VPC keamanan default Anda secara otomatis ditetapkan ke titik akhir Anda.

Tipe: Objek [EndpointDetails](#)

Wajib: Tidak

EndpointType

Mendefinisikan jenis titik akhir yang terhubung dengan server Anda. Jika server Anda terhubung ke VPC titik akhir, server Anda tidak dapat diakses melalui internet publik.

Tipe: String

Nilai yang Valid: PUBLIC | VPC | VPC_ENDPOINT

Wajib: Tidak

HostKeyFingerprint

Menentukan SHA256 sidik jari Base64 yang dikodekan dari kunci host server. Nilai ini setara dengan output dari `ssh-keygen -l -f my-new-server-key` perintah.

Tipe: String

Wajib: Tidak

IdentityProviderDetails

Menentukan informasi untuk memanggil otentikasi yang disediakan pelanggan. API Bidang ini tidak diisi ketika server `AWS_DIRECTORY_SERVICE` atau `SERVICE_MANAGED`.

IdentityProviderType

Tipe: Objek [IdentityProviderDetails](#)

Wajib: Tidak

IdentityProviderType

Modus otentikasi untuk server. Nilai defaultnya adalah `SERVICE_MANAGED`, yang memungkinkan Anda untuk menyimpan dan mengakses kredensi pengguna dalam layanan. AWS Transfer Family

Gunakan `AWS_DIRECTORY_SERVICE` untuk menyediakan akses ke grup Direktori Aktif di AWS Directory Service for Microsoft Active Directory atau Microsoft Active Directory di lingkungan lokal Anda atau AWS menggunakan AD Connector. Opsi ini juga mengharuskan Anda untuk memberikan ID Direktori dengan menggunakan `IdentityProviderDetails` parameter.

Gunakan nilai `API_GATEWAY` untuk mengintegrasikan dengan penyedia identitas pilihan Anda. `API_GATEWAY` mengharuskan Anda untuk menyediakan endpoint Amazon API Gateway URL untuk memanggil otentikasi dengan menggunakan parameter. `IdentityProviderDetails`

Gunakan `AWS_LAMBDA` nilai untuk langsung menggunakan AWS Lambda fungsi sebagai penyedia identitas Anda. Jika Anda memilih nilai ini, Anda harus menentukan fungsi ARN untuk Lambda dalam `Function` parameter untuk tipe `IdentityProviderDetails` data.

Tipe: String

Nilai yang Valid: `SERVICE_MANAGED` | `API_GATEWAY` | `AWS_DIRECTORY_SERVICE` | `AWS_LAMBDA`

Wajib: Tidak

LoggingRole

Nama Sumber Daya Amazon (ARN) dari peran AWS Identity and Access Management (IAM) yang memungkinkan server mengaktifkan CloudWatch pencatatan Amazon untuk Amazon S3 atau Amazon. EFS events Saat disetel, Anda dapat melihat aktivitas pengguna di CloudWatch log Anda.

Tipe: String

Batasan Panjang: Panjang minimum sebesar 0. Panjang maksimum 2048.

Pola: `(|arn:.*role/\S+)`

Wajib: Tidak

PostAuthenticationLoginBanner

Menentukan string untuk ditampilkan ketika pengguna terhubung ke server. String ini ditampilkan setelah pengguna mengautentikasi.

Note

SFTPProtokol tidak mendukung spanduk tampilan pasca-otentikasi.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 4096.

Pola: `[\x09-\x0D\x20-\x7E]*`

Wajib: Tidak

PreAuthenticationLoginBanner

Menentukan string untuk ditampilkan ketika pengguna terhubung ke server. String ini ditampilkan sebelum pengguna mengautentikasi. Misalnya, spanduk berikut menampilkan detail tentang penggunaan sistem:

```
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.
```

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 4096.

Pola: `[\x09-\x0D\x20-\x7E]*`

Wajib: Tidak

ProtocolDetails

Pengaturan protokol yang dikonfigurasi untuk server Anda.

- Untuk menunjukkan mode pasif (untuk FTP dan FTPS protokol), gunakan parameter. `PassiveIp` Masukkan alamat dotted-quad tunggal, seperti IPv4 alamat IP eksternal firewall, router, atau penyeimbang beban.

- Untuk mengabaikan kesalahan yang dihasilkan saat klien mencoba menggunakan SETSTAT perintah pada file yang Anda unggah ke bucket Amazon S3, gunakan `SetStatOption` parameter. Agar AWS Transfer Family server mengabaikan SETSTAT perintah dan mengunggah file tanpa perlu membuat perubahan apa pun pada SFTP klien Anda, tetapkan nilainya. `ENABLE_NO_OP` Jika Anda menyetel `SetStatOption` parameter `ENABLE_NO_OP`, Transfer Family akan menghasilkan entri CloudWatch log ke Amazon Logs, sehingga Anda dapat menentukan kapan klien melakukan SETSTAT panggilan.
- Untuk menentukan apakah AWS Transfer Family server Anda melanjutkan sesi terbaru yang dinegosiasikan melalui ID sesi unik, gunakan parameter `TlsSessionResumptionMode`.
- `As2Transports` menunjukkan metode transportasi untuk AS2 pesan. Saat ini, hanya HTTP didukung.

Tipe: Objek [ProtocolDetails](#)

Wajib: Tidak

Protocols

Menentukan protokol transfer file atau protokol di mana klien protokol transfer file Anda dapat terhubung ke titik akhir server Anda. Protokol yang tersedia adalah:

- SFTP(Secure Shell (SSH) Protokol Transfer File): Transfer file SSH
- FTPS(File Transfer Protocol Secure): Transfer file dengan TLS enkripsi
- FTP (Protokol Transfer File): Transfer file tidak terenkripsi
- AS2(Pernyataan Penerapan 2): digunakan untuk mengangkut data terstruktur business-to-business

Note

- Jika Anda memilih FTPS, Anda harus memilih sertifikat yang disimpan di AWS Certificate Manager (ACM) yang digunakan untuk mengidentifikasi server Anda ketika klien terhubung dengannya FTPS.
- Jika `Protocol` termasuk salah satu FTP atau FTPS, maka `EndpointType` harus VPC dan `IdentityProviderType` harus baik `AWS_DIRECTORY_SERVICE`, `AWS_LAMBDA`, atau `API_GATEWAY`.
- Jika `Protocol` termasuk FTP, maka `AddressAllocationIds` tidak dapat dikaitkan.
- Jika `Protocol` disetel hanya ke SFTP, `EndpointType` dapat diatur ke `PUBLIC` dan `IdentityProviderType` dapat disetel salah satu jenis identitas yang

didukung:SERVICE_MANAGED,AWS_DIRECTORY_SERVICE,AWS_LAMBDA, atauAPI_GATEWAY.

- Jika Protocol termasukAS2, maka EndpointType harusVPC, dan domain harus Amazon S3.

Tipe: Array string

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 4 item.

Nilai yang Valid: SFTP | FTP | FTPS | AS2

Wajib: Tidak

S3StorageOptions

Menentukan apakah atau tidak kinerja untuk direktori Amazon S3 Anda dioptimalkan. Ini dinonaktifkan secara default.

Secara default, pemetaan direktori home memiliki TYPE file. DIRECTORY Jika Anda mengaktifkan opsi ini, Anda kemudian perlu secara eksplisit mengatur HomeDirectoryMapEntry Type ke FILE jika Anda ingin pemetaan memiliki target file.

Tipe: Objek [S3StorageOptions](#)

Wajib: Tidak

SecurityPolicyName

Menentukan nama kebijakan keamanan yang dilampirkan ke server.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 100.

Pola: TransferSecurityPolicy-.*+

Wajib: Tidak

ServerId

Menentukan pengidentifikasi unik yang ditetapkan sistem untuk server yang Anda buat instance.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: `s-([0-9a-f]{17})`

Wajib: Tidak

State

Kondisi server yang dijelaskan. Nilai `ONLINE` menunjukkan bahwa server dapat menerima pekerjaan dan mentransfer file. StateNilai `OFFLINE` berarti bahwa server tidak dapat melakukan operasi transfer file.

Status `STARTING` dan `STOPPING` menunjukkan bahwa server berada dalam keadaan perantara, baik tidak sepenuhnya dapat merespons, atau tidak sepenuhnya offline. Nilai `START_FAILED` atau `STOP_FAILED` dapat menunjukkan kondisi kesalahan.

Tipe: String

Nilai yang Valid: `OFFLINE` | `ONLINE` | `STARTING` | `STOPPING` | `START_FAILED` | `STOP_FAILED`

Wajib: Tidak

StructuredLogDestinations

Menentukan grup log yang log server Anda dikirim.

Untuk menentukan grup log, Anda harus menyediakan ARN untuk grup log yang ada. Dalam hal ini, format grup log adalah sebagai berikut:

`arn:aws:logs:region-name:amazon-account-id:log-group:log-group-name:*`

Sebagai contoh, `arn:aws:logs:us-east-1:111122223333:log-group:mytestgroup:*`.

Jika sebelumnya Anda telah menentukan grup log untuk server, Anda dapat menghapusnya, dan pada dasarnya mematikan logging terstruktur, dengan memberikan nilai kosong untuk parameter ini dalam `update-server` panggilan. Sebagai contoh:

```
update-server --server-id s-1234567890abcdef0 --structured-log-destinations
```

Tipe: Array string.

Anggota Array: Jumlah minimum 0 item. Jumlah maksimum 1 item.

Batasan Panjang: Panjang minimum 20. Panjang maksimum 1600.

Pola: `arn:\S+`

Wajib: Tidak

Tags

Menentukan pasangan kunci-nilai yang dapat Anda gunakan untuk mencari dan mengelompokkan server yang ditugaskan ke server yang dijelaskan.

Tipe: Array objek [Tag](#)

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 50 item.

Wajib: Tidak

UserCount

Menentukan jumlah pengguna yang ditugaskan ke server yang Anda tentukan dengan `ServerId`

Tipe: Integer

Wajib: Tidak

WorkflowDetails

Menentukan ID alur kerja untuk alur kerja yang akan ditetapkan dan peran eksekusi yang digunakan untuk mengeksekusi alur kerja.

Selain alur kerja untuk mengeksekusi ketika file diunggah sepenuhnya, juga `WorkflowDetails` dapat berisi ID alur kerja (dan peran eksekusi) untuk alur kerja untuk mengeksekusi pada upload sebagian. Upload sebagian terjadi ketika sesi server terputus saat file masih diunggah.

Tipe: Objek [WorkflowDetails](#)

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

DescribedUser

Menjelaskan properti pengguna yang ditentukan.

Daftar Isi

Arn

Menentukan Amazon Resource Name (ARN) unik untuk pengguna yang diminta untuk dijelaskan.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 1600.

Pola: `arn:\S+`

Wajib: Ya

HomeDirectory

Direktori arahan (folder) untuk pengguna ketika mereka masuk ke server menggunakan klien.

Contoh HomeDirectory adalah `/bucket_name/home/mydirectory`.

Note

Parameter HomeDirectory hanya digunakan jika HomeDirectoryType diatur ke PATH.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 1024.

Pola: `(|/.*)`

Wajib: Tidak

HomeDirectoryMappings

Pemetaan direktori logis yang menentukan jalur dan kunci Amazon S3 atau EFS Amazon apa yang harus terlihat oleh pengguna Anda dan bagaimana Anda ingin membuatnya terlihat. Anda harus menentukan Entry dan Target memasangkan, di mana Entry menunjukkan bagaimana jalur dibuat terlihat dan Target merupakan jalur Amazon S3 atau Amazon yang sebenarnya. EFS

Jika Anda hanya menentukan target, itu ditampilkan apa adanya. Anda juga harus memastikan bahwa peran AWS Identity and Access Management (IAM) Anda menyediakan akses ke jalur diTarget. Nilai ini dapat diatur hanya ketika HomeDirectoryType diatur ke LOGICAL.

Dalam kebanyakan kasus, Anda dapat menggunakan nilai ini alih-alih kebijakan sesi untuk mengunci pengguna Anda ke direktori home yang ditunjuk (chroot[™]). Untuk melakukan ini, Anda dapat mengatur Entry ke '/' dan mengatur Target ke nilai HomeDirectory parameter.

Tipe: Array objek [HomeDirectoryMapEntry](#)

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 50000 item.

Wajib: Tidak

HomeDirectoryType

Jenis direktori pendaratan (folder) yang Anda inginkan direktori home pengguna Anda ketika mereka masuk ke server. Jika Anda mengaturnyaPATH, pengguna akan melihat bucket Amazon S3 absolut atau EFS jalur Amazon seperti pada klien protokol transfer file mereka. Jika Anda menyetelnyaLOGICAL, Anda harus menyediakan pemetaan HomeDirectoryMappings untuk bagaimana Anda ingin membuat jalur Amazon S3 atau EFS Amazon terlihat oleh pengguna Anda.

Note

Jika HomeDirectoryType yaLOGICAL, Anda harus memberikan pemetaan, menggunakan parameter. HomeDirectoryMappings Jika, di sisi lain, HomeDirectoryType adalahPATH, Anda memberikan jalur absolut menggunakan HomeDirectory parameter. Anda tidak dapat memiliki keduanya HomeDirectory dan HomeDirectoryMappings di template Anda.

Tipe: String

Nilai yang Valid: PATH | LOGICAL

Wajib: Tidak

Policy

Kebijakan sesi untuk pengguna Anda sehingga Anda dapat menggunakan peran AWS Identity and Access Management (IAM) yang sama di beberapa pengguna. Kebijakan ini mencakup

akses pengguna ke sebagian bucket Amazon S3 mereka. Variabel yang dapat Anda gunakan dalam kebijakan ini meliputi `${Transfer:UserName}`, `${Transfer:HomeDirectory}`, dan `${Transfer:HomeBucket}`.

Tipe: String

Batasan Panjang: Panjang minimum sebesar 0. Panjang maksimum 2048.

Wajib: Tidak

PosixProfile

Menentukan POSIX identitas lengkap, termasuk ID pengguna (Uid), ID grup (Gid), dan grup sekunder IDs (SecondaryGids) apa pun, yang mengontrol akses pengguna ke sistem file Amazon Elastic File System (AmazonEFS) Anda. POSIXIzin yang ditetapkan pada file dan direktori di sistem file Anda menentukan tingkat akses yang didapat pengguna saat mentransfer file masuk dan keluar dari sistem file Amazon EFS Anda.

Tipe: Objek [PosixProfile](#)

Wajib: Tidak

Role

Nama Sumber Daya Amazon (ARN) dari peran AWS Identity and Access Management (IAM) yang mengontrol akses pengguna ke bucket Amazon S3 atau sistem file EFS Amazon. Kebijakan yang dilampirkan pada peran ini menentukan tingkat akses yang ingin Anda berikan kepada pengguna saat mentransfer file masuk dan keluar dari bucket Amazon S3 atau sistem file EFS Amazon. IAMPeran tersebut juga harus berisi hubungan kepercayaan yang memungkinkan server mengakses sumber daya Anda saat melayani permintaan transfer pengguna Anda.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 2048.

Pola: `arn:.*role/\S+`

Wajib: Tidak

SshPublicKeys

Menentukan bagian kunci publik dari kunci Secure Shell (SSH) yang disimpan untuk pengguna yang dijelaskan.

Tipe: Array objek [SshPublicKey](#)

Anggota Array: Jumlah minimum 0 item. Jumlah maksimum 5 item.

Wajib: Tidak

Tags

Menentukan pasangan kunci-nilai untuk pengguna yang diminta. Tag dapat digunakan untuk mencari dan mengelompokkan pengguna untuk berbagai tujuan.

Tipe: Array objek [Tag](#)

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 50 item.

Wajib: Tidak

UserName

Menentukan nama pengguna yang diminta untuk dijelaskan. Nama pengguna digunakan untuk tujuan otentikasi. Ini adalah string yang akan digunakan oleh pengguna Anda ketika mereka masuk ke server Anda.

Tipe: String

Batasan Panjang: Panjang minimum 3. Panjang maksimum 100.

Pola: `[\w][\w@.-]{2,99}`

Diperlukan: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

DescribedWorkflow

Menjelaskan properti alur kerja yang ditentukan

Daftar Isi

Arn

Menentukan Amazon Resource Name (ARN) yang unik untuk alur kerja.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 1600.

Pola: `arn:\S+`

Wajib: Ya

Description

Menentukan deskripsi teks untuk alur kerja.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 256.

Pola: `[\w- ]*`

Wajib: Tidak

OnExceptionSteps

Menentukan langkah-langkah (tindakan) untuk mengambil jika kesalahan ditemui selama pelaksanaan alur kerja.

Tipe: Array objek [WorkflowStep](#)

Anggota Array: Jumlah minimum 0 item. Jumlah maksimum 8 item.

Wajib: Tidak

Steps

Menentukan rincian untuk langkah-langkah yang ada dalam alur kerja yang ditentukan.

Tipe: Array objek [WorkflowStep](#)

Anggota Array: Jumlah minimum 0 item. Jumlah maksimum 8 item.

Wajib: Tidak

Tags

Pasangan nilai kunci yang dapat digunakan untuk mengelompokkan dan mencari alur kerja. Tag adalah metadata yang dilampirkan ke alur kerja untuk tujuan apa pun.

Tipe: Array objek [Tag](#)

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 50 item.

Wajib: Tidak

WorkflowId

Pengidentifikasi unik untuk alur kerja.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: `w-([a-z0-9]{17})`

Diperlukan: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

EfsFileLocation

Menentukan rincian untuk lokasi file untuk file yang sedang digunakan dalam alur kerja. Hanya berlaku jika Anda menggunakan Amazon Elastic File Systems (AmazonEFS) untuk penyimpanan.

Daftar Isi

FileSystemId

Pengidentifikasi sistem file, yang ditugaskan oleh AmazonEFS.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 128.

Pola: (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})

Wajib: Tidak

Path

Pathname untuk folder yang digunakan oleh alur kerja.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 65536.

Pola: [^\x00]+

Diperlukan: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

EndpointDetails

Pengaturan titik akhir virtual private cloud (VPC) yang dikonfigurasi untuk server berkemampuan protokol transfer file Anda. Dengan VPC titik akhir, Anda dapat membatasi akses ke server dan sumber daya hanya di dalam Anda. VPC Untuk mengontrol lalu lintas internet yang masuk, panggil UpdateServer API dan lampirkan alamat IP Elastis ke titik akhir server Anda.

Note

Setelah 19 Mei 2021, Anda tidak akan dapat membuat server menggunakan EndpointType=VPC_ENDPOINT di AWS akun Anda jika akun Anda belum melakukannya sebelum 19 Mei 2021. Jika Anda telah membuat server dengan EndpointType=VPC_ENDPOINT di AWS akun Anda pada atau sebelum 19 Mei 2021, Anda tidak akan terpengaruh. Setelah tanggal ini, gunakan EndpointType =VPC. Untuk informasi selengkapnya, lihat [Menghentikan penggunaan _ VPC ENDPOINT](#).

Daftar Isi

AddressAllocationIds

Daftar alokasi alamat IDs yang diperlukan untuk melampirkan alamat IP Elastis ke titik akhir server Anda.

ID alokasi alamat sesuai dengan ID alokasi alamat IP Elastis. Nilai ini dapat diambil dari allocationId bidang dari tipe data EC2 [Alamat](#) Amazon. Salah satu cara untuk mengambil nilai ini adalah dengan memanggil file. EC2 [DescribeAddresses](#) API

Parameter ini bersifat opsional. Tetapkan parameter ini jika Anda ingin membuat VPC titik akhir Anda menghadap publik. Untuk detailnya, lihat [Membuat titik akhir yang menghadap internet untuk](#) server Anda.

Note

Properti ini hanya dapat diatur sebagai berikut:

- EndpointType harus diatur ke VPC
- Server Transfer Family harus offline.
- Anda tidak dapat mengatur parameter ini untuk server Transfer Family yang menggunakan FTP protokol.

- Server harus sudah SubnetIds terisi (SubnetIds dan AddressAllocationIds tidak dapat diperbarui secara bersamaan).
- AddressAllocationIds tidak dapat berisi duplikat, dan harus sama panjangnya dengan SubnetIds. Misalnya, jika Anda memiliki tiga subnetIds, Anda juga harus menentukan tiga alokasi IDs alamat.
- Panggil UpdateServer API untuk mengatur atau mengubah parameter ini.

Tipe: Array string

Wajib: Tidak

SecurityGroupIds

Daftar grup keamanan IDs yang tersedia untuk dilampirkan ke titik akhir server Anda.

 Note

Properti ini hanya dapat diatur saat EndpointType diatur ke VPC. Anda dapat mengedit SecurityGroupIds properti [UpdateServer](#) API hanya jika Anda mengubah EndpointType dari PUBLIC atau VPC_ENDPOINT ke VPC. Untuk mengubah grup keamanan yang terkait dengan VPC titik akhir server Anda setelah pembuatan, gunakan Amazon EC2 [ModifyVpcEndpoint](#) API.

Tipe: Array string

Kendala Panjang: Panjang minimum 11. Panjang maksimum 20.

Pola: sg-[0-9a-f]{8,17}

Wajib: Tidak

SubnetIds

Daftar subnet IDs yang diperlukan untuk meng-host endpoint server Anda di situs Anda. VPC

 Note

Properti ini hanya dapat diatur saat EndpointType diatur ke VPC.

Tipe: Array string

Wajib: Tidak

VpcEndpointId

Pengidentifikasi titik VPC akhir.

 Note

Properti ini hanya dapat diatur saat EndpointType diatur ke VPC_ENDPOINT.
Untuk informasi selengkapnya, lihat [Menghentikan penggunaan _VPC_ENDPOINT](#).

Tipe: String

Kendala Panjang: Panjang tetap 22.

Pola: vpce-[0-9a-f]{17}

Wajib: Tidak

VpcId

VPCPengidentifikasi VPC di mana titik akhir server akan di-host.

 Note

Properti ini hanya dapat diatur saat EndpointType diatur ke VPC.

Tipe: String

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)

- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

ExecutionError

Menentukan pesan kesalahan dan jenis, untuk kesalahan yang terjadi selama pelaksanaan alur kerja.

Daftar Isi

Message

Menentukan pesan deskriptif yang sesuai dengan. `ErrorType`

Tipe: String

Diperlukan: Ya

Type

Menentukan jenis kesalahan.

- `ALREADY_EXISTS`: terjadi untuk langkah salin, jika opsi `timpa` tidak dipilih dan file dengan nama yang sama sudah ada di lokasi target.
- `BAD_REQUEST`: permintaan buruk umum: misalnya, langkah yang mencoba menandai EFS file kembali `BAD_REQUEST`, karena hanya file S3 yang dapat ditandai.
- `CUSTOM_STEP_FAILED`: terjadi ketika langkah kustom memberikan callback yang menunjukkan kegagalan.
- `INTERNAL_SERVER_ERROR`: kesalahan catch-all yang dapat terjadi karena berbagai alasan.
- `NOT_FOUND`: terjadi ketika entitas yang diminta, misalnya file sumber untuk langkah penyalinan, tidak ada.
- `PERMISSION_DENIED`: terjadi jika kebijakan Anda tidak berisi izin yang benar untuk menyelesaikan satu atau beberapa langkah dalam alur kerja.
- `TIMEOUT`: terjadi ketika waktu eksekusi habis.



Note

Anda dapat mengatur `TimeoutSeconds` untuk langkah khusus, di mana saja dari 1 detik hingga 1800 detik (30 menit).

- `THROTTLED`: terjadi jika Anda melebihi tingkat isi ulang eksekusi baru dari satu alur kerja per detik.

Tipe: String

Nilai yang Valid: PERMISSION_DENIED | CUSTOM_STEP_FAILED | THROTTLED
| ALREADY_EXISTS | NOT_FOUND | BAD_REQUEST | TIMEOUT |
INTERNAL_SERVER_ERROR

Wajib: Ya

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

ExecutionResults

Menentukan langkah-langkah dalam alur kerja, serta langkah-langkah untuk mengeksekusi jika terjadi kesalahan selama eksekusi alur kerja.

Daftar Isi

OnExceptionSteps

Menentukan langkah-langkah (tindakan) untuk mengambil jika kesalahan ditemui selama pelaksanaan alur kerja.

Tipe: Array objek [ExecutionStepResult](#)

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 50 item.

Wajib: Tidak

Steps

Menentukan rincian untuk langkah-langkah yang ada dalam alur kerja yang ditentukan.

Tipe: Array objek [ExecutionStepResult](#)

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 50 item.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

ExecutionStepResult

Menentukan rincian berikut untuk langkah: kesalahan (jika ada), output (jika ada), dan jenis langkah.

Daftar Isi

Error

Menentukan rincian untuk kesalahan, jika itu terjadi selama pelaksanaan langkah alur kerja yang ditentukan.

Tipe: Objek [ExecutionError](#)

Wajib: Tidak

Outputs

Nilai untuk pasangan kunci/nilai diterapkan sebagai tag ke file. Hanya berlaku jika tipe langkahnyaTAG.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 65536.

Wajib: Tidak

StepType

Salah satu jenis langkah yang tersedia.

- **COPY**- Salin file ke lokasi lain.
- **CUSTOM**- Lakukan langkah khusus dengan target AWS Lambda fungsi.
- **DECRYPT**- Dekripsi file yang dienkripsi sebelum diunggah.
- **DELETE**- Hapus file.
- **TAG**- Tambahkan tag ke file.

Tipe: String

Nilai yang Valid: COPY | CUSTOM | TAG | DELETE | DECRYPT

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

FileLocation

Menentukan Amazon S3 EFS atau rincian file yang akan digunakan dalam langkah.

Daftar Isi

EfsFileLocation

Menentukan EFS identifier Amazon dan path untuk file yang digunakan.

Tipe: Objek [EfsFileLocation](#)

Wajib: Tidak

S3FileLocation

Menentukan rincian S3 untuk file yang digunakan, seperti bucket,ETag, dan sebagainya.

Tipe: Objek [S3FileLocation](#)

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

HomeDirectoryMapEntry

Mewakili sebuah objek yang berisi entri dan target untuk HomeDirectoryMappings.

Berikut ini adalah contoh Entry dan Target pair untuk chroot.

```
[ { "Entry": "/", "Target": "/bucket_name/home/mydirectory" } ]
```

Daftar Isi

Entry

Mewakili entri untuk HomeDirectoryMappings.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 1024.

Pola: /. *

Wajib: Ya

Target

Mewakili target peta yang digunakan dalam HomeDirectoryMapEntry.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 1024.

Pola: /. *

Wajib: Ya

Type

Menentukan jenis pemetaan. Atur tipe ke FILE jika Anda ingin pemetaan menunjuk ke file, atau DIRECTORY direktori mengarah ke direktori.

Note

Secara default, pemetaan direktori home memiliki Type of DIRECTORY saat Anda membuat server Transfer Family. Anda perlu mengatur Type secara eksplisit FILE jika Anda ingin pemetaan memiliki target file.

Tipe: String

Nilai yang Valid: FILE | DIRECTORY

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

IdentityProviderDetails

Mengembalikan informasi yang terkait dengan jenis otentikasi pengguna yang digunakan untuk pengguna server berkemampuan protokol transfer file. Server hanya dapat memiliki satu metode otentikasi.

Daftar Isi

DirectoryId

Pengidentifikasi AWS Directory Service direktori yang ingin Anda gunakan sebagai penyedia identitas Anda.

Tipe: String

Panjang Batasan: Panjang maksimum 12.

Pola: d-[0-9a-f]{10}

Wajib: Tidak

Function

Fungsi ARN untuk Lambda untuk digunakan untuk penyedia Identity.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 170.

Pola: arn:[a-z-]+:lambda:.*

Wajib: Tidak

InvocationRole

Parameter ini hanya berlaku jika Anda IdentityProviderTypeAPI_GATEWAY. Menyediakan tipe InvocationRole yang digunakan untuk mengotentikasi akun pengguna.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 2048.

Pola: arn:.*role/S+

Wajib: Tidak

SftpAuthenticationMethods

Untuk SFTP server yang diaktifkan, dan hanya untuk penyedia identitas khusus, Anda dapat menentukan apakah akan mengautentikasi menggunakan kata sandi, SSH key pair, atau keduanya.

- **PASSWORD**- pengguna harus memberikan kata sandi mereka untuk terhubung.
- **PUBLIC_KEY**- pengguna harus memberikan kunci pribadi mereka untuk terhubung.
- **PUBLIC_KEY_OR_PASSWORD**- pengguna dapat mengautentikasi dengan kata sandi atau kunci mereka. Ini adalah nilai default.
- **PUBLIC_KEY_AND_PASSWORD**- pengguna harus memberikan kunci pribadi dan kata sandi mereka untuk terhubung. Server memeriksa kunci terlebih dahulu, dan kemudian jika kuncinya valid, sistem meminta kata sandi. Jika kunci pribadi yang diberikan tidak cocok dengan kunci publik yang disimpan, otentikasi gagal.

Tipe: String

Nilai yang Valid: **PASSWORD** | **PUBLIC_KEY** | **PUBLIC_KEY_OR_PASSWORD** | **PUBLIC_KEY_AND_PASSWORD**

Wajib: Tidak

Url

Memberikan lokasi titik akhir layanan yang digunakan untuk mengotentikasi pengguna.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum sebesar 255.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)

- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

InputFileLocation

Menentukan lokasi untuk file yang sedang diproses.

Daftar Isi

EfsFileLocation

Menentukan detail untuk file Amazon Elastic File System (AmazonEFS) yang sedang didekripsi.

Tipe: Objek [EfsFileLocation](#)

Wajib: Tidak

S3FileLocation

Menentukan detail untuk file Amazon S3 yang sedang disalin atau didekripsi.

Tipe: Objek [S3InputFileLocation](#)

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

ListedAccess

Daftar properti untuk satu atau beberapa akses terkait yang ditentukan.

Daftar Isi

ExternalId

Pengidentifikasi unik yang diperlukan untuk mengidentifikasi grup tertentu dalam direktori Anda. Pengguna grup yang Anda kaitkan memiliki akses ke EFS sumber daya Amazon S3 atau Amazon Anda melalui protokol yang diaktifkan. AWS Transfer Family Jika Anda tahu nama grup, Anda dapat melihat SID nilai dengan menjalankan perintah berikut menggunakan Windows PowerShell.

```
Get-ADGroup -Filter {samAccountName -like "YourGroupName*"} -Properties  
* | Select SamAccountName, ObjectSid
```

Dalam perintah itu, ganti YourGroupName dengan nama grup Active Directory Anda.

Ekspresi reguler yang digunakan untuk memvalidasi parameter ini adalah string karakter yang terdiri dari huruf besar dan huruf kecil karakter alfanumerik tanpa spasi. Anda juga dapat menyertakan garis bawah atau salah satu karakter berikut: =, . @: /-

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 256.

Pola: S-1-[\d-]+

Wajib: Tidak

HomeDirectory

Direktori arahan (folder) untuk pengguna ketika mereka masuk ke server menggunakan klien.

Contoh HomeDirectory adalah /bucket_name/home/mydirectory.

Note

Parameter HomeDirectory hanya digunakan jika HomeDirectoryType diatur ke PATH.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 1024.

Pola: (| / . *)

Wajib: Tidak

HomeDirectoryType

Jenis direktori pendaratan (folder) yang Anda inginkan direktori home pengguna Anda ketika mereka masuk ke server. Jika Anda mengaturnya `PATH`, pengguna akan melihat bucket Amazon S3 absolut atau EFS jalur Amazon seperti pada klien protokol transfer file mereka. Jika Anda menyetelnya `LOGICAL`, Anda harus menyediakan pemetaan `HomeDirectoryMappings` untuk bagaimana Anda ingin membuat jalur Amazon S3 atau EFS Amazon terlihat oleh pengguna Anda.

Note

Jika `HomeDirectoryType` ya `LOGICAL`, Anda harus memberikan pemetaan, menggunakan parameter `HomeDirectoryMappings`. Jika, di sisi lain, `HomeDirectoryType` adalah `PATH`, Anda memberikan jalur absolut menggunakan `HomeDirectory` parameter. Anda tidak dapat memiliki keduanya `HomeDirectory` dan `HomeDirectoryMappings` di template Anda.

Tipe: String

Nilai yang Valid: `PATH` | `LOGICAL`

Wajib: Tidak

Role

Nama Sumber Daya Amazon (ARN) dari peran AWS Identity and Access Management (IAM) yang mengontrol akses pengguna ke bucket Amazon S3 atau sistem file EFS Amazon. Kebijakan yang dilampirkan pada peran ini menentukan tingkat akses yang ingin Anda berikan kepada pengguna saat mentransfer file masuk dan keluar dari bucket Amazon S3 atau sistem file EFS Amazon. IAM Peran tersebut juga harus berisi hubungan kepercayaan yang memungkinkan server mengakses sumber daya Anda saat melayani permintaan transfer pengguna Anda.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 2048.

Pola: `arn:.*role/\S+`

Diperlukan: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

ListedAgreement

Menjelaskan sifat-sifat perjanjian.

Daftar Isi

AgreementId

Pengidentifikasi unik untuk perjanjian. Pengenal ini dikembalikan saat Anda membuat perjanjian.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: a-([0-9a-f]{17})

Wajib: Tidak

Arn

Nama Sumber Daya Amazon (ARN) dari perjanjian yang ditentukan.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 1600.

Pola: arn:\S+

Wajib: Tidak

Description

Deskripsi saat ini untuk perjanjian. Anda dapat mengubahnya dengan memanggil UpdateAgreement operasi dan memberikan deskripsi baru.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 200.

Pola: [\p{Graph}]+

Wajib: Tidak

LocalProfileId

Pengidentifikasi unik untuk profil AS2 lokal.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: p-([0-9a-f]{17})

Wajib: Tidak

PartnerProfileId

Pengidentifikasi unik untuk profil mitra.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: p-([0-9a-f]{17})

Wajib: Tidak

ServerId

Pengidentifikasi unik untuk perjanjian.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: s-([0-9a-f]{17})

Wajib: Tidak

Status

Perjanjian itu bisa berupa ACTIVE atau INACTIVE.

Tipe: String

Nilai yang Valid: ACTIVE | INACTIVE

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

ListedCertificate

Menjelaskan properti sertifikat.

Daftar Isi

ActiveDate

Tanggal opsional yang menentukan kapan sertifikat menjadi aktif.

Tipe: Timestamp

Wajib: Tidak

Arn

Nama Sumber Daya Amazon (ARN) dari sertifikat yang ditentukan.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 1600.

Pola: `arn:\S+`

Wajib: Tidak

CertificateId

Array pengidentifikasi untuk sertifikat yang diimpor. Anda menggunakan pengenalan ini untuk bekerja dengan profil dan profil mitra.

Tipe: String

Kendala Panjang: Panjang tetap 22.

Pola: `cert-([0-9a-f]{17})`

Wajib: Tidak

Description

Nama atau deskripsi singkat yang digunakan untuk mengidentifikasi sertifikat.

Tipe: String

Panjang Batasan: Panjang minimum 1. Panjang maksimum 200.

Pola: $[\backslash p\{Graph\}]^+$

Wajib: Tidak

InactiveDate

Tanggal opsional yang menentukan kapan sertifikat menjadi tidak aktif.

Tipe: Timestamp

Wajib: Tidak

Status

Sertifikat dapat berupa `ACTIVE`, `PENDING_ROTATION`, atau `INACTIVE`.

`PENDING_ROTATION` berarti bahwa sertifikat ini akan menggantikan sertifikat saat ini ketika kedaluwarsa.

Tipe: String

Nilai yang Valid: `ACTIVE` | `PENDING_ROTATION` | `INACTIVE`

Wajib: Tidak

Type

Jenis untuk sertifikat. Jika kunci pribadi telah ditentukan untuk sertifikat, jenisnya adalah `CERTIFICATE_WITH_PRIVATE_KEY`. Jika tidak ada kunci pribadi, jenisnya adalah `CERTIFICATE`.

Tipe: String

Nilai yang Valid: `CERTIFICATE` | `CERTIFICATE_WITH_PRIVATE_KEY`

Wajib: Tidak

Usage

Menentukan apakah sertifikat ini digunakan untuk penandatanganan atau enkripsi.

Tipe: String

Nilai yang Valid: `SIGNING` | `ENCRYPTION`

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

ListedConnector

Mengembalikan rincian konektor yang ditentukan.

Daftar Isi

Arn

Nama Sumber Daya Amazon (ARN) dari konektor yang ditentukan.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 1600.

Pola: `arn:\S+`

Wajib: Tidak

ConnectorId

Pengidentifikasi unik untuk konektor.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: `c-([0-9a-f]{17})`

Wajib: Tidak

Url

URLDari pasangan AS2 atau SFTP titik akhir.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum sebesar 255.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

ListedExecution

Mengembalikan properti eksekusi yang ditentukan.

Daftar Isi

ExecutionId

Pengidentifikasi unik untuk eksekusi alur kerja.

Tipe: String

Batas Panjang: Panjang tetap 36.

Pola: `[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}`

Wajib: Tidak

InitialFileLocation

Struktur yang menggambarkan Amazon S3 atau lokasi EFS file. Ini adalah lokasi file ketika eksekusi dimulai: jika file sedang disalin, ini adalah lokasi file awal (sebagai lawan dari tujuan).

Tipe: Objek [FileLocation](#)

Wajib: Tidak

ServiceMetadata

Objek kontainer untuk detail sesi yang terkait dengan alur kerja.

Tipe: Objek [ServiceMetadata](#)

Wajib: Tidak

Status

Status adalah salah satu eksekusi. Dapat dalam proses, selesai, pengecualian ditemui, atau menangani pengecualian.

Tipe: String

Nilai yang Valid: IN_PROGRESS | COMPLETED | EXCEPTION | HANDLING_EXCEPTION

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

ListedHostKey

Mengembalikan properti dari kunci host yang ditentukan.

Daftar Isi

Arn

Nama Sumber Daya Amazon (ARN) yang unik dari kunci host.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 1600.

Pola: `arn:\S+`

Wajib: Ya

DateImported

Tanggal di mana kunci host ditambahkan ke server.

Tipe: Timestamp

Wajib: Tidak

Description

Deskripsi saat ini untuk kunci host. Anda dapat mengubahnya dengan memanggil `UpdateHostKey` operasi dan memberikan deskripsi baru.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 200.

Pola: `[\p{Print}]*`

Wajib: Tidak

Fingerprint

Sidik jari kunci publik, yang merupakan urutan pendek byte yang digunakan untuk mengidentifikasi kunci publik yang lebih panjang.

Tipe: String

Wajib: Tidak

HostKeyId

Pengidentifikasi unik untuk kunci host.

Tipe: String

Kendala Panjang: Panjang tetap 25.

Pola: hostkey-[0-9a-f]{17}

Wajib: Tidak

Type

Algoritma enkripsi yang digunakan untuk kunci host. TypeParameter ditentukan dengan menggunakan salah satu nilai berikut:

- ssh-rsa
- ssh-ed25519
- ecdsa-sha2-nistp256
- ecdsa-sha2-nistp384
- ecdsa-sha2-nistp521

Tipe: String

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

ListedProfile

Mengembalikan properti dari profil yang telah ditentukan.

Daftar Isi

Arn

Nama Sumber Daya Amazon (ARN) dari profil yang ditentukan.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 1600.

Pola: `arn:\S+`

Wajib: Tidak

As2Id

As2IdItu adalah AS2-nama, seperti yang didefinisikan dalam [RFC4130](#). Untuk transfer masuk, ini adalah AS2-From header untuk AS2 pesan yang dikirim dari mitra. Untuk konektor keluar, ini adalah AS2-To header untuk AS2 pesan yang dikirim ke mitra menggunakan `StartFileTransfer` API operasi. ID ini tidak dapat menyertakan spasi.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 128.

Pola: `[\p{Print}\s]*`

Wajib: Tidak

ProfileId

Pengenal unik untuk AS2 profil lokal atau mitra.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: `p-([0-9a-f]{17})`

Wajib: Tidak

ProfileType

Menunjukkan apakah hanya akan mencantumkan profil LOCAL tipe atau hanya PARTNER mengetik profil. Jika tidak disediakan dalam permintaan, perintah mencantumkan semua jenis profil.

Tipe: String

Nilai yang Valid: LOCAL | PARTNER

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

ListedServer

Mengembalikan properti dari server berkemampuan protokol transfer file yang ditentukan.

Daftar Isi

Arn

Menentukan Amazon Resource Name (ARN) unik untuk server yang akan dicantumkan.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 1600.

Pola: `arn:\S+`

Wajib: Ya

Domain

Menentukan domain sistem penyimpanan yang digunakan untuk transfer file.

Tipe: String

Nilai yang Valid: `S3` | `EFS`

Wajib: Tidak

EndpointType

Menentukan jenis VPC endpoint yang server Anda terhubung ke. Jika server Anda terhubung ke VPC titik akhir, server Anda tidak dapat diakses melalui internet publik.

Tipe: String

Nilai yang Valid: `PUBLIC` | `VPC` | `VPC_ENDPOINT`

Wajib: Tidak

IdentityProviderType

Modus otentikasi untuk server. Nilai defaultnya adalah `SERVICE_MANAGED`, yang memungkinkan Anda untuk menyimpan dan mengakses kredensi pengguna dalam layanan. AWS Transfer Family

Gunakan `AWS_DIRECTORY_SERVICE` untuk menyediakan akses ke grup Direktori Aktif di AWS Directory Service for Microsoft Active Directory atau Microsoft Active Directory di lingkungan

lokal Anda atau AWS menggunakan AD Connector. Opsi ini juga mengharuskan Anda untuk memberikan ID Direktori dengan menggunakan `IdentityProviderDetails` parameter.

Gunakan nilai `API_GATEWAY` untuk mengintegrasikan dengan penyedia identitas pilihan Anda. `API_GATEWAY` mengharuskan Anda untuk menyediakan endpoint Amazon API Gateway URL untuk memanggil otentikasi dengan menggunakan parameter `IdentityProviderDetails`.

Gunakan `AWS_LAMBDA` nilai untuk langsung menggunakan AWS Lambda fungsi sebagai penyedia identitas Anda. Jika Anda memilih nilai ini, Anda harus menentukan fungsi ARN untuk Lambda dalam `Function` parameter untuk tipe `IdentityProviderDetails` data.

Tipe: String

Nilai yang Valid: `SERVICE_MANAGED` | `API_GATEWAY` | `AWS_DIRECTORY_SERVICE` | `AWS_LAMBDA`

Wajib: Tidak

LoggingRole

Nama Sumber Daya Amazon (ARN) dari peran AWS Identity and Access Management (IAM) yang memungkinkan server mengaktifkan CloudWatch pencatatan Amazon untuk Amazon S3 atau Amazon. Event Logs Saat disetel, Anda dapat melihat aktivitas pengguna di CloudWatch log Anda.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 2048.

Pola: `arn:.*role/\S+`

Wajib: Tidak

ServerId

Menentukan sistem unik yang ditetapkan identifier untuk server yang terdaftar.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: `s-([0-9a-f]{17})`

Wajib: Tidak

State

Kondisi server yang dijelaskan. Nilai ONLINE menunjukkan bahwa server dapat menerima pekerjaan dan mentransfer file. StateNilai OFFLINE berarti bahwa server tidak dapat melakukan operasi transfer file.

Status STARTING dan STOPPING menunjukkan bahwa server berada dalam keadaan perantara, baik tidak sepenuhnya dapat merespons, atau tidak sepenuhnya offline. Nilai START_FAILED atau STOP_FAILED dapat menunjukkan kondisi kesalahan.

Tipe: String

Nilai yang Valid: OFFLINE | ONLINE | STARTING | STOPPING | START_FAILED | STOP_FAILED

Wajib: Tidak

UserCount

Menentukan jumlah pengguna yang ditugaskan ke server yang Anda tentukan dengan.

ServerId

Tipe: Integer

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

ListedUser

Mengembalikan properti dari pengguna yang Anda tentukan.

Daftar Isi

Arn

Menyediakan Amazon Resource Name (ARN) unik untuk pengguna yang ingin Anda pelajari.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 1600.

Pola: `arn:\S+`

Wajib: Ya

HomeDirectory

Direktori arahan (folder) untuk pengguna ketika mereka masuk ke server menggunakan klien.

Contoh `HomeDirectory` adalah `/bucket_name/home/mydirectory`.

Note

Parameter `HomeDirectory` hanya digunakan jika `HomeDirectoryType` diatur ke `PATH`.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 1024.

Pola: `(|/.*)`

Wajib: Tidak

HomeDirectoryType

Jenis direktori pendaratan (folder) yang Anda inginkan direktori home pengguna Anda ketika mereka masuk ke server. Jika Anda mengaturnya `PATH`, pengguna akan melihat bucket Amazon S3 absolut atau EFS jalur Amazon seperti pada klien protokol transfer file mereka. Jika Anda

menyetelnya LOGICAL, Anda harus menyediakan pemetaan HomeDirectoryMappings untuk bagaimana Anda ingin membuat jalur Amazon S3 atau EFS Amazon terlihat oleh pengguna Anda.

 Note

Jika HomeDirectoryType ya LOGICAL, Anda harus memberikan pemetaan, menggunakan parameter. HomeDirectoryMappings Jika, di sisi lain, HomeDirectoryType adalah PATH, Anda memberikan jalur absolut menggunakan HomeDirectory parameter. Anda tidak dapat memiliki keduanya HomeDirectory dan HomeDirectoryMappings di template Anda.

Tipe: String

Nilai yang Valid: PATH | LOGICAL

Wajib: Tidak

Role

Nama Sumber Daya Amazon (ARN) dari peran AWS Identity and Access Management (IAM) yang mengontrol akses pengguna ke bucket Amazon S3 atau sistem file EFS Amazon Anda. Kebijakan yang dilampirkan pada peran ini menentukan tingkat akses yang ingin Anda berikan kepada pengguna saat mentransfer file masuk dan keluar dari bucket Amazon S3 atau sistem file EFS Amazon Anda. IAMPeran tersebut juga harus berisi hubungan kepercayaan yang memungkinkan server mengakses sumber daya Anda saat melayani permintaan transfer pengguna Anda.

 Note

IAMPeran yang mengontrol akses pengguna Anda ke bucket Amazon S3 untuk server Domain=S3 dengan, atau sistem file EFS Anda untuk server dengan. Domain=EFS Kebijakan yang dilampirkan pada peran ini menentukan tingkat akses yang ingin Anda berikan kepada pengguna saat mentransfer file masuk dan keluar dari bucket atau EFS sistem file S3 Anda.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 2048.

Pola: `arn:.*role/\S+`

Wajib: Tidak

SshPublicKeyCount

Menentukan jumlah kunci SSH publik yang disimpan untuk pengguna yang Anda tentukan.

Tipe: Integer

Wajib: Tidak

UserName

Menentukan nama pengguna yang ARN telah ditentukan. Nama pengguna digunakan untuk tujuan otentikasi.

Tipe: String

Batasan Panjang: Panjang minimum 3. Panjang maksimum 100.

Pola: `[\w][\w@.-]{2,99}`

Diperlukan: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

ListedWorkflow

Berisi pengenalan, deskripsi teks, dan Amazon Resource Name (ARN) untuk alur kerja.

Daftar Isi

Arn

Menentukan Amazon Resource Name (ARN) yang unik untuk alur kerja.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 1600.

Pola: `arn:\S+`

Wajib: Tidak

Description

Menentukan deskripsi teks untuk alur kerja.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 256.

Pola: `[\w- ]*`

Wajib: Tidak

WorkflowId

Pengidentifikasi unik untuk alur kerja.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: `w-([a-z0-9]{17})`

Diperlukan: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

LoggingConfiguration

Terdiri dari peran logging dan nama grup log.

Daftar Isi

LoggingRole

Nama Sumber Daya Amazon (ARN) dari peran AWS Identity and Access Management (IAM) yang memungkinkan server mengaktifkan CloudWatch pencatatan Amazon untuk Amazon S3 atau Amazon. EFSevents Saat disetel, Anda dapat melihat aktivitas pengguna di CloudWatch log Anda.

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 2048.

Pola: `arn:.*role/\S+`

Wajib: Tidak

LogGroupName

Nama grup CloudWatch logging untuk AWS Transfer Family server tempat alur kerja ini berada.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 512.

Pola: `[\.\-_\/#A-Za-z0-9]*`

Diperlukan: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)

- [AWS SDK untuk Ruby V3](#)

PosixProfile

POSIXIdentitas lengkap, termasuk ID pengguna (Uid), ID grup (Gid), dan grup sekunder IDs (SecondaryGids) apa pun, yang mengontrol akses pengguna ke sistem EFS file Amazon Anda. POSIXIzin yang ditetapkan pada file dan direktori di sistem file Anda menentukan tingkat akses yang didapat pengguna saat mentransfer file masuk dan keluar dari sistem file Amazon EFS Anda.

Daftar Isi

Gid

ID POSIX grup yang digunakan untuk semua EFS operasi oleh pengguna ini.

Tipe: Long

Rentang yang Valid: Nilai minimum 0. Nilai maksimum 4294967295.

Wajib: Ya

Uid

ID POSIX pengguna yang digunakan untuk semua EFS operasi oleh pengguna ini.

Tipe: Long

Rentang yang Valid: Nilai minimum 0. Nilai maksimum 4294967295.

Wajib: Ya

SecondaryGids

POSIXGrup sekunder yang IDs digunakan untuk semua EFS operasi oleh pengguna ini.

Jenis: Array panjang

Anggota Array: Jumlah minimum 0 item. Jumlah maksimum 16 item.

Rentang yang Valid: Nilai minimum 0. Nilai maksimum 4294967295.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

ProtocolDetails

Pengaturan protokol yang dikonfigurasi untuk server Anda.

Daftar Isi

As2Transports

Menunjukkan metode transport untuk AS2 pesan. Saat ini, hanya HTTP didukung.

Tipe: Array string

Anggota Array: Jumlah tetap 1 item.

Nilai yang Valid: HTTP

Wajib: Tidak

PassiveIp

Menunjukkan mode pasif, untuk FTP dan FTPS protokol. Masukkan satu IPv4 alamat, seperti alamat IP publik firewall, router, atau penyeimbang beban. Sebagai contoh:

```
aws transfer update-server --protocol-details PassiveIp=0.0.0.0
```

Ganti 0.0.0.0 pada contoh di atas dengan alamat IP aktual yang ingin Anda gunakan.

Note

Jika Anda mengubah PassiveIp nilainya, Anda harus berhenti dan kemudian memulai ulang server Transfer Family Anda agar perubahan diterapkan. Untuk detail tentang penggunaan mode pasif (PASV) di NAT lingkungan, lihat [Mengonfigurasi FTPS server Anda di belakang firewall atau NAT dengan AWS Transfer Family](#).

Nilai khusus

AUTO Dan 0.0.0.0 merupakan nilai khusus untuk PassiveIp parameter. Nilai ditetapkan PassiveIp=AUTO secara default ke FTP dan FTPS ketika server. Dalam hal ini, server secara otomatis merespons dengan salah satu titik akhir IPs dalam respons. PASV PassiveIp=0.0.0.0 memiliki aplikasi yang lebih unik untuk penggunaannya. Misalnya, jika Anda memiliki lingkungan High Availability (HA) Network Load Balancer (NLB), di mana Anda

memiliki 3 subnet, Anda hanya dapat menentukan satu alamat IP menggunakan parameter. `PassiveIp` ini mengurangi efektivitas memiliki Ketersediaan Tinggi. Dalam hal ini, Anda dapat menentukan `PassiveIp=0.0.0.0`. Ini memberitahu klien untuk menggunakan alamat IP yang sama dengan koneksi Control dan memanfaatkan semua AZs untuk koneksi mereka. Namun, perhatikan bahwa tidak semua FTP klien mendukung `PassiveIp=0.0.0.0` respons tersebut. FileZilla dan Win SCP mendukungnya. Jika Anda menggunakan klien lain, periksa untuk melihat apakah klien Anda mendukung `PassiveIp=0.0.0.0` respons.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 15.

Wajib: Tidak

SetStatOption

Gunakan file `SetStatOption` untuk mengabaikan kesalahan yang dihasilkan saat klien mencoba menggunakan SETSTAT pada file yang Anda unggah ke bucket S3.

Beberapa klien transfer SFTP file dapat mencoba mengubah atribut file jarak jauh, termasuk stempel waktu dan izin, menggunakan perintah, seperti SETSTAT saat mengunggah file. Namun, perintah ini tidak kompatibel dengan sistem penyimpanan objek, seperti Amazon S3. Karena ketidakcocokan ini, unggahan file dari klien ini dapat mengakibatkan kesalahan bahkan ketika file tersebut berhasil diunggah.

Tetapkan nilai `ENABLE_NO_OP` agar server Transfer Family mengabaikan SETSTAT perintah, dan unggah file tanpa perlu membuat perubahan apa pun pada SFTP klien Anda. Meskipun `SetStatOption ENABLE_NO_OP` pengaturan mengabaikan kesalahan, itu menghasilkan entri log di Amazon CloudWatch Logs, sehingga Anda dapat menentukan kapan klien melakukan SETSTAT panggilan.

Note

Jika Anda ingin mempertahankan stempel waktu asli untuk file Anda, dan memodifikasi atribut file lain menggunakan SETSTAT, Anda dapat menggunakan Amazon EFS sebagai penyimpanan backend dengan Transfer Family.

Tipe: String

Nilai yang Valid: DEFAULT | ENABLE_NO_OP

Wajib: Tidak

TlsSessionResumptionMode

Properti yang digunakan dengan server Transfer Family yang menggunakan FTPS protokol. TLS Session Resumption menyediakan mekanisme untuk melanjutkan atau berbagi kunci rahasia yang dinegosiasikan antara kontrol dan koneksi data untuk suatu sesi. FTPS `TlsSessionResumptionMode` menentukan apakah server melanjutkan sesi terbaru yang dinegosiasikan melalui ID sesi unik atau tidak. Akomodasi ini tersedia selama `CreateServer` dan `UpdateServer` panggilan telepon. Jika `TlsSessionResumptionMode` nilai tidak ditentukan selama `CreateServer`, itu diatur ke secara ENFORCED default.

- **DISABLED:** server tidak memproses permintaan klien dimulainya kembali TLS sesi dan membuat TLS sesi baru untuk setiap permintaan.
- **ENABLED:** server memproses dan menerima klien yang melakukan dimulainya kembali TLS sesi. Server tidak menolak koneksi data klien yang tidak melakukan pemrosesan klien dimulainya kembali TLS sesi.
- **ENFORCED:** server memproses dan menerima klien yang melakukan dimulainya kembali TLS sesi. Server menolak koneksi data klien yang tidak melakukan pemrosesan klien dimulainya kembali TLS sesi. Sebelum Anda menetapkan nilainya **ENFORCED**, uji klien Anda.

Note

Tidak semua FTPS klien melakukan dimulainya kembali TLS sesi. Jadi, jika Anda memilih untuk menegakkan dimulainya kembali TLS sesi, Anda mencegah koneksi dari FTPS klien yang tidak melakukan negosiasi protokol. Untuk menentukan apakah Anda dapat menggunakan **ENFORCED** nilainya atau tidak, Anda perlu menguji klien Anda.

Tipe: String

Nilai yang Valid: **DISABLED** | **ENABLED** | **ENFORCED**

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

S3FileLocation

Menentukan rincian untuk lokasi file untuk file yang sedang digunakan dalam alur kerja. Hanya berlaku jika Anda menggunakan penyimpanan S3.

Daftar Isi

Bucket

Menentukan bucket S3 yang berisi file yang digunakan.

Tipe: String

Batasan Panjang: Panjang minimum 3. Panjang maksimum 63.

Pola: `[a-z0-9][\.\-a-z0-9]{1,61}[a-z0-9]`

Wajib: Tidak

Etag

Tag entitas adalah hash dari objek. ETagMerefleksikan perubahan hanya pada isi objek, bukan metadatanya.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 65536.

Pola: `.+`

Wajib: Tidak

Key

Nama yang ditetapkan ke file saat dibuat di Amazon S3. Anda harus menggunakan kunci objek tersebut untuk mengambil objeknya.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 1024.

Pola: `[\P{M}\p{M}]*`

Wajib: Tidak

VersionId

Menentukan versi file.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 1024.

Pola: . +

Diperlukan: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

S3InputFileLocation

Menentukan lokasi file Amazon S3 masukan pelanggan. Jika digunakan di `dalamcopyStepDetails.DestinationFileLocation`, itu harus menjadi tujuan salinan S3.

Anda perlu menyediakan ember dan kunci. Kunci dapat mewakili jalur atau file. Ini ditentukan oleh apakah Anda mengakhiri nilai kunci dengan karakter garis miring maju (/). Jika karakter terakhir adalah "/", maka file Anda disalin ke folder, dan namanya tidak berubah. Jika, sebaliknya, karakter terakhir adalah alfanumerik, file yang Anda unggah diganti namanya menjadi nilai jalur. Dalam hal ini, jika file dengan nama itu sudah ada, itu akan ditimpa.

Misalnya, jika jalur `Andashared-files/bob/`, file yang Anda unggah disalin ke folder `shared-files/bob/`. Jika jalur `Andashared-files/today`, setiap file yang diunggah disalin ke `shared-files` folder dan diberi nama `today`: setiap unggahan menimpa versi file bob sebelumnya.

Daftar Isi

Bucket

Menentukan bucket S3 untuk file input pelanggan.

Tipe: String

Batasan Panjang: Panjang minimum 3. Panjang maksimum 63.

Pola: `[a-z0-9][\.-a-z0-9]{1,61}[a-z0-9]`

Wajib: Tidak

Key

Nama yang ditetapkan ke file saat dibuat di Amazon S3. Anda harus menggunakan kunci objek tersebut untuk mengambil objeknya.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 1024.

Pola: `[\P{M}\p{M}]*`

Diperlukan: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

S3StorageOptions

Opsi penyimpanan Amazon S3 yang dikonfigurasi untuk server Anda.

Daftar Isi

DirectoryListingOptimization

Menentukan apakah atau tidak kinerja untuk direktori Amazon S3 Anda dioptimalkan. Ini dinonaktifkan secara default.

Secara default, pemetaan direktori home memiliki TYPE file. DIRECTORY Jika Anda mengaktifkan opsi ini, Anda kemudian perlu secara eksplisit menyetel HomeDirectoryMapEntry Type ke FILE jika Anda ingin pemetaan memiliki target file.

Tipe: String

Nilai yang Valid: ENABLED | DISABLED

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

S3Tag

Menentukan pasangan kunci-nilai yang ditugaskan ke file selama pelaksanaan langkah Tagging.

Daftar Isi

Key

Nama yang ditetapkan untuk tag yang Anda buat.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 128.

Pola: (`[\\p{L}\\p{Z}\\p{N}_.: /+=\\-@]*`)

Wajib: Ya

Value

Nilai yang sesuai dengan kunci.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 256.

Pola: (`[\\p{L}\\p{Z}\\p{N}_.: /+=\\-@]*`)

Diperlukan: Ya

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

ServiceMetadata

Objek kontainer untuk detail sesi yang terkait dengan alur kerja.

Daftar Isi

UserDetails

Server ID (`ServerId`), Session ID (`SessionId`) dan user (`UserName`) membentuk `fileUserDetails`.

Tipe: Objek [UserDetails](#)

Wajib: Ya

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

SftpConnectorConfig

Berisi rincian untuk objek SFTP konektor. Objek konektor digunakan untuk mentransfer file ke dan dari SFTP server mitra.

Note

Karena tipe `SftpConnectorConfig` data digunakan untuk membuat dan memperbarui SFTP konektor, parameternya, `TrustedHostKeys` dan `UserSecretId` ditandai sebagai tidak diperlukan. Ini agak menyesatkan, karena tidak diperlukan saat Anda memperbarui SFTP konektor yang ada, tetapi diperlukan saat Anda membuat SFTP konektor baru.

Daftar Isi

TrustedHostKeys

Bagian publik dari kunci host, atau kunci, yang digunakan untuk mengidentifikasi server eksternal yang Anda hubungkan. Anda dapat menggunakan `ssh-keyscan` perintah terhadap SFTP server untuk mengambil kunci yang diperlukan.

Tiga elemen format kunci SSH publik standar adalah `<key type>`, `<body base64>`, dan opsional `<comment>`, dengan spasi di antara setiap elemen. Tentukan hanya `<key type>` dan `<body base64>`: jangan masukkan `<comment>` bagian kunci.

Untuk kunci host tepercaya, AWS Transfer Family menerima RSA dan ECDSA kunci.

- Untuk RSA kunci, `<key type>` string adalah `ssh-rsa`.
- Untuk ECDSA kunci, `<key type>` string adalah `ecdsa-sha2-nistp256`, `ecdsa-sha2-nistp384`, atau `ecdsa-sha2-nistp521`, tergantung pada ukuran kunci yang Anda hasilkan.

Jalankan perintah ini untuk mengambil kunci host SFTP server, di mana nama SFTP server Anda berada `ftp.host.com`.

```
ssh-keyscan ftp.host.com
```

Ini mencetak kunci host publik ke output standar.

```
ftp.host.com ssh-rsa AAAAB3Nza...<long-string-for-public-key>
```

Salin dan tempel string ini ke TrustedHostKeys bidang untuk create-connector perintah atau ke bidang Kunci host tepercaya di konsol.

Tipe: Array string

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 10 item.

Batasan Panjang: Panjang minimum 1. Panjang maksimum 2048.

Wajib: Tidak

UserSecretId

Pengidentifikasi rahasia (di AWS Secrets Manager) yang berisi kunci pribadi SFTP pengguna, kata sandi, atau keduanya. Pengenal harus berupa Amazon Resource Name (ARN) dari rahasia.

Tipe: String

Batasan Panjang: Panjang minimum 1. Panjang maksimum 2048.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

SshPublicKey

Memberikan informasi tentang kunci Secure Shell (SSH) publik yang terkait dengan pengguna Transfer Family untuk server berkemampuan protokol transfer file tertentu (sebagaimana diidentifikasi oleh). `ServerId` Informasi yang dikembalikan mencakup tanggal kunci diimpor, konten kunci publik, dan ID kunci publik. Pengguna dapat menyimpan lebih dari satu kunci SSH publik yang terkait dengan nama pengguna mereka di server tertentu.

Daftar Isi

`DateImported`

Menentukan tanggal ketika kunci publik ditambahkan ke pengguna Transfer Family.

Tipe: Timestamp

Wajib: Ya

`SshPublicKeyBody`

Menentukan isi dari kunci SSH publik seperti yang ditentukan oleh. `PublicKeyId`

AWS Transfer Family menerima RSA, ECDSA, dan ED25519 kunci.

Tipe: String

Batasan Panjang: Panjang minimum sebesar 0. Panjang maksimum 2048.

Wajib: Ya

`SshPublicKeyId`

Menentukan `SshPublicKeyId` parameter berisi identifier dari kunci publik.

Tipe: String

Kendala Panjang: Panjang tetap 21.

Pola: `key-[0-9a-f]{17}`

Diperlukan: Ya

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

Tag

Membuat pasangan kunci-nilai untuk sumber daya tertentu. Tag adalah metadata yang dapat Anda gunakan untuk mencari dan mengelompokkan sumber daya untuk berbagai tujuan. Anda dapat menerapkan tag ke server, pengguna, dan peran. Kunci tag dapat mengambil lebih dari satu nilai. Misalnya, untuk mengelompokkan server untuk tujuan akuntansi, Anda dapat membuat tag yang disebut Group dan menetapkan nilai Research dan Accounting ke grup itu.

Daftar Isi

Key

Nama yang ditetapkan untuk tag yang Anda buat.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 128.

Wajib: Ya

Value

Berisi satu atau beberapa nilai yang Anda tetapkan ke nama kunci yang Anda buat.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 256.

Wajib: Ya

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

TagStepDetails

Setiap tipe langkah memiliki `StepDetails` strukturnya sendiri.

Pasangan kunci/nilai yang digunakan untuk menandai file selama eksekusi langkah alur kerja.

Daftar Isi

Name

Nama langkah, digunakan sebagai pengenalan.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 30.

Pola: `[\w-]*`

Wajib: Tidak

SourceFileLocation

Menentukan file mana yang akan digunakan sebagai masukan ke langkah alur kerja: baik output dari langkah sebelumnya, atau file yang awalnya diunggah untuk alur kerja.

- Untuk menggunakan file sebelumnya sebagai input, masukkan `${previous.file}`. Dalam hal ini, langkah alur kerja ini menggunakan file output dari langkah alur kerja sebelumnya sebagai input. Ini adalah nilai default.
- Untuk menggunakan lokasi file yang awalnya diunggah sebagai masukan untuk langkah ini, masukkan `${original.file}`.

Tipe: String

Batasan Panjang: Panjang minimum 0. Panjang maksimum 256.

Pola: `\$\{(\w+.\w+)\}`

Wajib: Tidak

Tags

Array yang berisi dari 1 hingga 10 pasangan kunci/nilai.

Tipe: Array objek [S3Tag](#)

Anggota Array: Jumlah minimum 1 item. Jumlah maksimum 10 item.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

UserDetails

Menentukan nama pengguna, ID server, dan ID sesi untuk alur kerja.

Daftar Isi

ServerId

Pengidentifikasi unik yang ditetapkan sistem untuk instance server Transfer.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: `s-([0-9a-f]{17})`

Wajib: Ya

UserName

String unik yang mengidentifikasi pengguna Transfer Family yang terkait dengan server.

Tipe: String

Batasan Panjang: Panjang minimum 3. Panjang maksimum 100.

Pola: `[\w][\w@.-]{2,99}`

Wajib: Ya

SessionId

Pengidentifikasi unik yang ditetapkan sistem untuk sesi yang sesuai dengan alur kerja.

Tipe: String

Batasan Panjang: Panjang minimum 3. Panjang maksimum sebesar 32.

Pola: `[\w-]*`

Diperlukan: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C ++](#)
- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

WorkflowDetail

Menentukan ID alur kerja untuk alur kerja yang akan ditetapkan dan peran eksekusi yang digunakan untuk mengeksekusi alur kerja.

Selain alur kerja untuk mengeksekusi ketika file diunggah sepenuhnya, juga `WorkflowDetails` dapat berisi ID alur kerja (dan peran eksekusi) untuk alur kerja untuk mengeksekusi pada upload sebagian. Upload sebagian terjadi ketika sesi server terputus saat file masih diunggah.

Daftar Isi

ExecutionRole

Termasuk izin yang diperlukan untuk operasi S3EFS, dan Lambda yang dapat diasumsikan oleh Transfer, sehingga semua langkah alur kerja dapat beroperasi pada sumber daya yang diperlukan

Tipe: String

Batasan Panjang: Panjang minimum 20. Panjang maksimum 2048.

Pola: `arn:.*role/\S+`

Wajib: Ya

WorkflowId

Pengidentifikasi unik untuk alur kerja.

Tipe: String

Kendala Panjang: Panjang tetap 19.

Pola: `w-([a-z0-9]{17})`

Diperlukan: Ya

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C++](#)

- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

WorkflowDetails

Wadah untuk tipe `WorkflowDetail` data. Ini digunakan oleh tindakan yang memicu alur kerja untuk memulai eksekusi.

Daftar Isi

OnPartialUpload

Pemicu yang memulai alur kerja jika file hanya diunggah sebagian. Anda dapat melampirkan alur kerja ke server yang mengeksekusi setiap kali ada unggahan sebagian.

Unggahan sebagian terjadi saat file terbuka saat sesi terputus.

Tipe: Array objek [WorkflowDetail](#)

Anggota Array: Jumlah minimum 0 item. Jumlah maksimum 1 item.

Wajib: Tidak

OnUpload

Pemicu yang memulai alur kerja: alur kerja mulai dijalankan setelah file diunggah.

Untuk menghapus alur kerja terkait dari server, Anda dapat memberikan `OnUpload` objek kosong, seperti pada contoh berikut.

```
aws transfer update-server --server-id s-01234567890abcdef --workflow-  
details '{"OnUpload":[]}'
```

Tipe: Array objek [WorkflowDetail](#)

Anggota Array: Jumlah minimum 0 item. Jumlah maksimum 1 item.

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C++](#)

- [AWS SDK untuk Go](#)
- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

WorkflowStep

Blok bangunan dasar alur kerja.

Daftar Isi

CopyStepDetails

Detail untuk langkah yang melakukan salinan file.

Terdiri dari nilai-nilai berikut:

- Deskripsi
- Lokasi Amazon S3 untuk tujuan salinan file.
- Bendera yang menunjukkan apakah akan menimpa file yang ada dengan nama yang sama. Default-nya adalah FALSE.

Tipe: Objek [CopyStepDetails](#)

Wajib: Tidak

CustomStepDetails

Detail untuk langkah yang memanggil AWS Lambda fungsi.

Terdiri dari nama, target, dan batas waktu fungsi Lambda (dalam hitungan detik).

Tipe: Objek [CustomStepDetails](#)

Wajib: Tidak

DecryptStepDetails

Detail untuk langkah yang mendekripsi file terenkripsi.

Terdiri dari nilai-nilai berikut:

- Nama deskriptif
- Lokasi Amazon S3 atau Amazon Elastic File System EFS (Amazon) untuk file sumber untuk didekripsi.
- EFS Lokasi S3 atau Amazon untuk tujuan dekripsi file.
- Bendera yang menunjukkan apakah akan menimpa file yang ada dengan nama yang sama. Default-nya adalah FALSE.
- Jenis enkripsi yang digunakan Saat ini, hanya PGP enkripsi yang didukung.

Tipe: Objek [DecryptStepDetails](#)

Wajib: Tidak

DeleteStepDetails

Detail untuk langkah yang menghapus file.

Tipe: Objek [DeleteStepDetails](#)

Wajib: Tidak

TagStepDetails

Detail untuk langkah yang membuat satu atau beberapa tag.

Anda menentukan satu atau lebih tag. Setiap tag berisi pasangan kunci-nilai.

Tipe: Objek [TagStepDetails](#)

Wajib: Tidak

Type

Saat ini, jenis langkah berikut didukung.

- **COPY**- Salin file ke lokasi lain.
- **CUSTOM**- Lakukan langkah kustom dengan target AWS Lambda fungsi.
- **DECRYPT**- Dekripsi file yang dienkripsi sebelum diunggah.
- **DELETE**- Hapus file.
- **TAG**- Tambahkan tag ke file.

Tipe: String

Nilai yang Valid: COPY | CUSTOM | TAG | DELETE | DECRYPT

Wajib: Tidak

Lihat Juga

Untuk informasi selengkapnya tentang penggunaan ini API di salah satu bahasa khusus AWS SDKs, lihat berikut ini:

- [AWS SDK untuk C++](#)
- [AWS SDK untuk Go](#)

- [AWS SDK untuk Java V2](#)
- [AWS SDK untuk Ruby V3](#)

Membuat permintaan API

Selain menggunakan konsol, Anda dapat menggunakan AWS Transfer Family API untuk mengonfigurasi dan mengelola server secara terprogram. Bagian ini menjelaskan AWS Transfer Family operasi, penandatanganan permintaan untuk otentikasi, dan penanganan kesalahan. Untuk informasi tentang wilayah dan titik akhir yang tersedia untuk Transfer Family, lihat [AWS Transfer Family titik akhir dan kuota](#) di Referensi Umum AWS

Note

Anda juga dapat menggunakan AWS SDK saat mengembangkan aplikasi dengan Transfer Family;. AWSSDK untuk Java, .NET, dan PHP membungkus Transfer Family API yang mendasarinya, menyederhanakan tugas pemrograman Anda. Untuk informasi tentang mengunduh perpustakaan SDK, lihat [Perpustakaan kode sampel](#).

Topik

- [Transfer Family membutuhkan header permintaan](#)
- [Transfer Family meminta masukan dan penandatanganan](#)
- [Tanggapan kesalahan](#)
- [Pustaka yang tersedia](#)

Transfer Family membutuhkan header permintaan

Bagian ini menjelaskan header yang diperlukan yang harus Anda kirim dengan setiap permintaan POST. AWS Transfer Family Anda menyertakan header HTTP untuk mengidentifikasi informasi kunci tentang permintaan termasuk operasi yang ingin Anda panggil, tanggal permintaan, dan informasi yang menunjukkan otorisasi Anda sebagai pengirim permintaan. Header tidak peka huruf besar/kecil dan urutan header tidak penting.

Contoh berikut menunjukkan header yang digunakan dalam [ListServers](#) operasi.

```
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}
```

Berikut ini adalah header yang harus disertakan dengan permintaan POST Anda ke Transfer Family. Header yang ditunjukkan di bawah ini yang dimulai dengan “x-amz” khusus untuk AWS. Semua header lain yang terdaftar adalah header umum yang digunakan dalam transaksi HTTP.

Header	Deskripsi
Authorization	Header otorisasi diperlukan. Formatnya adalah tanda tangan permintaan Sigv4 standar, yang didokumentasikan pada permintaan AWSAPI Penandatanganan .
Content-Type	Gunakan <code>application/x-amz-json-1.1</code> sebagai jenis konten untuk semua permintaan ke Transfer Family. Content-Type: application/x-amz-json-1.1
Host	Gunakan header host untuk menentukan titik akhir Transfer Family tempat Anda mengirim permintaan. Misalnya, <code>transfer.us-east-1.amazonaws.com</code> adalah titik akhir untuk wilayah AS Timur (Ohio). Untuk informasi selengkapnya tentang titik akhir yang tersedia untuk Transfer Family, lihat AWS Transfer Family titik akhir dan kuota di Referensi Umum AWS Host: transfer. <i>region</i> .amazonaws.com
x-amz-date	Anda harus memberikan cap waktu baik di Date header HTTP atau AWS x-amz-date header. (Beberapa pustaka klien HTTP tidak

Header	Deskripsi
	mengizinkan Anda mengatur Date header.) Saat <code>x-amz-date</code> header hadir, Transfer Family mengabaikan Date header apa pun selama otentikasi permintaan. Formatnya harus ISO8601, <code>x-amz-date</code> dalam format <code>YYYYMMDD'T'HHMMSS'Z'</code>. <pre>x-amz-date: <i>YYYYMMDD'T'HHMMSS'Z'</i></pre>
<code>x-amz-target</code>	Header ini menentukan versi API dan operasi yang Anda minta. Nilai header target dibentuk dengan menggabungkan versi API dengan nama API dan dalam format berikut. <pre>x-amz-target: TransferService. <i>operationName</i></pre> Nilai <code>operationName</code> (<code>ListServers</code> misalnya) dapat ditemukan dari daftar API, ListServers
<code>x-amz-security-token</code>	Header ini diperlukan ketika kredensi yang digunakan untuk menandatangani permintaan bersifat sementara atau kredensi sesi (untuk detailnya, lihat Menggunakan kredensial sementara dengan AWS sumber daya di Panduan Pengguna IAM. Lihat Menambahkan tanda tangan ke permintaan HTTP di Referensi Umum Amazon Web Services untuk informasi selengkapnya.

Transfer Family meminta masukan dan penandatanganan

Semua input permintaan harus dikirim sebagai bagian dari muatan JSON di badan permintaan. Untuk Tindakan di mana semua bidang permintaan bersifat opsional `ListServers`, misalnya, Anda masih perlu menyediakan objek JSON kosong di badan permintaan, seperti `{}`. Struktur permintaan/respons payload Transfer Family didokumentasikan dalam referensi API yang ada, misalnya, [DescribeServer](#)

Transfer Family mendukung otentikasi menggunakan AWS Signature Versi 4. Untuk detailnya, lihat [Menandatangani permintaan AWS API](#).

Tanggapan kesalahan

Ketika ada kesalahan, informasi header respons berisi:

- Tipe Konten: `application/x-amz-json-1.1`
- Kode status yang sesuai 4xx atau 5xx HTTP

Tubuh respons kesalahan berisi informasi tentang kesalahan yang terjadi. Contoh respon kesalahan berikut menunjukkan sintaks output elemen respon umum untuk semua respon kesalahan.

```
{
  "__type": "String",
  "Message": "String", <!-- Message is lowercase in some instances -->
  "Resource": String,
  "ResourceType": String
  "RetryAfterSeconds": String
}
```

Tabel berikut menjelaskan bidang respons kesalahan JSON yang ditunjukkan dalam sintaks sebelumnya.

__jenis

Salah satu pengecualian dari panggilan Transfer Family API.

Jenis: String

Pesan atau pesan

Salah satu pesan kode kesalahan operasi.

Note

Beberapa pengecualian digunakan `message`, dan yang lainnya menggunakan `Message`. Anda dapat memeriksa kode untuk antarmuka Anda untuk menentukan kasus yang tepat. Atau, Anda dapat menguji setiap opsi untuk melihat mana yang berfungsi.

Jenis: String

Sumber

Sumber daya yang kesalahannya dipanggil. Misalnya, jika Anda mencoba membuat pengguna yang sudah ada, itu Resource adalah nama pengguna untuk pengguna yang ada.

Jenis: String

ResourceType

Jenis sumber daya yang kesalahannya dipanggil. Misalnya, jika Anda mencoba membuat pengguna yang sudah ada, ResourceType isUser.

Jenis: String

RetryAfterSeconds

Jumlah detik untuk menunggu sebelum mencoba kembali perintah.

Jenis: String

Contoh respons kesalahan

Badan JSON berikut dikembalikan jika Anda memanggil DescribeServer API dan menentukan server yang tidak ada.

```
{
  "__type": "ResourceNotFoundException",
  "Message": "Unknown server",
  "Resource": "s-11112222333344444",
  "ResourceType": "Server"
}
```

Badan JSON berikut dikembalikan jika menjalankan API menyebabkan pelambatan terjadi.

```
{
  "__type": "ThrottlingException",
  "RetryAfterSeconds": "1"
}
```

Badan JSON berikut dikembalikan jika Anda menggunakan CreateServer API dan Anda tidak memiliki izin yang cukup untuk membuat server Transfer Family.

```
{
  "__type": "AccessDeniedException",
  "Message": "You do not have sufficient access to perform this action."
}
```

Badan JSON berikut dikembalikan jika Anda menggunakan CreateUser API dan menentukan pengguna yang sudah ada.

```
{
  "__type": "ResourceExistsException",
  "Message": "User already exists",
  "Resource": "Alejandro-Rosalez",
  "ResourceType": "User"
}
```

Pustaka yang tersedia

AWS menyediakan pustaka, kode sampel, tutorial, dan sumber daya lainnya untuk pengembang perangkat lunak yang lebih suka membangun aplikasi menggunakan API khusus bahasa alih-alih alat baris perintah dan API Kueri. Pustaka ini menyediakan fungsi dasar (tidak termasuk dalam API), seperti otentikasi permintaan, percobaan ulang permintaan, dan penanganan kesalahan sehingga lebih mudah untuk memulai. Lihat [Alat untuk dibangun AWS](#)

Untuk pustaka dan kode sampel dalam semua bahasa, lihat [Contoh kode & pustaka](#).

Parameter Umum

Daftar berikut berisi parameter yang digunakan semua tindakan untuk menandatangani permintaan Tanda Tangan Versi 4 dengan string kueri. Setiap parameter khusus tindakan tercantum dalam topik untuk tindakan tersebut. Untuk informasi selengkapnya tentang Tanda Tangan Versi 4, lihat [Menandatangani permintaan AWS API](#) di Panduan Pengguna IAM.

Action

Tindakan yang harus dilakukan.

Tipe: string

Wajib: Ya

Version

Versi API yang ditulis dalam permintaan, dinyatakan dalam format HH-BB-TTTT.

Tipe: string

Wajib: Ya

X-Amz-Algorithm

Algoritme hash yang Anda gunakan untuk membuat tanda tangan permintaan.

Syarat: Tentukan parameter ini ketika Anda menyertakan informasi autentikasi dalam string kueri alih-alih di header otorisasi HTTP.

Tipe: string

Nilai yang Valid: AWS4-HMAC-SHA256

Diperlukan: Kondisional

X-Amz-Credential

Nilai lingkup kredensial, yang merupakan string yang menyertakan access key Anda, tanggal, wilayah yang Anda targetkan, layanan yang Anda minta, dan string penghentian ("aws4_request"). Nilai dinyatakan dalam format berikut: access_key/HHBBTTTT/wilayah/layanan/aws4_request.

Untuk informasi selengkapnya, lihat [Membuat permintaan AWS API yang ditandatangani](#) di Panduan Pengguna IAM.

Syarat: Tentukan parameter ini ketika Anda menyertakan informasi autentikasi dalam string kueri alih-alih di header otorisasi HTTP.

Tipe: string

Diperlukan: Kondisional

X-Amz-Date

Tanggal yang digunakan untuk membuat tanda tangan. Format harus berupa format dasar ISO 8601 (YYYYMMDD'T'HMMSS'Z'). Misalnya, waktu tanggal berikut adalah nilai X-Amz-Date yang valid: 20120325T120000Z.

Syarat: X-Amz-Date bersifat opsional untuk semua permintaan; nilai ini dapat digunakan untuk mengganti tanggal yang digunakan untuk menandatangani permintaan. Jika header Tanggal

ditentukan dalam format dasar ISO 8601, X-Amz-Date tidak diperlukan. Ketika X-Amz-Date digunakan, ia selalu mengganti nilai header Tanggal. Untuk informasi selengkapnya, lihat [Elemen tanda tangan permintaan AWS API](#) di Panduan Pengguna IAM.

Tipe: string

Diperlukan: Kondisional

X-Amz-Security-Token

Token keamanan sementara yang diperoleh melalui panggilan ke AWS Security Token Service (AWS STS). Untuk daftar layanan yang mendukung kredensi keamanan sementara AWS STS, lihat layanan [Layanan AWS yang berfungsi dengan IAM di Panduan Pengguna IAM](#).

Kondisi: Jika Anda menggunakan kredensi keamanan sementara dari AWS STS, Anda harus menyertakan token keamanan.

Tipe: string

Diperlukan: Kondisional

X-Amz-Signature

Menentukan tanda tangan yang dikodekan oleh hex yang dihitung dari string to sign dan kunci penandatanganan turunan.

Syarat: Tentukan parameter ini ketika Anda menyertakan informasi autentikasi dalam string kueri alih-alih di header otorisasi HTTP.

Tipe: string

Diperlukan: Kondisional

X-Amz-SignedHeaders

Menentukan semua header HTTP yang disertakan sebagai bagian dari permintaan kanonik. Untuk informasi selengkapnya tentang menentukan header yang ditandatangani, lihat [Membuat permintaan AWS API yang ditandatangani](#) di Panduan Pengguna IAM.

Syarat: Tentukan parameter ini ketika Anda menyertakan informasi autentikasi dalam string kueri alih-alih di header otorisasi HTTP.

Tipe: string

Diperlukan: Kondisional

Kesalahan Umum

Bagian ini berisi daftar kesalahan yang umum terjadi pada tindakan API dari semua layanan AWS. Untuk kesalahan khusus pada tindakan API untuk layanan ini, lihat topik untuk tindakan API tersebut.

AccessDeniedException

Anda tidak memiliki akses yang memadai untuk melakukan tindakan ini.

Kode Status HTTP: 400

IncompleteSignature

Tanda tangan permintaan tidak sesuai dengan standar AWS.

Kode Status HTTP: 400

InternalFailure

Pemrosesan permintaan telah gagal karena kesalahan yang tidak diketahui, pengecualian atau kegagalan.

Kode Status HTTP: 500

InvalidAction

Tindakan atau operasi yang diminta tidak valid. Verifikasi bahwa tindakan diketik dengan benar.

Kode Status HTTP: 400

InvalidClientTokenId

Sertifikat X.509 atau access key ID AWS yang diberikan tidak ada dalam catatan kami.

Kode Status HTTP: 403

NotAuthorized

Anda tidak memiliki izin untuk melakukan tindakan ini.

Kode Status HTTP: 400

OptInRequired

Access key ID AWS membutuhkan berlangganan untuk layanan.

Kode Status HTTP: 403

RequestExpired

Permintaan menjangkau layanan lebih dari 15 menit setelah stempel tanggal pada permintaan atau lebih dari 15 menit setelah tanggal kedaluwarsa permintaan (seperti untuk URL pre-signed), atau stempel tanggal pada permintaan lebih dari 15 menit di masa mendatang.

Kode Status HTTP: 400

ServiceUnavailable

Permintaan telah gagal karena kegagalan sementara server.

Kode Status HTTP: 503

ThrottlingException

Permintaan ditolak karena throttling permintaan.

Kode Status HTTP: 400

ValidationError

Input gagal untuk memenuhi batasan yang ditentukan oleh layanan AWS.

Kode Status HTTP: 400

Riwayat dokumen untuk AWS Transfer Family

Tabel berikut menjelaskan dokumentasi untuk rilis ini AWS Transfer Family.

- APIversi: transfer-2018-11-05
- Pembaruan dokumentasi terbaru: 16 Januari 2024

Perubahan	Deskripsi	Tanggal
Integrasikan dengan Amazon EventBridge	AWS Transfer Family sekarang secara otomatis menerbitkan acara ke Amazon EventBridge untuk semua operasi transfer file. Untuk detailnya, lihat Mengelola Transfer Family acara menggunakan Amazon EventBridge .	Februari 8, 2024
Penambahan kebijakan keamanan baru	AWS Transfer Family telah menambahkan kebijakan baru FIPS dan FIPS non-keamanan. Selain itu, kebijakan keamanan default yang ditetapkan ke server selalu merupakan kebijakan keamanan terbaru. Untuk detailnya, lihat Kebijakan keamanan untuk AWS Transfer Family .	Februari 5, 2024
Support untuk alamat IP statis untuk SFTP konektor dan AS2	Transfer Family sekarang menyediakan alamat IP statis untuk SFTP konektor dan AS2. Ini memungkinkan koneksi dengan SFTP server	Januari 16, 2024

Perubahan	Deskripsi	Tanggal
	jarak jauh yang diamankan oleh kontrol IP allowlisting. Untuk AS2, kami memperkenalkan alamat IP statis untuk MDN respons asinkron dari server. AS2	
Panduan pengguna telah diatur ulang untuk menyelaraskan lebih dekat dengan versi terbaru. AWS Transfer Family	Transfer Family telah menambahkan beberapa fitur sejak panduan ini berasal, yang mengharuskan restrukturisasi panduan.	Januari 3, 2024
Penyempurnaan pemetaan direktori logis Pengoptimalan kinerja daftar Amazon S3	Transfer Family sekarang mendukung pemetaan direktori logis hingga 2,1 MB. Anda juga sekarang dapat mendeklarasikan apakah pemetaan pengguna ke file. Untuk informasi selengkapnya, lihat Aturan untuk menggunakan direktori logis. Saat membuat atau memperbarui server yang menggunakan Amazon S3 untuk penyimpanan, Anda sekarang dapat mengoptimalkan kinerja daftar direktori (atau folder) S3 Anda. Untuk informasi selengkapnya, lihat Mengkonfigurasi titik SFTP akhir, FTPS, atau FTP server.	17 November 2023

Perubahan	Deskripsi	Tanggal
Port alternatif untuk SFTP server dengan titik akhir virtual private cloud (VPC)	Sekarang Anda dapat mengaktifkan port non-standar alternatif untuk server SFTP Transfer Family Anda yang memiliki titik VPC akhir. Untuk informasi selengkapnya, lihat Buat server di cloud pribadi virtual .	17 November 2023
Support untuk SFTP konektor	SFTPKonektor memperluas kemampuan AWS Transfer Family untuk berkomunikasi dengan server jarak jauh baik di cloud maupun di tempat. Untuk informasi selengkapnya, lihat Mengirim dan mengambil file dengan menggunakan konektor SFTP .	25 Juli 2023
Support untuk otentikasi AS2 Dasar	Transfer Family sekarang mendukung penggunaan otentikasi Dasar untuk server yang menggunakan protokol Applicability Statement 2 (AS2). Untuk informasi selengkapnya, lihat Otentikasi dasar untuk konektor AS2 .	Juni 30, 2023

Perubahan	Deskripsi	Tanggal
Support untuk JSON logging terstruktur	Transfer Family sekarang mendukung pengiriman JSON log terstruktur ke Amazon CloudWatch, mengelompokkan steam log ke grup log kustom, dan melakukan kueri log umum di seluruh protokol. Untuk informasi selengkapnya, lihat CloudWatch Pencatatan Amazon untuk AWS Transfer Family .	Juni 24, 2023
Support untuk beberapa metode otentikasi	Transfer Family memiliki dukungan untuk otentikasi dengan menggunakan password, public/private key pair, atau keduanya. Ini tersedia untuk server yang menggunakan SFTP protokol dan penyedia identitas khusus. Untuk informasi selengkapnya, lihat Buat server SFTP yang diaktifkan .	17 Mei 2023

Perubahan	Deskripsi	Tanggal
Dukungan untuk dekripsi Pretty Good Privacy (PGP) dengan file yang diproses Transfer Family dengan alur kerja	Transfer Family memiliki dukungan bawaan untuk dekripsi Pretty Good Privacy (PGP). Anda dapat menggunakan PGP dekripsi pada file yang diunggah melalui SFTP/FTPS, atau ke Amazon Simple Storage Service (FTP/Amazon S3) Simple Storage Service (Amazon) atau Amazon Elastic File System (Amazon). EFS Untuk informasi selengkapnya, silakan lihat Hasilkan dan kelola PGP kunci dan Gunakan PGP dekripsi dalam alur kerja Anda .	21 Desember 2022
Dukungan yang dikelola sepenuhnya untuk protokol transfer file Applicability Statement 2 (AS2) dengan server Transfer Family	Anda dapat membuat server yang menggunakan AS2 protokol untuk mengirim dan menerima informasi ke dan dari mitra dagang yang berada di dalam atau di luar AWS lingkungan. Untuk informasi selengkapnya, lihat Mengkonfigurasi AS2 .	25 Juli 2022

Perubahan	Deskripsi	Tanggal
Support untuk display banner saat membuat server	Anda dapat menambahkan pesan yang disesuaikan saat membuat server. Anda dapat menampilkan pesan pra-otentikasi (semua protokol), dan pesan pasca-otentikasi (untuk dan server). FTP FTPS Untuk informasi selengkapnya, lihat Buat server SFTP yang diaktifkan , Buat server FTPS yang diaktifkan , atau Buat server FTP yang diaktifkan .	Februari 17, 2022
Support untuk AWS Lambda sebagai penyedia identitas	Anda sekarang dapat terhubung ke penyedia identitas khusus menggunakan AWS Lambda server Transfer Family mereka. Sebelumnya, Anda harus menyediakan Amazon API Gateway URL untuk mengintegrasikan penyedia identitas kustom. Untuk informasi selengkapnya, lihat Menggunakan AWS Lambda untuk mengintegrasikan penyedia identitas Anda .	November 16, 2021

Perubahan	Deskripsi	Tanggal
Support untuk Alur Kerja Transfer File Terkelola	Alur Kerja Transfer File Terkelola memberi Anda abstraksi pemrosesan pasca-unggah untuk tugas umum yang saat ini Anda lakukan secara manual. Untuk informasi selengkapnya, lihat AWS Transfer Family alur kerja terkelola .	2 September 2021
Support untuk AWS Directory Service for Microsoft Active Directory	Selain penyedia identitas terkelola layanan dan kustom, Anda sekarang dapat menggunakan AWS Directory Service for Microsoft Active Directory untuk mengelola akses pengguna untuk otentikasi dan otorisasi. Untuk informasi selengkapnya, lihat Menggunakan penyedia identitas AWS Directory Service .	24 Mei 2021
Baru Wilayah AWS	AWS Transfer Family sekarang tersedia di Wilayah Afrika (Cape Town). Untuk informasi selengkapnya tentang titik akhir Transfer Family, lihat AWS Transfer Family titik akhir dan kuota di Referensi Umum AWS	24 Februari 2021

Perubahan	Deskripsi	Tanggal
Baru Wilayah AWS	AWS Transfer Family sekarang tersedia di Wilayah Asia Pasifik (Hong Kong) dan Timur Tengah (Bahrain). Untuk informasi selengkapnya tentang titik akhir Transfer Family, lihat AWS Transfer Family titik akhir dan kuota di Referensi Umum AWS	17 Februari 2021
Support untuk Amazon EFS sebagai penyimpanan data	Transfer Family sekarang mendukung transfer file masuk dan keluar dari Amazon Elastic File System (AmazonEFS). Amazon EFS adalah sistem NFS file elastis yang sederhana, dapat diskalakan, dan dikelola sepenuhnya. Untuk informasi selengkapnya, lihat Konfigurasi sistem EFS file Amazon .	Januari 06, 2021
Support untuk AWS WAF	Transfer Family sekarang mendukung AWS WAF, firewall aplikasi web yang membantu melindungi aplikasi web dan API operasi dari serangan. Untuk informasi selengkapnya, lihat Tambahkan firewall aplikasi web .	24 November 2020

Perubahan	Deskripsi	Tanggal
Support untuk beberapa grup keamanan di virtual private cloud (VPC)	Anda sekarang dapat melampirkan beberapa grup keamanan ke server di fileVPC. Untuk informasi selengkapnya, lihat Buat server di cloud pribadi virtual .	15 Oktober 2020
Baru Wilayah AWS	Transfer Family sekarang tersedia di AWS GovCloud (US) Wilayah. Untuk informasi selengkapnya tentang titik akhir Transfer Family untuk AWS GovCloud (US) Wilayah, lihat AWS Transfer Family titik akhir dan kuota di. Referensi Umum AWS Untuk informasi tentang penggunaan Transfer Family di AWS GovCloud (US) Wilayah, lihat AWS Transfer Family di Panduan AWS GovCloud (US) Pengguna.	30 September 2020
Kebijakan keamanan dengan algoritma kriptografi yang didukung sekarang dapat dilampirkan ke server Anda	Anda sekarang dapat melampirkan kebijakan keamanan yang berisi serangkaian algoritma kriptografi yang didukung ke server Anda. Untuk informasi selengkapnya, lihat Kebijakan keamanan untuk AWS Transfer Family .	12 Agustus 2020

Perubahan	Deskripsi	Tanggal
Dukungan untuk titik akhir Standar Pemrosesan Informasi Federal (FIPS)	FIPSEndpoint yang diaktifkan sekarang tersedia di Amerika Utara. Wilayah AWS Untuk Wilayah yang tersedia, lihat AWS Transfer Family titik akhir dan kuota di. Referensi Umum AWS FIPS Untuk mengaktifkan titik akhir SFTP server yang diaktifkan, lihat. Buat server SFTP yang diaktifkan FIPS Untuk mengaktifkan titik akhir FTPS server yang diaktifkan, lihat. Buat server FTPS yang diaktifkan FIPS Untuk mengaktifkan titik akhir FTP server yang diaktifkan, lihat. Buat server FTP yang diaktifkan	12 Agustus 2020
Peningkatan panjang karakter nama pengguna dan karakter tambahan yang diizinkan	Nama pengguna sekarang dapat berisi tanda (@) dan periode (.), dan dapat memiliki panjang maksimum 100 karakter. Untuk menambahkan pengguna, lihat Mengelola pengguna untuk titik akhir server .	12 Agustus 2020

Perubahan	Deskripsi	Tanggal
Support untuk pembuatan peran Amazon CloudWatch logging AWS Identity and Access Management (IAM) otomatis	Transfer Family sekarang mendukung pembuatan otomatis IAM peran CloudWatch logging untuk melihat aktivitas pengguna akhir. Untuk informasi selengkapnya, lihat Buat server SFTP yang diaktifkan , Buat server FTPS yang diaktifkan , atau Buat server FTP yang diaktifkan .	30 Juli 2020
AWS Transfer Family sekarang mendukung IP Sumber sebagai faktor otorisasi.	Transfer Family menambahkan dukungan untuk menggunakan alamat IP sumber pengguna akhir sebagai faktor otorisasi, memungkinkan Anda untuk menerapkan lapisan keamanan tambahan saat mengotorisasi akses melalui Secure File Transfer Protocol (SFTP), File Transfer Protocol over (), atau File Transfer Protocol SSL (FTPS). FTP Untuk informasi selengkapnya, lihat Bekerja dengan penyedia identitas khusus .	9 Juni 2020

Perubahan	Deskripsi	Tanggal
AWS Transfer SFTP untuk sekarang AWS Transfer Family dan menambahkan dukungan untuk FTP dan FTPS.	Anda sekarang dapat menggunakan dua protokol tambahan untuk transfer file pengguna Anda: File Transfer Protocol Secure (FTPS) dan File Transfer Protocol (). FTP Pengguna dapat memindahkan, menjalankan, mengamankan, dan mengintegrasikan alur kerja FTP berbasis FTP over SSL (FTPS) dan plaintext AWS, selain dukungan Secure File Transfer Protocol () SFTP yang ada.	23 April 2020
Support untuk grup keamanan virtual private cloud (VPC) dan alamat IP Elastis	Anda sekarang dapat membuat daftar yang diizinkan untuk alamat IP yang masuk menggunakan grup keamanan, memberikan lapisan keamanan tambahan untuk server. Anda juga dapat mengaitkan alamat IP Elastis dengan titik akhir server Anda. Dengan melakukan ini, Anda dapat mengaktifkan pengguna di belakang firewall untuk mengizinkan akses ke titik akhir tersebut. Untuk informasi selengkapnya, lihat Buat server di cloud pribadi virtual .	10 Januari 2020

Perubahan	Deskripsi	Tanggal
Support untuk bekerja di VPC	Anda sekarang dapat membuat server di fileVPC. Anda dapat menggunakan server Anda untuk mentransfer data melalui klien Anda ke dan dari bucket Amazon S3 tanpa melalui internet publik. Untuk informasi selengkapnya, lihat Buat server di cloud pribadi virtual .	27 Maret 2019
Versi pertama AWS Transfer Family dirilis.	Rilis awal ini mencakup pengaturan petunjuk arah, menjelaskan cara memulai, dan memberikan informasi tentang konfigurasi klien, konfigurasi pengguna, dan aktivitas pemantauan.	November 25, 2018

Daftar istilah AWS

Untuk terminologi AWS terbaru, lihat [Daftar istilah AWS](#) di Referensi Glosarium AWS.

Terjemahan disediakan oleh mesin penerjemah. Jika konten terjemahan yang diberikan bertentangan dengan versi bahasa Inggris aslinya, utamakan versi bahasa Inggris.