



Add a permission의

AWS Transfer Family



AWS Transfer Family: Add a permission의

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

Amazon의 상표 및 트레이드 드레스는 Amazon 외 제품 또는 서비스와 함께, Amazon 브랜드 이미지를 떨어뜨리거나 고객에게 혼동을 일으킬 수 있는 방식으로 사용할 수 없습니다. Amazon이 소유하지 않은 기타 모든 상표는 Amazon과 제휴 관계이거나 관련이 있거나 후원 관계와 관계없이 해당 소유자의 자산입니다.

Table of Contents

AWS Transfer Family란 무엇인가요?	1
AWS Transfer Family 작동 방식	4
Transfer Family와 관련된 블로그 게시물	5
사전 조건	8
리전, 엔드포인트 및 할당량	8
에 가입 AWS	8
스토리지 구성	9
Amazon S3 버킷 구성	10
Amazon EFS 파일 시스템 구성	14
IAM 역할 및 정책 생성	18
사용자 역할 생성	19
세션 정책의 작동 방식	22
읽기/쓰기 액세스 정책의 예	25
Transfer Family 자습서	29
서버 엔드포인트 시작하기	30
사전 조건	30
1단계: AWS Transfer Family 콘솔에 로그인	31
2단계: SFTP 지원 서버 생성	31
3단계: 서비스 관리 사용자 추가	32
4단계: 클라이언트를 사용하여 파일 전송	33
복호화 워크플로 생성	35
1단계: 실행 역할 구성	36
2단계: 관리형 워크플로 생성	37
3단계: 서버에 워크플로 추가 및 사용자 생성	38
4단계: PGP 키 쌍 생성	39
5단계: PGP 프라이빗 키 저장 AWS Secrets Manager	40
6단계: 파일 암호화	41
7단계: 워크플로 실행 및 결과 보기	42
SFTP 커넥터 생성 및 사용	43
커넥터 송신 유형	44
1단계: 필요한 지원 리소스 생성	45
2단계: SFTP 커넥터 생성 및 테스트	51
3단계: SFTP 커넥터를 사용하여 파일 전송 및 검색	56
원격 SFTP 서버로 사용할 Transfer Family 서버를 생성하는 절차	60

사용자 지정 자격 증명 공급자 사용	63
사전 조건	63
1단계: CloudFormation 스택 생성	64
2단계: 서버의 API Gateway 메서드 구성을 확인합니다.	65
3단계: Transfer Family 서버 세부 정보 보기	66
4단계: 사용자가 서버에 연결할 수 있는지 테스트	67
5단계: SFTP 연결 및 파일 전송 테스트	67
6단계: 버킷에 대한 액세스 제한	68
Amazon EFS를 사용하는 경우 Lambda 업데이트	70
AS2 구성 설정	71
1단계: AS2용 인증서 생성	73
2단계: 인증서를 Transfer Family 인증서 리소스로 가져오기	76
3단계: 사용자와 거래 파트너를 위한 프로필 생성	77
4단계: AS2 프로토콜을 사용하는 Transfer Family 서버 생성	78
5단계: 사용자와 사용자의 파트너 간의 계약서 작성	81
6단계: 사용자와 사용자의 거래 파트너 간의 커넥터를 생성합니다.	83
7단계: Transfer Family를 사용하여 AS2를 통한 파일 교환 테스트	84
기본 웹 앱 설정	86
사전 조건	86
1단계: 필요한 지원 리소스 생성	86
2단계: Transfer Family 웹 앱 생성	87
3단계: 버킷에 대한 교차 오리진 리소스 공유(CORS) 구성	88
4단계: Transfer Family 웹 앱에 사용자 추가	89
5단계: Amazon S3에 위치 등록 및 액세스 권한 생성	90
6단계: 사용자로 Transfer Family 웹 앱에 액세스	92
다음 단계	94
Okta를 웹 앱의 자격 증명 공급자로 통합	94
선택적 다중 버킷 액세스를 웹 앱 설정	97
사전 조건	97
1단계: Transfer Family 웹 앱 생성	97
2단계: S3 액세스를 위한 IAM 역할 구성	98
3단계: S3 Access Grants 설정	100
4단계: S3 버킷 위치 등록	101
5단계: 액세스 권한 부여 생성	101
6단계: S3 버킷에 대한 CORS 정책 구성	102
7단계: 구성 테스트	103

결론	103
SFTP, FTPS, FTP용 Transfer Family	104
자격 증명 공급자 옵션	104
AWS Transfer Family 엔드포인트 유형 매트릭스	106
Transfer Family 서버 엔드포인트 구성	110
SFTP 지원 서버 생성	112
FTPS 지원 서버 생성	119
FTP 지원 서버 생성	127
VPC에 서버 생성	135
사용자 지정 호스트 이름으로 작업	159
FTP/FTPS NLB 고려 사항	163
서버 엔드포인트를 통해 파일 전송	163
가용 SFTP/FTPS/FTP 명령	166
Amazon VPC 엔드포인트 찾기	168
setstat 오류 방지	169
OpenSSH 사용	34
WinSCP 사용	171
Cyberduck 사용하기	33
FileZilla 사용	174
Perl 클라이언트 사용	175
LFTP 사용	176
업로드 후 처리	176
SFTP 메시지	177
사용자 관리	179
아마존 EFS와 아마존 S3	180
논리적 디렉터리	180
Active Directory 그룹 할당량	182
서비스 관리형 사용자	182
사용자 지정 ID 제공업체	192
MS AD용 디렉터리 서비스	226
Entra ID용 디렉터리 서비스	237
논리적 디렉터리 사용	244
chroot 및 디렉터리 구조 이해	245
논리적 디렉터리 사용 규칙	245
논리적 디렉터리 구현	247
논리적 디렉터리 구성 예제	250

Amazon EFS를 위한 논리적 디렉터리 구성	251
사용자 지정 AWS Lambda 응답	252
FSx for NetApp ONTAP 파일 시스템에 액세스	253
개요	254
사전 조건	254
FSx 스토리지가 Transfer Family와 작동하는 방식	255
FSx용 S3 액세스 포인트 생성	256
FSx에서 S3 액세스 포인트 별칭 사용	258
FSx 스토리지를 Transfer Family 구성	260
FSx 스토리지 사용자 관리	262
파일 전송 클라이언트 구성	263
FSx 스토리지 문제 해결	265
Transfer Family 웹 앱	266
AWS 리전 Transfer Family 웹 앱용	266
AWS Transfer Family 웹 앱의 브라우저 호환성	267
Transfer Family 웹 앱을 생성하는 방법	267
자격 증명 공급자 구성	269
IAM 역할 구성	271
Transfer Family 웹 앱 구성	274
Transfer Family 웹 앱 생성	274
IP 주소 지정	276
VPC에서 Transfer Family 웹 앱 생성	277
Transfer Family 웹 앱 생성	277
생성 후 단계	279
CORS(Cross-Origin Resource Sharing) 정책	279
특정 VPC 엔드포인트에 대한 액세스 제한	281
Transfer Family 웹 앱에 사용자 또는 그룹 할당 또는 추가	282
버킷에 대한 교차 오리진 리소스 공유(CORS) 설정	285
Amazon S3 Access Grants 구성	287
사용자 지정 URL 사용	292
Transfer Family 웹 앱에 대한 로깅	295
Amazon S3 데이터 이벤트 활성화	295
인증 로그 예제	298
로그 항목 보기	305
웹 앱 문제 해결	306
네트워크 오류 문제 해결	306

구성된 버킷이 나타나지 않는 문제 해결	307
사용자 지정 URL 오류 문제 해결	307
기타 오류 문제 해결	308
웹 앱에 표시되는 중복 S3 버킷	308
최종 사용자 지침	309
웹 앱 할당량	309
IAM Identity Center 사용자	309
타사 최종 사용자	311
Transfer Family 최종 사용자 인터페이스	311
사용 가능한 작업	312
SFTP 커넥터	315
SFTP 커넥터 생성	315
SFTP 커넥터 송신 유형 선택	316
IP 주소 지정 모드 선택	317
자격 증명을 Secrets Manager에 저장	318
서비스 관리형 송신을 사용하여 SFTP 커넥터 생성	319
VPC 기반 송신을 사용하여 SFTP 커넥터 생성	329
SFTP 커넥터를 테스트합니다.	340
VPC 연결	341
SFTP 커넥터 사용	342
파일 전송	343
원격 디렉터리의 콘텐츠 나열	345
원격 서버에서 파일 이동 및 삭제	346
SFTP 커넥터 모니터링	348
커넥터 API를 사용하여 파일 전송 요청 상태 쿼리	348
Amazon EventBridge에서 SFTP 커넥터 이벤트 보기	349
Amazon CloudWatch에서 SFTP 커넥터 로그 보기	349
VPC 송신 유형 커넥터 모니터링	349
SFTP 커넥터 관리	351
SFTP 커넥터 업데이트	352
SFTP 커넥터 세부 정보 보기	352
Transfer for SFTP 커넥터 할당량	353
Transfer for SFTP 커넥터 할당량	354
SFTP 커넥터 크기 조정	355
참조 아키텍처	356
블로그 게시물	356

워크숍	356
Solutions	356
VPC 참조 아키텍처	357
AS2용 Transfer Family	359
AS2 사용 사례	360
AS2 CloudFormation 템플릿	364
AS2 템플릿 사용자 지정	364
AS2 배포 테스트	364
AS2 템플릿 배포 모범 사례	365
AS2 구성	366
AS2 구성	367
AS2 할당량	368
AS2 특성 및 기능	371
AS2 인증서 관리	373
AS2 인증서 가져오기	373
AS2 인증서 교체	375
AS2 프로필 생성	377
AS2 서버 생성	378
Transfer Family 콘솔을 사용하여 AS2 서버 생성	378
템플릿을 사용하여 AS2 서버 생성	381
AS2 계약 생성	384
AS2 커넥터 구성	386
AS2 커넥터 생성	386
AS2 커넥터 알고리즘	389
AS2 커넥터의 기본 인증	390
AS2 커넥터에 대한 기본 인증을 활성화합니다.	391
커넥터 세부 정보 보기	394
AS2 메시지 전송	397
AS2 메시지 수신	398
AS2용 HTTPS 구성	399
AS2 커넥터를 사용하여 파일 전송	404
파일 이름 및 위치	406
상태 코드	409
샘플 JSON 파일	409
AS2용 사용자 지정 HTTP 헤더	412
템플릿 개요	412

작동 방식	412
주요 기능	413
구현 세부 정보	413
배포 및 사용	414
AS2 모니터링	415
AS2 상태 코드	416
AS2 오류 코드	417
인증서 만료 모니터링	428
DaysUntilExpiry 지표	428
인증서 모니터링 모범 사례	429
경보 구성 예	429
파일 처리 워크플로 관리	431
워크플로 만들기	433
워크플로 구성 및 실행	434
워크플로 세부 정보 보기	436
사전 정의된 단계 사용	439
파일 복사	439
파일 복호화	444
파일 태그 지정	450
파일 삭제	451
워크플로에 명명된 변수	452
태그 및 삭제 워크플로 예시	452
사용자 지정 파일 처리 단계 사용	457
여러 Lambda 함수를 연속적으로 사용	458
사용자 정의 처리 후 파일에 액세스	459
파일 업로드 AWS Lambda 시로 전송된 이벤트 예	460
사용자 지정 워크플로 단계를 위한 Lambda 함수 예	461
사용자 지정 단계 IAM 권한	462
워크플로에 대한 IAM 정책	462
워크플로 신뢰 관계	464
실행 역할 예시: 복호화, 복사, 태그 지정	465
실행 역할 예시: 함수 실행 및 삭제	467
워크플로의 예외 처리	467
워크플로 실행 모니터링	468
워크플로를 위한 CloudWatch 로깅	468
워크플로를 위한 CloudWatch 지표	471

템플릿에서 워크플로 생성	471
Transfer Family 서버에서 워크플로 제거	475
제한 및 제약 조건	476
서버 관리	478
서버 목록 보기	478
서버 삭제	478
SFTP 서버 세부 정보 보기	480
AS2 서버 세부 정보 보기	481
IPv6 지원	483
IPv6 제한 사항	483
서버용 IPv6 구성	484
듀얼 스택 AS2 퍼블릭 서버에 ALB 사용	484
서버 세부 정보 편집	486
File Transfer 프로토콜 편집	489
서버 엔드포인트 편집	491
로깅 편집	492
보안 정책 편집	493
관리형 워크플로 변경	494
서버의 디스플레이 배너 변경	495
서버를 온라인이나 오프라인으로 전환	495
자격 증명 공급자 구성 편집	496
서비스 관리형 자격 증명 공급자로 변경	497
AWS 디렉터리 서비스로 변경	497
사용자 지정 자격 증명 공급자로 변경	498
자격 증명 공급자 전환 중 사용자 보존	503
자격 증명 공급자 변경 시 중요 고려 사항	503
서버 호스트 키 관리	503
호스트 키를 가져와야 하는 경우	504
추가 서버 호스트 키 추가	505
서버 호스트 키 삭제	507
서버 호스트 키 교체	507
추가 서버 호스트 키 정보	509
콘솔 내 사용량 모니터링	509
액세스 통제 관리	515
S3 버킷 액세스 정책 생성	515
세션 정책 생성	517

세션 정책 예제	518
세션 정책에 대한 중첩 대체	520
동적 권한 관리	521
Transfer Family 권한 아키텍처 이해	521
권한 관리에 대한 두 가지 접근 방식	521
세션 정책 구현	522
사용자 유형별 구현	523
예: 세션 정책을 사용하여 역할 관리 간소화	523
CloudTrail 로깅	526
CloudTrail 로깅 활성화	527
Amazon S3 데이터 이벤트 활성화	528
서버 생성을 위한 로그 항목의 예	528
데이터 액세스 로그	530
성공적인 데이터 액세스를 위한 로그 항목의 예	530
일반적인 데이터 액세스 작업	532
CloudWatch 로깅	534
Transfer Family에 대한 CloudWatch 로깅 유형	534
서버에 대한 로깅 생성	536
서버에 대한 로깅 생성	539
서버의 로깅 업데이트	539
서버 구성 보기	542
워크플로에 대한 로깅 관리	544
CloudWatch에 대한 역할 구성	547
Transfer Family 로그 스트림 보기	549
Amazon CloudWatch 경보 생성	552
S3 API 호출을 S3 액세스 로그에 로깅	553
혼동된 대리자 문제를 제한하는 예	553
Transfer Family의 CloudWatch 로그 구조	555
Transfer Family에 대한 JSON 구조화된 로그	555
Transfer Family에 대한 레거시 로그	558
CloudWatch 로그 항목 예	561
전송 세션 로그 항목의 예	562
SFTP 커넥터의 로그 항목 예	563
VPC Lattice 커넥터의 로그 항목 예제	565
키 교환 알고리즘 실패에 대한 로그 항목 예제	566
CloudWatch 지표 사용	567

Transfer Family 크기	571
User Notifications	572
CloudWatch 쿼리	572
를 사용하여 이벤트 관리 EventBridge	575
Transfer Family 이벤트	576
SFTP, FTPS 및 FTP 서버 이벤트	576
SFTP 커넥터 이벤트	578
AS2 이벤트	579
Transfer Family 이벤트 전송	580
이벤트 패턴 생성	581
이벤트에 대한 Transfer Family 이벤트 패턴 테스트	583
권한	583
추가 리소스	583
이벤트 세부 정보 참조	583
서버 이벤트	584
커넥터 이벤트	591
AS2 이벤트	599
보안	605
VPC 연결 보안 이점	606
VPC Lattice 보안 모델	606
VPC 연결에 대한 보안 모범 사례	607
서버에 대한 보안 정책	607
암호화 알고리즘	609
보안 정책 세부 정보	620
SFTP 커넥터에 대한 보안 정책	638
암호화 알고리즘	638
SFTP 커넥터 보안 정책 세부 정보	639
포스트 쿼터 보안 정책	641
SSH의 포스트 쿼터 하이브리드 키 교환 소개	642
사용 방법	642
테스트 방법	643
데이터 보호	646
Transfer Family의 데이터 암호화	647
저장 시 암호화	648
키 관리	649
알고리즘 개요	650

SSH 인증	650
PGP 알고리즘	651
SSH 키 생성	653
SSH 키 교체	658
PGP 키 생성	660
PGP 키 관리	661
지원되는 PGP 클라이언트	666
ID 및 액세스 관리	667
대상	667
ID를 통한 인증	668
정책을 사용하여 액세스 관리	669
AWS Transfer Family 에서 IAM을 사용하는 방법	670
자격 증명 기반 정책 예	674
태그 기반 정책 예제	677
ID 및 액세스 문제 해결	680
IAM 조건 키	682
규정 준수 확인	684
복원력	685
APIs용 VPC 엔드포인트	685
VPC 엔드포인트 정책을 사용하여 액세스 제어	686
VPC 엔드포인트 생성	687
인프라 보안	687
NLB 및 NAT 고려 사항	688
VPC 연결 인프라 보안	689
웹 애플리케이션 방화벽	690
교차 서비스 혼동된 대리자 방지	691
Transfer Family 사용자 역할 전송	692
Transfer Family 워크폴로 역할 전송	694
Transfer Family 커넥터 역할	695
Transfer Family 로깅/호출 역할 이전	697
AWS 관리형 정책	698
AWSTransferConsoleFullAccess	699
AWSTransferFullAccess	699
AWSTransferLoggingAccess	699
AWSTransferReadOnlyAccess	699
정책 업데이트	699

Transfer Family Terraform 모듈	701
SFTP 서버	701
SFTP 커넥터	701
AS2	702
B2B Data Interchange	702
Transfer Family 문제 해결	703
인증 문제	703
인증 실패—SSH/SFTP	704
관리형 AD 불일치 영역 문제	704
Active Directory 그룹 제한 초과	705
기타 인증 문제	705
Amazon API Gateway 문제 해결	706
ID 제공자 테스트 문제 해결	708
웹 앱에서 Amazon S3 버킷 복제	308
SFTP 커넥터 문제	709
SFTP 커넥터의 신뢰할 수 있는 호스트 키 추가 문제 해결	709
키 협상 실패	710
SFTP 커넥터 제한	710
SFTP 커넥터 성능 최적화	711
VPC 연결 문제 해결	712
기타 SFTP 커넥터 문제	714
SFTP 연결 문제	715
SFTP 연결 문제 해결	715
SFTP 클라이언트 문제 해결	716
파일 업로드 문제 해결	716
VPC 송신 유형 SFTP 커넥터 문제 해결	717
사용자 지정 자격 증명 공급자 문제	719
API Gateway 통합 오류 문제 해결	719
Lambda 함수 제한 시간 문제 해결	721
일관된 Lambda 제한 시간 문제 해결	722
KeyError 예외 문제 해결	722
워크플로 문제	722
관리형 워크플로 문제 해결	723
워크플로 복호화 문제 해결	725
EFS 문제	729
Amazon SES 문제 해결	729

스토리지 및 암호화 문제	731
암호화된 Amazon S3 버킷에 대한 정책 문제 해결	732
ResourceNotFound 예외 문제 해결	732
문제 모니터링 및 알림	733
누락되거나 불완전한 CloudWatch 지표 문제 해결	733
누락된 EventBridge 이벤트 문제 해결	734
리전 간 전송 문제	735
리전 간 전송 권한 문제 해결	735
리전 간 전송 성능 문제 해결	737
Terraform 배포 문제	738
Terraform 리소스 생성 실패 문제 해결	738
Terraform 상태 관리 문제 해결	739
WAF 통합 문제	740
WAF 차단 합법적인 트래픽 문제 해결	740
사용자 지정 자격 증명 공급자와의 WAF 통합 문제 해결	741
서비스 관리형 사용자 문제	741
서비스 관리 사용자 문제 해결	742
AS2 문제	743
AS2 문제 해결	743
AS2 인증서 문제	743
AS2 MDN 수신 문제	744
인증서 만료 모니터링 문제	744
API 참조	746
API 요청 만들기	747
Transfer Family 필수 요청 헤더	748
Transfer Family 요청 입력 및 서명	748
오류 응답	748
사용 가능한 라이브러리	751
자격 증명 공급자	751
규칙	752
ID 형식	752
ARN 형식	753
DNS 및 엔드포인트	754
API 엔드포인트	754
서버 엔드포인트	754
문서 이력	756

..... dcclxix

AWS Transfer Family란 무엇인가요?

AWS Transfer Family 는 AWS 스토리지 서비스 내부 및 외부로 파일을 전송할 수 있는 보안 전송 서비스입니다. Transfer Family는 AWS 클라우드 플랫폼의 일부입니다.는 SFTP, AS2, FTPS, FTP 및 웹 브라우저 기반 전송을 통해 스토리지 서비스로 직접 또는 AWS 스토리지 서비스에서 파일을 전송할 수 있도록 완전 관리형 지원을 AWS Transfer Family 제공합니다. 인증, 액세스 및 방화벽에 대한 기존 클라이언트 측 구성을 유지하여 파일 전송 워크플로를 원활하게 마이그레이션, 자동화 및 모니터링할 수 있으므로 고객, 파트너 및 내부 팀 또는 해당 애플리케이션에는 변경 사항이 없습니다. 자세한 내용을 알아보고 Amazon Web Services로 클라우드 애플리케이션 구축을 시작하려면 [시작하기 AWS](#)를 참조하세요.

AWS Transfer Family 는 다음 AWS 스토리지 서비스에서 또는 다음 스토리지 서비스로의 데이터 전송을 지원합니다.

- Amazon Simple Storage Service(S3) 스토리지 Amazon S3에 관한 자세한 내용은 [Amazon Simple Storage Service 시작하기](#)를 참조하세요.
- Amazon Elastic File System(Amazon EFS) Network File System(NFS) 파일 시스템 Amazon EFS에 관한 자세한 내용은 [Amazon Elastic File System이란?](#)을 참조하세요.

AWS Transfer Family 는 다음 프로토콜을 통한 데이터 전송을 지원합니다.

- SFTP(Secure File Transfer Protocol): 버전 3

공식 IETF 문서는 [SSH 파일 전송 프로토콜 draft-ietf-secsh-filexfer-02.txt](#)입니다.

- File Transfer 프로토콜 보안(FTPS)
- File Transfer 프로토콜(FTP)
- 적용성 보고서 2(AS2)
- 브라우저 기반 전송

Note

FTP 및 FTPS 데이터 연결의 경우 Transfer Family가 데이터 채널을 설정하는 데 사용하는 포트 범위는 8192~8200입니다.

File Transfer 프로토콜은 금융 서비스, 의료, 광고, 소매를 비롯한 다양한 산업의 데이터 교환 워크플로에서 활용됩니다. Transfer Family는 파일 전송 워크플로의 마이그레이션을 간소화합니다 AWS.

다음은 Amazon S3에서 Transfer Family를 사용하는 몇 가지 일반적인 사용 사례입니다.

- 공급업체 및 파트너와 같은 타사의 업로드를 AWS 위한의 데이터 레이크입니다.
- 고객과의 구독 기반 데이터 배포.
- 조직 내의 내부 전송.

다음은 Amazon EFS에서 Transfer Family를 사용하는 몇 가지 일반적인 사용 사례입니다.

- 데이터 분산
- 공급망
- 콘텐츠 관리
- 웹 서비스 애플리케이션

다음은 AS2에서 Transfer Family를 사용하는 몇 가지 일반적인 사용 사례입니다.

- 프로토콜에 내장된 데이터 보호 및 보안 기능을 필요로 하는 컴플라이언스 요구 사항이 있는 워크플로
- 공급망 물류
- 결제 워크플로
- 기업 간 거래(B2B)
- 전사적 자원 관리(ERP) 및 고객 관계 관리(CRM) 시스템과의 통합

다음은 Transfer Family 웹 앱을 사용하는 몇 가지 일반적인 사용 사례입니다.

- 더 광범위하고 다양한 비즈니스 사용자에게 Amazon S3의 데이터에 대한 액세스 간소화
- 작업 인력을 위한 중앙 집중식 데이터 액세스 관리
- 관리형 인터페이스를 통한 Amazon S3 Access Grants 시각화

Transfer Family를 사용하면 서버 인프라를 실행할 필요 없이의 파일 전송 프로토콜 지원 서버 AWS (또는 관리형 파일 전송 웹 인터페이스)에 액세스할 수 있습니다. 이 서비스를 사용하여 최종 사용자의 클라이언트 및 구성을 그대로 AWS 유지하면서 파일 전송 기반 워크플로를 로 마이그레이션할 수 있습니다. 서버의 경우 먼저 호스트 이름을 서버 엔드포인트와 연결한 다음 사용자를 추가하고 적절한 액세스

스 수준으로 프로비저닝합니다. 이 작업이 끝나면, 사용자의 전송 요청은 Transfer Family 서버 엔드포인트에서 바로 처리됩니다.

Transfer Family 웹 앱에서 구성 설정을 결정하고 선택적 사용자 지정을 적용합니다. 이렇게 하면 사용자가 로그인하여 Amazon S3와 직접 데이터를 주고받을 수 있습니다.

Transfer Family는 다음과 같은 이점이 있습니다.

- 사용자의 필요에 맞게 실시간으로 확장되는 종합 관리형 서비스입니다.
- 애플리케이션을 수정하거나 File Transfer 프로토콜 인프라를 실행하지 않아도 됩니다.
- 내구성이 뛰어난 Amazon S3 스토리지의 데이터를 사용하면 처리, 분석, 보고, 감사 및 보관 함수 AWS 서비스 에 네이티브를 사용할 수 있습니다.
- Amazon EFS를 데이터 스토어로 사용하면 AWS 클라우드 서비스 및 온프레미스 리소스와 함께 사용할 수 있는 완전관리형 탄력적 파일 시스템을 얻을 수 있습니다. Amazon EFS는 애플리케이션을 중단하지 않고 페타바이트까지 온디맨드 확장 및 축소되도록 구축되어 파일을 추가하고 제거할 때 마다 파일을 자동 확장 및 축소합니다. 따라서 확장을 수용하기 위해 용량을 프로비저닝하고 관리할 필요가 없습니다.
- AWS Transfer Family를 사용하여 업로드한 파일의 처리를 쉽게 설정, 실행, 자동화 및 모니터링할 수 있게 해주는 완전 관리형 서버리스 File Transfer 워크플로 서비스입니다.
- 선결제 비용이 없으며, 서비스를 사용한 만큼만 지불하면 됩니다.

다음 섹션에서는 Transfer Family의 다양한 특성에 대한 설명, 시작하기 자습서, 다양한 프로토콜 지원 서버를 설정하는 방법에 대한 자세한 지침, 다양한 타입의 ID 제공자를 사용하는 방법, 서비스의 API 참조를 확인할 수 있습니다.

Transfer Family를 시작하려면 다음을 참조하세요.

- [AWS Transfer Family 작동 방식](#)
- [사전 조건](#)
- [AWS Transfer Family 서버 엔드포인트 시작하기](#)
- [Transfer Family 웹 앱](#)

AWS Transfer Family 작동 방식

AWS Transfer Family 는 다음 프로토콜 또는 웹 브라우저를 통해 Amazon Simple Storage Service(Amazon S3) 스토리지 또는 Amazon Elastic File System(Amazon EFS) 파일 시스템으로 파일을 송수신하는 데 사용할 수 있는 완전관리형 AWS 서비스입니다.

- SFTP(Secure File Transfer Protocol): 버전 3

공식 IETF 문서는 [SSH 파일 전송 프로토콜 draft-ietf-secsh-filexfer-02.txt](#)입니다.

- File Transfer 프로토콜 보안(FTPS)
- File Transfer 프로토콜(FTP)
- 적용성 보고서 2(AS2)
- 브라우저 기반 전송

AWS Transfer Family 는 최대 3개의 가용 영역을 지원하며 연결 및 전송 요청을 위한 Auto Scaling 중 복 플릿을 지원합니다. 지연 시간 기반 라우팅을 사용하여 중복성을 높이고 네트워크 지연 시간을 최소화하는 방법에 대한 예는 블로그 게시물 [SFTP 서버 AWS 전송으로 네트워크 지연 시간 최소화를 참조하세요](#).

Transfer Family 관리형 File Transfer 워크플로(MFTW)는 완전 관리형 서버리스 File Transfer 워크플로 서비스로, AWS Transfer Family를 사용하여 업로드한 파일의 처리를 쉽게 설정, 실행, 자동화 및 모니터링할 수 있습니다. 고객은 MFTW를 사용하여 Transfer Family를 사용하여 전송된 데이터의 복사, 태그 지정, 스캔, 필터링, 압축/압축 해제, 암호화/암호 해독과 같은 다양한 처리 단계를 자동화할 수 있습니다. 이를 통해 추적 및 감사에 대한 엔드 투 엔드 가시성이 제공됩니다. 자세한 내용은 [AWS Transfer Family 관리형 워크플로](#)를 참조하세요.

AWS Transfer Family 는 모든 표준 파일 전송 프로토콜 클라이언트를 지원합니다. 다음은 대표적인 클라이언트입니다.

- [OpenSSH](#) – Macintosh 및 Linux 명령줄 유틸리티.
- [WinSCP](#) – Windows 전용 그래픽 클라이언트.
- [Cyberduck](#) – Linux, Macintosh, Microsoft Windows 그래픽 클라이언트.
- [FileZilla](#) – Linux, Macintosh, Windows 그래픽 클라이언트.

AWS 는 다음과 같은 Transfer Family 워크숍을 제공합니다.

- 관리형 SFTP/FTPS 엔드포인트를 활용하고 사용자 관리를 AWS Transfer Family 위해 Amazon Cognito 및 DynamoDB를 활용하는 파일 전송 솔루션을 구축합니다. 이 워크숍의 세부 정보는 [여기](#)에서 확인할 수 있습니다.
- AS2가 활성화된 Transfer Family 엔드포인트와 Transfer Family AS2 커넥터를 빌드합니다. 여기에 서이 워크숍의 세부 정보를 볼 수 [있습니다](#).
- 기존 애플리케이션을 수정하거나 서버 인프라를 관리할 필요 AWS 없이에서 확장 가능하고 안전한 파일 전송 아키텍처를 구축하는 방법에 대한 규범적 지침과 실습을 제공하는 솔루션을 구축합니다. 이 워크숍의 세부 정보는 [여기](#)에서 확인할 수 있습니다.

Transfer Family와 관련된 블로그 게시물

다음 표에는 Transfer Family 고객에게 유용한 정보가 들어 있는 블로그 게시물이 나와 있습니다. 테이블은 역순으로 되어 있으므로 가장 최근 게시물은 테이블의 시작 부분에 있습니다.

블로그 게시물 제목 및 링크	Date
Okta를에 대한 사용자 지정 자격 증명 공급자로 배포 AWS Transfer Family	2025년 8월 11일
paper-to-electronic 의료 클레임 처리 자동화 AWS	2025년 5월 29일
맬웨어 방지를 위해 AWS Transfer Family 및 GuardDuty를 사용하는 방법	2025년 4월 30일
를 사용하여 FICO가 ETL 자동화를 통해 파일 전송을 현대화하는 방법 AWS Transfer Family	2025년 4월 24일
완전 관리형 Amazon S3 파일 전송을 위한 AWS Transfer Family 웹 앱 발표	2024년 12월 1일
서버 보안을 개선하기 위한 6가지 AWS Transfer Family 팁	2024년 9월 24일
에 대한 사용자 지정 자격 증명 공급자를 사용하여 Active Directory 인증 간소화 AWS Transfer Family	2024년 8월 12일

블로그 게시물 제목 및 링크	Date
AWS Transfer Family SFTP 커넥터 및 PGP 암호화를 사용하여 안전하고 규정을 준수하는 관리형 파일 전송 설계	2024년 5월 16일
AWS Transfer Family 및 Amazon S3에서 Amazon Cognito를 자격 증명 공급자로 사용 Amazon S3	2024년 5월 14일
Transfer Family가 안전하고 규정을 준수하는 관리형 파일 전송 솔루션을 구축하는 데 도움이 되는 방법	2024년 1월 3일
를 사용하여 맬웨어 위협 탐지 AWS Transfer Family	2023년 7월 20일
를 사용하여 SAP 워크로드 확장 AWS Transfer Family	2023년 7월 13일
PGP 및를 사용하여 파일 암호화 및 복호화 AWS Transfer Family	2023년 6월 21일
Azure Active Directory 및를 AWS Transfer Family 사용하여 인증 AWS Lambda	2022년 12월 15일
AWS Transfer Family 관리형 워크플로를 사용하여 파일 전송 알림 사용자 지정	2022년 10월 14일
AWS Transfer Family 워크플로를 사용하여 클라우드 네이티브 File Transfer 플랫폼 구축	2022년 1월 5일
A AWS Transfer Family 및를 사용하여 사용자 셀프 서비스 키 관리를 활성화합니다 AWS Lambda.	2021년 12월 17일
AWS Transfer Family 및 Amazon S3를 사용하여 데이터 액세스 제어 개선	2021년 10월 5일

블로그 게시물 제목 및 링크	Date
AWS Global Accelerator 및 AWS Transfer Family 서비스를 사용하여 인터넷 연결 파일 전송 처리량 개선	2021년 6월 7일
AWS 웹 애플리케이션 방화벽 및 Amazon API Gateway AWS Transfer Family 를 사용한 보안	2021년 5월 5일
AWS 웹 애플리케이션 방화벽 및 Amazon API Gateway AWS Transfer Family 를 사용한 보안	2021년 1월 15일
AWS Transfer Family Amazon Elastic File System에 대한 지원	2021년 1월 7일
를 AWS Transfer Family 사용하기 위한 암호 인증 활성화 AWS Secrets Manager	2020년 11월 5일
AWS Transfer Family 및를 사용하여 데이터 액세스 중앙 집중화 AWS Storage Gateway	2020년 6월 22일
서버리스 애플리케이션에서 AWS Lambda 용 Amazon EFS 사용	2020년 6월 18일
IP 허용 목록을 사용하여 AWS Transfer Family 서버 보호	2020년 4월 8일
SFTP 서버의 AWS 전송으로 네트워크 지연 시간 최소화	2020년 2월 19일
SFTP 서버를 로 리프트 앤 시프트 마이그레이션 AWS	2020년 2월 12일
chroot 및 논리적 디렉터리를 사용하여 AWS SFTP 구조 간소화	2019년 9월 26일
에서 Okta를 자격 증명 공급자로 사용 AWS Transfer Family	2019년 5월 30일

사전 조건

다음 섹션에서는 AWS Transfer Family 서비스를 사용하는 데 필요한 사전 조건을 설명합니다. 최소한 Amazon Simple Storage Service(Amazon S3) 버킷을 생성하고 AWS Identity and Access Management (IAM) 역할을 통해 해당 버킷에 대한 액세스를 제공해야 합니다. 또한 사용자는 신뢰 관계도 수립해야 합니다. 이러한 신뢰 관계가 있어야 Transfer Family가 IAM 역할을 수임해 버킷에 액세스하고, 사용자의 파일 전송 요청을 처리할 수 있습니다.

AWS Transfer Family 서버의 IPv6 지원에 대한 자세한 내용은 서버 관리 [Transfer Family 서버에 대한 IPv6 지원](#) 장의 섹션을 참조하세요.

주제

- [Transfer Family 서버에 지원되는 AWS 리전, 엔드포인트 및 할당량](#)
- [에 가입 AWS](#)
- [AWS Transfer Family 서버에 사용할 스토리지 구성](#)
- [IAM 역할 및 정책 생성](#)

Transfer Family 서버에 지원되는 AWS 리전, 엔드포인트 및 할당량

AWS 서비스에 프로그래밍 방식으로 연결하려면 엔드포인트를 사용합니다. 예를 들어 미국 동부(오하이오) 리전()의 고객을 위한 엔드포인트는 us-east-2입니다transfer.us-east-2.amazonaws.com. 한도라고도 하는 서비스 할당량은 AWS 계정에 대한 서비스 리소스 또는 작업의 최대 수입니다. 이 가이드에서는 [AS2 할당량](#) 및에서 할당량을 찾을 수 있습니다[Transfer for SFTP 커넥터 할당량](#).

지원되는 AWS 리전, 엔드포인트 및 서비스 할당량에 대한 자세한 내용은의 [AWS Transfer Family 엔드포인트 및 할당량을 참조하세요](#) Amazon Web Services 일반 참조.

Transfer Family 웹 앱의 경우 지원되는 리전이에 나열되어 있습니다[AWS 리전 Transfer Family 웹 앱 용](#). Transfer Family 웹 앱과 관련된 할당량은 섹션을 참조하세요[웹 앱 할당량](#).

에 가입 AWS

Amazon Web Services(AWS)에 가입하면 AWS포함한 모든 서비스에 AWS 계정이 자동으로 등록됩니다 AWS Transfer Family. 사용자에게는 사용한 서비스에 대해서만 요금이 청구됩니다.

AWS 계정이 이미 있는 경우 다음 작업으로 건너뛴니다. AWS 계정이 없는 경우 다음 절차에 따라 계정을 생성합니다.

이 없는 경우 다음 단계를 AWS 계정완료하여 생성합니다.

에 가입하려면 AWS 계정

1. <https://portal.aws.amazon.com/billing/signup>을 엽니다.
2. 온라인 지시 사항을 따르세요.

등록 절차 중 전화 또는 텍스트 메시지를 받고 전화 키패드로 확인 코드를 입력하는 과정이 있습니다.

에 가입하면 AWS 계정AWS 계정 루트 사용자의 생성됩니다. 루트 사용자에게는 계정의 모든 AWS 서비스 및 리소스에 액세스할 권한이 있습니다. 보안 모범 사례는 사용자에게 관리 액세스 권한을 할당하고, 루트 사용자만 사용하여 [루트 사용자 액세스 권한이 필요한 작업](#)을 수행하는 것입니다.

Transfer Family 사용 비용 견적을 AWS Pricing Calculator 얻는 데 사용할 요금 및에 대한 자세한 내용은 [AWS Transfer Family 요금](#)을 참조하세요.

AWS 리전 가용성에 대한 자세한 내용은 [AWS Transfer Family의 엔드포인트 및 할당량을 참조하세요](#) AWS 일반 참조.

AWS Transfer Family 서버에 사용할 스토리지 구성

이 주제에서는 사용할 수 있는 스토리지 옵션에 대해 설명합니다 AWS Transfer Family. Amazon S3 또는 Amazon EFS를 Transfer Family 서버의 스토리지로 사용할 수 있습니다.

목차

- [Amazon S3 버킷 구성](#)
 - [Amazon S3 액세스 포인트](#)
 - [Amazon S3 HeadObject 동작](#)
 - [파일을 쓰고 나열할 수 있는 권한만 부여](#)
 - [많은 수의 제로바이트 객체로 인해 지연 시간 문제가 발생합니다.](#)
- [Amazon EFS 파일 시스템 구성](#)
 - [Amazon EFS 파일 소유권](#)

- [Transfer Family를 위한 Amazon EFS 사용자 설정](#)
 - [Amazon EFS에서 Transfer Family 사용자 구성](#)
 - [Amazon EFS 루트 사용자 생성](#)
- [지원되는 Amazon EFS 명령](#)

Amazon S3 버킷 구성

AWS Transfer Family 는 Amazon S3 버킷에 액세스하여 사용자의 전송 요청을 처리하므로 파일 전송 프로토콜 지원 서버 설정의 일부로 Amazon S3 버킷을 제공해야 합니다. 기존 버킷을 사용할 수도 있고, 새 버킷을 만들어도 됩니다.

Note

동일한 AWS 리전에 있는 SFTP 서버 및 S3 버킷을 사용할 필요는 없지만 모범 사례로 권장됩니다.

사용자를 설정할 때, 각 사용자에게 IAM 역할을 할당해야 합니다. 이 역할은 Amazon S3 버킷에 대한 사용자의 액세스 수준을 결정합니다.

새 버킷 생성에 대한 자세한 내용은 Amazon Simple Storage Service 사용 설명서의 [S3 버킷을 생성하려면 어떻게 해야 하나요?](#)를 참조하세요.

Note

Amazon S3 객체 잠금을 사용하면 일정한 시간 동안 또는 무기한으로 객체를 덮어쓰지 않도록 할 수 있습니다. 이는 다른 서비스와 마찬가지로 Transfer Family에서도 동일한 방식으로 작동합니다. 객체가 존재하고 보호되는 경우 해당 파일에 쓰거나 삭제할 수 없습니다. Amazon S3 객체 잠금에 대한 자세한 내용은 [Amazon Simple Storage Service 사용 설명서](#)의 Amazon S3 객체 잠금 사용을 참조하세요.

Amazon S3 액세스 포인트

AWS Transfer Family 는 공유 데이터 세트에 대한 세분화된 [Amazon S3 액세스를 쉽게 관리할 수 있는 Amazon S3의 기능인 Amazon S3 액세스 포인트](#)를 지원합니다. Amazon S3 버킷 이름을 사용하는 모든 곳에서 S3 액세스 포인트 별칭을 사용할 수 있습니다. Amazon S3 버킷의 공유 데이터에 액세스

할 수 있는 서로 다른 권한을 가진 사용자를 위해 Amazon S3에 수백 개의 액세스 포인트를 생성할 수 있습니다.

예를 들어 액세스 포인트를 사용하여 서로 다른 세 팀이 동일한 공유 데이터 세트에 액세스하도록 허용할 수 있습니다. 한 팀은 S3에서 데이터를 읽고, 두 번째 팀은 S3에 데이터를 쓰고, 세 번째 팀은 S3에서 데이터를 읽고, 쓰고, 삭제할 수 있습니다. 위에서 언급한 대로 세분화된 액세스 제어를 구현하려면 여러 팀에 비대칭 액세스를 제공하는 정책이 포함된 S3 액세스 포인트를 생성할 수 있습니다. Transfer Family 서버와 함께 S3 액세스 포인트를 사용하면 수백 개의 사용 사례를 아우르는 복잡한 S3 버킷 정책을 만들지 않고도 세밀한 액세스 제어를 달성할 수 있습니다. Transfer Family 서버에서 S3 액세스 포인트를 사용하는 방법에 대한 자세한 내용은 [AWS Transfer Family 및 Amazon S3 블로그 게시물을 사용하여 데이터 액세스 제어 강화](#)를 참조하세요.

Note

AWS Transfer Family 는 현재 Amazon S3 다중 리전 액세스 포인트를 지원하지 않습니다.

Amazon S3 HeadObject 동작

Note

Transfer Family 서버를 생성하거나 업데이트할 때 Amazon S3 디렉터리의 성능을 최적화하여 HeadObject 호출을 제거할 수 있습니다.

Amazon S3에서 버킷과 객체는 기본 리소스이며 객체는 버킷에 저장됩니다. Amazon S3는 계층적 파일 시스템을 모방할 수 있지만 때로는 일반적인 파일 시스템과 다르게 동작할 수 있습니다. 예를 들어, Amazon S3에서는 디렉터리가 일급 개념이 아니라 객체 키를 기반으로 합니다. AWS Transfer Family 는 객체의 키를 슬래시 문자(/)로 분할하고 마지막 요소를 파일 이름으로 취급한 다음 접두사가 같은 파일 이름을 동일한 경로 아래에 그룹화하여 디렉터리 경로를 유추합니다. mkdir를 사용하거나 Amazon S3 콘솔을 사용하여 빈 디렉터리를 생성할 때 폴더 경로를 나타내는 0바이트 객체가 생성됩니다. 이러한 객체의 키는 후행 슬래시로 끝납니다. 이러한 0바이트 객체에 대한 설명은 Amazon S3 사용 설명서의 [폴더를 사용하여 Amazon S3 콘솔에서 객체 구성](#)에 설명되어 있습니다.

ls 명령을 실행하고 일부 결과가 Amazon S3 0바이트 객체(이러한 객체에는 슬래시 문자로 끝나는 키가 있음)가 생성되면 Transfer Family는 각 객체에 대한 HeadObject 요청을 발행합니다(자세한 내용은 Amazon Simple Storage Service API 참조의 [HeadObject](#) 참조). 이로 인해 Transfer Family를 통해 Amazon S3를 스토리지로 사용할 때 다음과 같은 문제가 발생할 수 있습니다.

파일을 쓰고 나열할 수 있는 권한만 부여

경우에 따라 Amazon S3 객체에 대한 쓰기 액세스 권한만 제공할 수 있습니다. 예를 들어 버킷의 객체를 쓰기(또는 업로드) 및 나열할 수 있는 액세스 권한을 제공하지만 객체를 읽기(다운로드)할 수는 없습니다. 파일 전송 클라이언트를 사용하여 `ls` 및 `mkdir` 명령을 수행하려면 Amazon S3 `ListObjects` 및 `PutObject` 권한이 있어야 합니다. 그러나 Transfer Family가 파일 쓰기 또는 나열을 `HeadObject` 호출해야 하는 경우이 호출에는 `GetObject` 권한이 필요하기 때문에 액세스 거부 오류와 함께 호출이 실패합니다.

Note

Transfer Family 서버를 생성하거나 업데이트할 때 Amazon S3 디렉터리의 성능을 최적화하여 `HeadObject` 호출을 제거할 수 있습니다.

이 경우 슬래시 AWS Identity and Access Management ()로 끝나는 객체에 대해서만 `GetObject` 권한을 추가하는 (IAM) 정책 조건을 추가하여 액세스 권한을 부여할 수 있습니다/. 이 조건은 파일에 대한 `GetObject` 호출을 방지하지만(읽을 수 없도록) 사용자가 폴더를 나열하고 순회할 수 있도록 허용합니다. 다음 예제 정책은 Amazon S3 버킷에 대한 쓰기 및 목록 액세스만 제공합니다. 이 정책을 사용하려면 `amzn-s3-demo-bucket`를 버킷 이름으로 바꿉니다.

Note

WinSCP의 업로드 동작을 해결하려면 다음 예제 정책에 나열된 대로 `"arn:aws:s3:::amzn-s3-demo-bucket/*.filepart"` 줄을 추가해야 합니다. 이 줄은 `.filepart` 객체를 적절하게 처리하여 장애를 방지합니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowListing",
      "Effect": "Allow",
      "Action": "s3:ListBucket",
      "Resource": "arn:aws:s3:::amzn-s3-demo-bucket"
    },
    {
      "Sid": "AllowReadWrite",
```



```

    "Effect": "Allow",
    "Action": [
        "s3:PutObject",
        "s3:GetObject",
        "s3:GetObjectVersion"
    ],
    "Resource": [
        "arn:aws:s3:::amzn-s3-demo-bucket/*"
    ]
},
{
    "Sid": "DenyIfNotFolder",
    "Effect": "Deny",
    "Action": [
        "s3:GetObject",
        "s3:GetObjectVersion"
    ],
    "NotResource": [
        "arn:aws:s3:::amzn-s3-demo-bucket/*/\"",
        "arn:aws:s3:::amzn-s3-demo-bucket/*.filepart"
    ]
}
]
}

```

Note

이 정책은 사용자의 파일 추가를 허용하지 않습니다. 즉, 이 정책을 적용받는 사용자는 파일을 열어 콘텐츠에 콘텐츠를 추가하거나 수정할 수 없습니다. 또한 사용 사례에서 파일을 업로드하기 전에 HeadObject를 호출해야 하는 경우에는 이 정책이 적용되지 않습니다.

많은 수의 제로바이트 객체로 인해 지연 시간 문제가 발생합니다.

Amazon S3 버킷에 이러한 0바이트 객체가 많은 경우 Transfer Family는 많은 HeadObject 호출을 실행하여 처리가 지연될 수 있습니다. 이 문제의 권장 해결 방법은 최적화된 디렉터리를 활성화하여 지연 시간을 줄이는 것입니다.

예를 들어 홈 디렉터리로 이동하고 하위 디렉터리가 10,000개 있다고 가정해 보겠습니다. 즉, Amazon S3 버킷에는 10,000개의 폴더가 있습니다. 이 시나리오에서 ls (list) 명령을 실행하면 목록 작업이 6~8 분 정도 걸립니다. 그러나 디렉터리를 최적화하는 경우이 작업은 몇 초 밖에 걸리지 않습니다. 서버

생성 또는 업데이트 절차 중에 추가 세부 정보 구성 화면에서이 옵션을 설정합니다. 이 절차는 [SFTP, FTPS 또는 FTP 서버 엔드포인트 구성](#) 주제 아래에 자세히 설명되어 있습니다.

Note

GUI 클라이언트는 사용자가 제어할 수 없는 `ls` 명령을 실행할 수 있으므로 가능한 경우이 설정을 활성화하는 것이 중요합니다.

디렉터리를 최적화하지 않거나 최적화할 수 없는 경우이 문제에 대한 대체 해결 방법은 0바이트 객체를 모두 삭제하는 것입니다. 다음 사항에 유의하세요.

- 빈 디렉터리가 더 이상 존재하지 않습니다. 디렉터리는 객체 키에 이름이 포함되어 있기 때문에 존재할 수 있습니다.
- 다른 사용자가 `mkdir`을 호출하여 문제가 생기는 것을 막지는 못합니다. 디렉터리 생성을 방지하는 정책을 만들어 이 문제를 완화할 수 있습니다.
- 일부 시나리오에서는 이러한 0바이트 객체를 사용합니다. 예를 들어 `/inboxes/customer1000`과 같은 구조를 사용하면 받은 편지함 디렉터리가 매일 정리됩니다.

마지막으로, 한 가지 가능한 해결 방법은 정책 조건을 통해 표시되는 객체 수를 제한하여 `HeadObject` 호출 수를 줄이는 것입니다. 실행 가능한 솔루션이 되려면 모든 하위 디렉터리의 제한된 집합만 볼 수 있다는 점을 인정해야 합니다.

Amazon EFS 파일 시스템 구성

AWS Transfer Family 는 Amazon Elastic File System(Amazon EFS)에 액세스하여 사용자의 전송 요청을 처리합니다. 따라서 파일 전송 프로토콜 지원 서버 설정의 일환으로 Amazon EFS 파일 시스템을 제공해야 합니다. 기존 파일 시스템을 사용할 수도 있고, 새 버킷을 만들어도 됩니다.

다음 사항에 유의하세요.

- Transfer Family 서버와 Amazon EFS 파일 시스템을 사용하는 경우 서버와 파일 시스템이 동일해야 합니다 AWS 리전.
- 서버와 파일 시스템이 같은 계정에 속할 필요는 없습니다. 서버와 파일 시스템이 같은 계정에 속하지 않는 경우 파일 시스템 정책에서 사용자 역할에 명시적 권한을 부여해야 합니다.

여러 계정을 설정하는 방법에 대한 자세한 내용은 AWS Organizations 사용 설명서 [의 조직 내 AWS 계정 관리를 참조하세요](#).

- 사용자를 설정할 때, 각 사용자에게 IAM 역할을 할당해야 합니다. 이 역할은 Amazon EFS 파일 시스템에 대한 사용자의 액세스 수준을 결정합니다.
- Amazon EFS 파일 시스템 탑재에 대한 자세한 내용은 [Amazon EFS 파일 시스템 탑재](#)를 참조하세요.

AWS Transfer Family 및 Amazon EFS가 함께 작동하는 방법에 대한 자세한 내용은 Amazon Amazon Elastic File System 사용 설명서의 [사용하여 Amazon EFS 파일 시스템의 파일에 AWS Transfer Family 액세스를 참조하세요](#).

Amazon EFS 파일 소유권

Amazon EFS는 Portable Operating System Interface(POSIX) 파일 권한 모델을 사용하여 파일 소유권을 나타냅니다.

POSIX에서 시스템의 사용자는 세 가지 고유한 권한 클래스로 분류됩니다. 사용자가 사용하여 Amazon EFS 파일 시스템에 저장된 파일에 액세스하도록 허용하는 AWS Transfer Family 경우 해당 사용자에게 “POSIX 프로필”을 할당해야 합니다. 이 프로필은 Amazon EFS 파일 시스템의 파일 및 디렉터리에 대한 액세스 권한을 결정하는 데 사용됩니다.

- 사용자(u): 파일 또는 디렉터리의 소유자. 일반적으로 파일이나 디렉토리를 만든 사람이 소유자이기도 합니다.
- 그룹(g): 공유하는 파일 및 디렉터리에 대해 동일한 액세스 권한이 필요한 사용자 집합입니다.
- 기타(o): 소유자 및 그룹 구성원을 제외하고 시스템에 액세스할 수 있는 다른 모든 사용자. 이 권한 클래스를 “Public”이라고도 합니다.

POSIX 권한 모델에서는 모든 파일 시스템 개체(파일, 디렉터리, 심볼릭 링크, 명명된 파이프, 소켓)가 앞서 언급한 세 가지 권한 집합과 연결됩니다. Amazon EFS 객체에는 연결된 Unix 스타일 모드가 있습니다. 이 모드 값은 해당 객체에 대한 작업을 수행할 수 있는 권한을 정의합니다.

또한 Unix 스타일 시스템에서 사용자와 그룹은 Amazon EFS가 파일 소유권을 나타내는 데 사용하는 숫자 식별자에 매핑됩니다. Amazon EFS의 경우 단일 소유자 및 단일 그룹이 객체를 소유합니다. Amazon EFS는 사용자가 파일 시스템 객체에 액세스하려고 할 때 매핑된 숫자 ID를 사용하여 권한을 확인합니다.

Transfer Family를 위한 Amazon EFS 사용자 설정

Amazon EFS 사용자를 설정하기 전에 다음 중 하나를 수행할 수 있습니다.

- Amazon EFS에서 사용자를 생성하고 홈 폴더를 설정할 수 있습니다. 세부 정보는 [Amazon EFS에서 Transfer Family 사용자 구성](#)를 참조하세요.
- 루트 사용자를 추가하는 것이 편하다면 [Amazon EFS 루트 사용자 생성](#)을 할 수 있습니다.

Note

Transfer Family 서버는 POSIX 권한을 설정하는 Amazon EFS 액세스 포인트를 지원하지 않습니다. Transfer Family 사용자의 POSIX 프로필(이전 섹션에서 설명)은 POSIX 권한을 설정하는 기능을 제공합니다. 이러한 권한은 UID, GID 및 보조 GID를 기반으로 세분화된 액세스를 위해 사용자 수준에서 설정됩니다.

Amazon EFS에서 Transfer Family 사용자 구성

Transfer Family는 사용자를 지정한 UID/GID 및 디렉터리에 매핑합니다. UID/GID/디렉토리가 EFS에 아직 없는 경우 Transfer에서 사용자에게 할당하기 전에 먼저 UID/GID/디렉토리를 생성해야 합니다. Amazon EFS 사용자 생성에 대한 자세한 내용은 Amazon Elastic File System 사용 설명서의 [NFS\(Network File System\) 수준에서의 사용자, 그룹 및 권한 작업](#)을 참조하세요.

Transfer Family에서 Amazon EFS 사용자를 설정하는 단계

1. [PosixProfile](#) 필드를 사용하여 Transfer Family에서 사용자에 대한 EFS UID 및 GID를 매핑합니다.
2. 사용자가 로그인 시 특정 폴더에서 시작하도록 하려면 [HomeDirectory](#) 필드 아래에 EFS 디렉토리를 지정할 수 있습니다.

CloudWatch 규칙 및 Lambda 함수를 사용하여 프로세스를 자동화할 수 있습니다. EFS와 상호 작용하는 Lambda 함수 예제는 [서버리스 애플리케이션에서 AWS Lambda 용 Amazon EFS 사용을 참조하세요](#).

또한 Transfer Family 사용자를 위한 논리적 디렉토리를 구성할 수 있습니다. 자세한 내용은 [논리적 디렉토리를 사용하여 Transfer Family 디렉터리 구조를 단순화합니다](#). 항목의 [Amazon EFS를 위한 논리적 디렉터리 구성](#)을 참조하세요.

Amazon EFS 루트 사용자 생성

조직에서 사용자 구성을 위해 SFTP/FTPS를 통한 루트 사용자 액세스를 허용하려는 경우 UID 및 GID가 0(루트 사용자)인 사용자를 생성한 다음 해당 루트 사용자를 사용하여 폴더를 생성하고 나머지 사

용자의 POSIX ID 소유자를 할당할 수 있습니다. 이 옵션의 장점은 Amazon EFS 파일 시스템을 탑재할 필요가 없다는 것입니다.

[Amazon EFS 서비스 관리 사용자 추가](#)에 설명된 단계를 수행하고, 사용자 ID와 그룹 ID 모두에 0을 입력합니다.

 Tip

이 수퍼유저 계정이 필요 이상으로 오래 존재하지 않도록 합니다. 또는 루트 사용자 계정을 유지하는 경우 잘 보호해야 합니다.

지원되는 Amazon EFS 명령

다음 명령은 AWS Transfer Family를 위한 Amazon EFS에서 지원됩니다.

- cd
- ls/dir
- pwd
- put
- get
- rename
- chown: 루트(즉, uid=0인 사용자)만 파일 및 디렉터리의 소유권과 권한을 변경할 수 있습니다.
- chmod: 루트(즉, uid=0인 사용자)만 파일 및 디렉터리의 소유권과 권한을 변경할 수 있습니다.
- chgrp: 루트 또는 파일 소유자만 파일 그룹을 보조 그룹 중 하나로 변경할 수 있는 파일 소유자만 지원됩니다.
- ln -s/symlink
- mkdir
- rm/delete
- rmdir
- chmtime

IAM 역할 및 정책 생성

이 주제에서는 함께 사용할 수 있는 정책 및 역할 유형을 설명하고 사용자 역할 생성 프로세스를 AWS Transfer Family에 안내합니다. 또한 세션 정책의 작동 방식을 설명하고 사용자 역할 예를 제공합니다.

AWS Transfer Family 는 다음과 같은 유형의 역할을 사용합니다.

- 사용자 역할 - 서비스 관리 사용자가 필요한 Transfer Family 리소스에 액세스할 수 있도록 허용합니다. Transfer Family 사용자 ARN의 컨텍스트에서 이 역할을 수 AWS Transfer Family 임합니다.
- 액세스 역할 - 전송 중인 Amazon S3 파일에만 액세스할 수 있습니다. 인바운드 AS2 전송의 경우 액세스 역할은 계약의 Amazon 리소스 이름(ARN)을 사용합니다. 아웃바운드 AS2 전송의 경우 액세스 역할은 커넥터의 ARN을 사용합니다.
- 간접 호출 역할 - Amazon API Gateway와 함께 서버의 사용자 지정 자격 증명 공급자로 사용합니다. Transfer Family는 Transfer Family 서버 ARN의 컨텍스트에서 이 역할을 담당합니다.
- 로깅 역할 - Amazon CloudWatch에 항목을 기록하는 데 사용됩니다. Transfer Family는 이 역할을 사용하여 File Transfer에 대한 정보와 함께 성공 및 실패 세부 정보를 기록합니다. Transfer Family는 Transfer Family 서버 ARN의 컨텍스트에서 이 역할을 담당합니다. 아웃바운드 AS2 전송의 경우, 로깅 역할은 커넥터 ARN을 사용합니다.
- 실행 역할 - Transfer Family 사용자가 전화를 걸어 워크플로를 시작할 수 있습니다. Transfer Family는 Transfer Family 워크플로 ARN의 컨텍스트에서 이 역할을 담당합니다.

이러한 역할 외에도 세션 정책을 사용할 수 있습니다. 세션 정책은 필요한 경우 액세스를 제한하는 데 사용됩니다. 이러한 정책은 독립형이라는 점에 유의하세요. 즉, 이러한 정책은 역할에 추가할 수 없습니다. 대신 Transfer Family 사용자에게 직접 세션 정책을 추가할 수 있습니다.

Note

서비스 관리 Transfer Family 사용자를 생성하는 경우 홈 폴더를 기반으로 정책 자동 생성을 선택할 수 있습니다. 사용자 액세스를 자신의 폴더에 제한하려는 경우 유용한 단축키입니다. 또한 [세션 정책의 작동 방식](#)에서 세션 정책 및 예에 대한 세부 정보를 볼 수 있습니다. 세션 정책에 대한 자세한 내용은 IAM 사용 설명서의 [세션 정책](#)에서도 찾아볼 수 있습니다.

주제

- [사용자 역할 생성](#)
- [세션 정책의 작동 방식](#)

- [읽기/쓰기 액세스 정책의 예](#)

사용자 역할 생성

사용자를 만들 때, 사용자 액세스에 관한 결정을 여러 번 해야 합니다. 이러한 결정에는 사용자가 액세스할 수 있는 Amazon S3 버킷 또는 Amazon EFS 파일 시스템, 각 Amazon S3 버킷의 일부 및 파일 시스템의 어떤 파일에 액세스할 수 있는지, 사용자에게 어떤 권한(예: PUT 또는 GET)이 포함됩니다.

액세스를 설정하려면 해당 액세스 정보를 제공하는 자격 증명 기반 AWS Identity and Access Management (IAM) 정책 및 역할을 생성합니다. 이 과정의 일부에서, 사용자가 파일 작업의 대상 또는 원본인 Amazon S3 bucket 또는 Amazon EFS에 액세스할 수 있게 됩니다. 이렇게 하려면 나중에 자세하게 설명하는, 다음과 같은 상위 수준 단계를 거쳐야 합니다.

사용자 역할 생성

1. 예에 대한 IAM 정책을 생성합니다 AWS Transfer Family. 이에 대한 설명은 [예에 대한 IAM 정책을 생성하려면 AWS Transfer Family](#)에 나와 있습니다.
2. IAM 역할을 생성하여 새 IAM 정책을 연결합니다. 예는 [읽기/쓰기 액세스 정책의 예](#)를 참조하세요.
3. AWS Transfer Family 와 IAM 역할 간에 신뢰 관계를 설정합니다. 이에 대한 설명은 [신뢰 관계를 구축하기 위해](#)에 나와 있습니다.

다음 절차에서는 IAM 정책 및 역할을 생성하는 방법을 설명합니다.

예에 대한 IAM 정책을 생성하려면 AWS Transfer Family

1. <https://console.aws.amazon.com/iam/>에서 IAM 콘솔을 엽니다.
2. 탐색 창에서 정책을 선택한 후 정책 생성을 선택합니다.
3. 정책 생성 페이지에서 JSON 탭을 선택합니다.
4. 표시되는 편집기에서, 편집기 내용을 IAM 역할에 연결할 IAM 정책으로 바꿉니다.

읽기/쓰기 액세스 권한을 부여하거나 사용자를 홈 디렉터리로 제한할 수 있습니다. 자세한 내용은 [읽기/쓰기 액세스 정책의 예](#)를 참조하세요.

5. 정책 검토를 선택하고 정책의 이름과 설명을 입력한 다음 정책 생성을 선택합니다.

그런 다음 IAM 역할을 만들고 새 IAM 정책을 역할에 연결합니다.

에 대한 IAM 역할을 생성하려면 AWS Transfer Family

1. 탐색 창에서 역할을 선택한 후 역할 생성을 선택합니다.

역할 생성 페이지에서 AWS 서비스가 선택되어 있는지 확인합니다.

2. 서비스 목록에서 전송을 선택하고, 다음: 권한을 선택합니다. 이렇게 하면 AWS Transfer Family 와 간의 신뢰 관계가 설정됩니다 AWS.
3. 권한 정책 연결 섹션에서, 방금 생성한 정책을 찾아 선택한 다음 다음: 태그를 선택합니다.
4. (옵션) 태그의 키와 값을 입력하고 다음: 검토를 선택합니다.
5. 검토 페이지에 새 역할의 명칭과 설명을 입력한 다음 역할 생성을 선택합니다.

다음으로 AWS Transfer Family 와 간에 신뢰 관계를 설정합니다 AWS.

신뢰 관계를 구축하기 위해

Note

이 예에서는 ArnLike 및 ArnEquals 모두를 사용합니다. 기능적으로 동일하므로 정책을 구성할 때 둘 중 하나를 사용할 수 있습니다. Transfer Family 설명서에서는 조건에 와일드 카드 문자가 포함된 경우에는 ArnLike를 사용하고, 정확한 일치 조건을 나타내기 위해서는 ArnEquals를 사용합니다.

1. IAM 콘솔에서 방금 생성한 역할을 선택합니다.
2. 요약 페이지에서 신뢰 관계를 선택한 다음 신뢰 관계 편집을 선택합니다.
3. 신뢰 관계 편집 편집기에서 서비스가 "transfer.amazonaws.com"인지 확인합니다. 액세스 정책은 다음과 같습니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "transfer.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```



```

    }
  ]
}
```

대리인 혼동 문제로부터 스스로를 보호하려면 `aws:SourceAccount` 및 `aws:SourceArn` 조건 키를 사용할 것을 권장합니다. 소스 계정은 서버의 소유자이고 소스 ARN은 사용자의 ARN입니다. 예제:

```

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

사용자 계정의 서버 대신 특정 서버로 제한하려는 경우에도 `ArnLike` 조건을 사용할 수 있습니다. 예제:

```

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

Note

상기 예에서 각 `###` `##` `##` `###`를 자신의 정보로 바꿉니다.

헛갈리는 대리인 문제에 대한 자세한 내용 및 기타 예는 [교차 서비스 혼동된 대리자 방지](#) 단원을 참조하세요.

4. 신뢰 정책 업데이트를 선택하여 액세스 정책을 업데이트합니다.

이제가 사용자를 대신하여 AWS 서비스를 AWS Transfer Family 호출하도록 허용하는 IAM 역할을 생성했습니다. 생성한 IAM 정책과 역할을 연결해, 사용자에게 액세스를 부여했습니다. [AWS Transfer](#)

[Family 서버 엔드포인트 시작하기](#) 섹션에서, 이 역할과 정책은 사용자 또는 일반 사용자에게 할당됩니다.

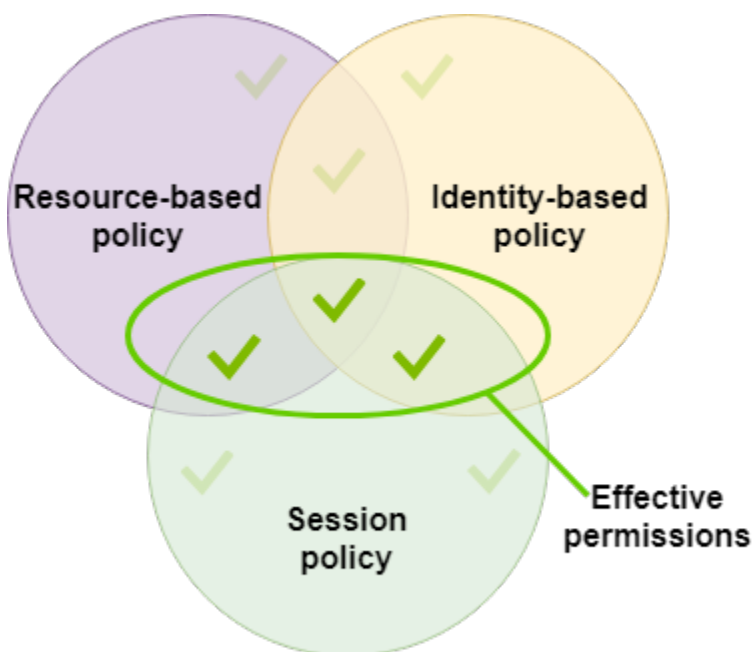
참고 항목

- IAM 역할에 대한 자세한 내용은 IAM 사용 설명서의 [AWS 서비스에 권한을 위임할 역할 생성을 참조하세요](#).
- Amazon S3 리소스의 자격 증명 기반 정책에 대한 자세한 내용은 Amazon Simple Storage Service 사용 설명서의 [Amazon S3의 자격 증명 및 액세스 관리](#)를 참조하세요.
- Amazon EFS 리소스의 자격 증명 기반 정책에 대해 자세히 알아보려면 Amazon Elastic File System 사용 설명서의 [IAM을 사용하여 파일 시스템 데이터 액세스 제어](#)를 참조하세요.

세션 정책의 작동 방식

관리자가 역할을 생성하면 역할에 여러 사용 사례 또는 팀 구성원을 포괄할 수 있는 광범위한 권한이 포함되는 경우가 많습니다. 관리자가 [콘솔 URL](#)을 구성하면 세션 정책을 사용하여 결과 세션에 대한 권한을 줄일 수 있습니다. 예를 들어 [읽기/쓰기 권한](#)이 있는 역할을 생성하는 경우 사용자의 액세스를 홈 디렉터리로만 제한하는 URL을 설정할 수 있습니다.

세션 정책은 역할 또는 사용자에게 대해 임시 세션을 프로그래밍 방식으로 생성할 때 파라미터로 전달하는 고급 정책입니다. 세션 정책은 객체 접두사에 사용자 이름이 포함된 버킷의 일부에만 액세스할 수 있도록 사용자를 잡그는 데 유용합니다. 다음 다이어그램은 세션 정책의 권한이 세션 정책과 리소스 기반 정책의 교집합과 세션 정책 및 자격 증명 기반 정책의 교집합임을 보여줍니다.



자세한 내용은 IAM 사용 설명서의 [세션 정책](#)을 참조하세요.

에서 AWS Transfer Family 세션 정책은 Amazon S3와 주고받는 경우에만 지원됩니다. 다음 예 정책은 사용자가 자신의 home 디렉터리에만 액세스하게 하는 세션 정책입니다. 다음 사항에 유의하세요.

- GetObjectACL 및 PutObjectACL 명령문은 크로스 계정 액세스를 활성화할 필요가 있는 경우에만 요구됩니다. 예컨대, Transfer Family 서버가 다른 계정의 버킷에 액세스할 필요가 있는 경우입니다.
- 세션 정책의 최대 길이는 2,048자입니다. 자세한 내용은 API 참조의 CreateUser 작업에 대한 [정책 요청 파라미터](#)를 참조하세요.
- Amazon S3 버킷이 AWS Key Management Service (AWS KMS)를 사용하여 암호화된 경우 정책에 추가 권한을 지정해야 합니다. 자세한 내용은 [데이터 보호 및 암호화](#)을 참조하세요.
- 세션 정책을 사용하여 각 사용자에게 대해 별도의 IAM 역할을 생성하지 않고 사용자 속성을 기반으로 액세스 권한을 생성하려면 [섹션을 참조하세요 동적 권한 관리 접근 방식](#).

```
{
  "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

위의 정책 예에서는 사용자의 홈 디렉터리가 디렉터리임을 나타내기 위해 후행 슬래시를 포함하도록 설정된 것으로 가정합니다. 그 반면에 후행 슬래시가 없이 사용자의 HomeDirectory를 설정하는 경우에는 이를 정책의 일부로 포함해야 합니다.

이전 예 정책에서, transfer:HomeFolder, transfer:HomeBucket, transfer:HomeDirectory 정책 파라미터의 사용을 기억해 두세요. 이러한 파라미터는 [HomeDirectory](#) 및 [API Gateway 메서드 구현](#)에 설명된 대로 사용자에게 맞게 구성된 HomeDirectory을 위해 설정됩니다 이러한 파라미터의 정의는 다음과 같습니다.

- transfer:HomeBucket 파라미터는 HomeDirectory의 첫 번째 구성 요소로 대체됩니다.
- transfer:HomeFolder 파라미터가 HomeDirectory 파라미터의 나머지 부분으로 대체됩니다.
- transfer:HomeDirectory 파라미터는 Resource 명령문에서 S3 Amazon 리소스 이름 (ARN)의 일부로 사용할 수 있도록 선행 슬래시(/)가 제거되었습니다.

Note

귀하가 논리적 디렉터를 사용하는 경우—즉 사용자의 homeDirectoryType가 LOGICAL인 경우—이러한 정책 파라미터(HomeBucket, HomeDirectory, 및 HomeFolder)는 지원되지 않습니다.

예를 들어, Transfer Family 사용자에게 대해 구성된 HomeDirectory 파라미터가 /home/bob/amazon/stuff/(이)라고 가정해 보겠습니다.

- transfer:HomeBucket를 /home(으)로 설정합니다.
- transfer:HomeFolder를 /bob/amazon/stuff/(으)로 설정합니다.
- transfer:HomeDirectory는 home/bob/amazon/stuff/가 됩니다.

첫 번째 "Sid"은(는) 사용자가 /home/bob/amazon/stuff/부터 시작하여 모든 디렉터리를 나열할 수 있게 해 줍니다.

두 번째 "Sid"은(는) 동일한 경로인 /home/bob/amazon/stuff/에 대한 사용자 put 및 get 액세스를 제한합니다.

읽기/쓰기 액세스 정책의 예

Amazon S3 버킷에 대한 읽기/쓰기 권한 부여

에 대한 다음 예제 정책은 Amazon S3 버킷의 객체에 대한 읽기/쓰기 액세스 권한을 AWS Transfer Family 부여합니다.

다음 사항에 유의하세요.

- amzn-s3-demo-bucket을 Amazon S3 버킷 이름으로 바꿉니다.
- GetObjectACL 및 PutObjectACL 명령문은 크로스 계정 액세스를 활성화할 필요가 있는 경우에만 요구됩니다. 예컨대, Transfer Family 서버가 다른 계정의 버킷에 액세스할 필요가 있는 경우입니다.
- GetObjectVersion 및 DeleteObjectVersion 보고서는 액세스 중인 Amazon S3 버킷에서 버전 관리가 활성화된 경우에만 필요합니다.

Note

버킷에 버전 관리를 활성화한 경우 Amazon S3에서만 버전 관리를 일시 중지할 수 있고 완전히 끌 수는 없으므로 이러한 권한이 필요합니다. 자세한 내용은 [버전 관리되지 않음, 버전 관리 활성화 및 버전 관리가 일시 중지된 버킷을 참조하세요](#).

```
{
```

```

"Version": "2012-10-17",
"Statement": [
  {
    "Sid": "AllowListingOfUserFolder",
    "Action": [
      "s3:ListBucket",
      "s3:GetBucketLocation"
    ],
    "Effect": "Allow",
    "Resource": [
      "arn:aws:s3:::amzn-s3-demo-bucket"
    ]
  },
  {
    "Sid": "HomeDirObjectAccess",
    "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:::amzn-s3-demo-bucket/*"
  }
]
}

```

Amazon EFS 파일 시스템의 파일에 대한 파일 시스템 액세스 권한 부여

Note

정책 외에도 POSIX 파일 권한이 적절한 액세스를 허용하고 있는지도 확인해야 합니다. 자세한 내용은 Amazon Elastic File System 사용 설명서의 [NFS\(네트워크 파일 시스템\) 수준에서 사용자, 그룹, 권한 작업](#)을 참조하세요.

다음 예 정책은 루트 파일 시스템에 Amazon EFS 파일 시스템의 파일에 대한 액세스 권한을 부여합니다.

 Note

다음 예에서 ##을 해당 리전으로, *account-id*를 파일이 있는 계정으로, *file-system-id*를 Amazon Elastic File System(Amazon EFS)의 ID로 바꾸세요.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "RootFileSystemAccess",
      "Effect": "Allow",
      "Action": [
        "elasticfilesystem:ClientRootAccess",
        "elasticfilesystem:ClientMount",
        "elasticfilesystem:ClientWrite"
      ],
      "Resource": "arn:aws:elasticfilesystem:us-east-1:123456789012:file-system/file-system-id"
    }
  ]
}
```

다음 예 정책은 Amazon EFS 파일 시스템의 파일에 대한 사용자 파일 시스템 액세스 권한을 부여합니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "UserFileSystemAccess",
      "Effect": "Allow",
      "Action": [
        "elasticfilesystem:ClientMount",
        "elasticfilesystem:ClientWrite"
      ],
    }
  ]
}
```

```
    "Resource": "arn:aws:elasticfilesystem:us-east-1:123456789012:file-  
system/file-system-id"  
  }  
]  
}
```


Transfer Family 자습서

AWS Transfer Family 사용 설명서는 여러 사용 사례에 대한 자세한 설명을 제공합니다.

- [AWS Transfer Family 서버 엔드포인트 시작하기](#) - 서비스 관리형 사용자로 SFTP Transfer Family 서버를 생성하고 클라이언트를 사용하여 파일을 전송합니다.
- [SFTP 커넥터 설정 및 사용](#) - Amazon S3 스토리지와 SFTP 서버 간에 파일을 전송하도록 SFTP 커넥터를 설정합니다.
- [Amazon API Gateway 메서드를 사용자 지정 자격 증명 공급자로 설정](#) - Amazon API Gateway 메서드를 AWS Transfer Family 서버에 파일을 업로드하기 위한 사용자 지정 자격 증명 공급자로 구성합니다.
- [파일 복호화를 위한 관리형 워크플로 설정](#) - 관리형 워크플로를 생성하여 파일을 해독하고, 암호화된 파일을 Amazon S3에 업로드하고, 해독된 콘텐츠를 봅니다.
- [AS2 구성 설정](#) - 인증서 가져오기, 프로필 및 계약 생성, 선택적 AS2 커넥터 구성 및 테스트를 포함하여 AS2 Transfer Family 서버를 설정합니다.
- 다음 자습서에서는 Transfer Family 웹 애플리케이션을 생성하고 구성하는 방법을 안내합니다. 필요에 가장 적합한 자습서를 선택합니다.

[자습서: 기본 Transfer Family 웹 앱 설정](#) - Transfer Family 웹 애플리케이션을 생성하고 구성하는 방법을 알아봅니다.

[자습서: 선택적 다중 버킷 액세스를 사용하여 AWS Transfer Family 웹 앱 설정](#) - 단일 사용자에게 대한 세분화된 Amazon S3 버킷 권한으로 Transfer Family 웹 애플리케이션을 구성합니다.

주제

- [AWS Transfer Family 서버 엔드포인트 시작하기](#)
- [파일 복호화를 위한 관리형 워크플로 설정](#)
- [SFTP 커넥터 설정 및 사용](#)
- [Amazon API Gateway 메서드를 사용자 지정 자격 증명 공급자로 설정](#)
- [AS2 구성 설정](#)
- [자습서: 기본 Transfer Family 웹 앱 설정](#)
- [자습서: 선택적 다중 버킷 액세스를 사용하여 AWS Transfer Family 웹 앱 설정](#)

AWS Transfer Family 서버 엔드포인트 시작하기

이 자습서를 사용하여 AWS Transfer Family (Transfer Family)를 시작합니다. Amazon S3 스토리지를 사용하여 공개적으로 액세스할 수 있는 엔드포인트가 있는 SFTP 지원 서버를 생성하고, 서비스 관리형 인증을 통해 사용자를 추가하고, Cyberduck으로 파일을 전송하는 방법을 알아봅니다.

주제

- [사전 조건](#)
- [1단계: AWS Transfer Family 콘솔에 로그인](#)
- [2단계: SFTP 지원 서버 생성](#)
- [3단계: 서비스 관리 사용자 추가](#)
- [4단계: 클라이언트를 사용하여 파일 전송](#)

사전 조건

시작하기 전에 먼저 [사전 조건](#) 요건을 충족해야 합니다. 이 설정의 일부로 Amazon Simple Storage Service(Amazon S3) 버킷과 AWS Identity and Access Management (IAM) 사용자 역할을 생성합니다.

AWS Transfer Family 콘솔을 사용하는 데 필요한 권한이 있으며, Amazon Simple Storage Service, Amazon Elastic File System AWS Certificate Manager, Amazon Route 53 등 Transfer Family가 사용하는 다른 AWS 서비스를 구성하는 데 필요한 권한이 있습니다. Amazon Elastic File System 예를 들어 Transfer Family를 AWS 사용하여 파일을 송수신하는 사용자의 경우 AmazonS3FullAccess는 Amazon S3 버킷을 설정하고 사용할 수 있는 권한을 부여합니다. Amazon S3 버킷을 생성하려면 이 정책의 일부 권한이 필요합니다.

Transfer Family 콘솔을 사용하려면 다음이 필요합니다.

- AWSTransferConsoleFullAccess는 SFTP 사용자에게 Transfer Family 리소스를 생성할 수 있는 권한을 부여합니다.
- IAMFullAccess(특히 IAM 역할 생성을 허용하는 정책)는 Transfer Family가 Amazon CloudWatch Logs에 서버의 로깅 역할을 자동으로 생성하거나 서버에 로그인하는 사용자의 사용자 역할을 자동으로 생성하도록 하려는 경우에만 필요합니다.
- VPC 서버 타입을 생성하고 삭제하려면 정책에 ec2:CreateVpcEndpoint 및 ec2:DeleteVpcEndpoints 작업을 추가해야 합니다. 보안을 위해 VPC 엔드포인트 액세스를 제한하는 방법에 대한 자세한 내용은 섹션을 참조하세요 [Transfer Family 서버에 대한 VPC 엔드포인트 액세스 제한](#).

 Note

AmazonS3FullAccess 및 IAMFullAccess 정책 자체는의 일반적인 사용에 필요하지 않습니다 AWS Transfer Family. 여기에는 필요한 모든 권한이 포함되는지 확인할 수 있는 간단한 방법이 나와 있습니다. 또한 이러한 정책은 모든 AWS 고객이 사용할 수 있는 표준 정책인 AWS 관리형 정책입니다. 이러한 정책의 개별 권한을 보고 목적에 필요한 최소 권한을 결정할 수 있습니다.

1단계: AWS Transfer Family 콘솔에 로그인

Transfer Family에 로그인하려면

1. 에 로그인 AWS Management Console 하고 <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 계정 ID 또는 별칭에 AWS 계정사용자 ID를 입력합니다.
3. IAM 사용자 이름에 Transfer Family용으로 생성한 사용자 역할의 이름을 입력합니다.
4. 암호에 AWS 계정 암호를 입력합니다.
5. 로그인을 선택합니다.

2단계: SFTP 지원 서버 생성

Secure Shell(SSH) File Transfer 프로토콜(SFTP)은 인터넷을 통한 안전한 데이터 전송에 사용되는 네트워크 프로토콜입니다. 이 프로토콜은 SSH의 전체 보안 및 인증 기능을 지원합니다. 금융 서비스, 의료, 소매 및 광고와 같은 다양한 산업의 사업 파트너 간의 민감한 정보를 비롯한 데이터를 교환하는 데 널리 사용됩니다.

SFTP 지원 서버를 생성하려면

1. 탐색 창에서 서버를 선택한 다음 서버 생성을 선택합니다.
2. 프로토콜 선택에서 SFTP를 선택하고 다음을 선택합니다.
3. 자격 증명 제공자 선택에서 Transfer Family에 사용자 ID 및 키를 저장하도록 관리 서비스를 선택한 후 다음을 선택합니다.
4. 엔드포인트 선택에서 다음을 수행합니다.
 - a. 엔드포인트 타입으로는 공적으로 액세스 가능 엔드포인트 타입을 선택합니다.

- b. 사용자 지정 호스트 이름의 경우 없음을 선택합니다.
 - c. 다음을 선택합니다.
5. 도메인 선택에서 Amazon S3를 선택합니다.
 6. 추가 세부 정보 구성의 암호화 알고리즘 옵션에서 서버에서 사용하도록 활성화된 암호화 알고리즘이 포함된 보안 정책을 선택합니다. 최신 보안 정책은 기본값입니다. 자세한 내용은 섹션을 참조하세요 [AWS Transfer Family 서버에 대한 보안 정책](#).

Note

서버에 대한 관리형 워크플로를 추가하는 경우에만 CloudWatch 로깅을 위한 새 역할 생성을 선택합니다. 서버 이벤트를 로깅하려면 IAM 역할을 생성할 필요가 없습니다.

7. 검토 및 생성에서 서버 생성을 선택합니다. 서버 페이지로 이동합니다.

새 서버 상태가 온라인으로 변경되기까지 몇 분 정도 걸릴 수 있습니다. 이때 서버에서 파일 작업을 수행할 수 있지만 먼저 사용자를 만들어야 합니다. 사용자 생성에 대한 자세한 내용은 섹션을 참조하세요 [서버 엔드포인트에 대한 사용자 관리](#).

3단계: 서비스 관리 사용자 추가

SFTP 지원 서버에 사용자를 추가하려면

1. 서버 페이지에서 사용자를 추가할 서버를 선택합니다.
2. 사용자 추가를 선택합니다.
3. 사용자 구성 섹션의 사용자 이름에 사용자 이름을 입력합니다. 이 사용자 이름은 3~100자여야 합니다. 사용자 이름에는 a~z, A~Z, 0~9, 밑줄 '_', 하이픈 '-', 마침표 '.' 및 at 기호 '@' 문자를 사용할 수 있습니다. 사용자 이름은 하이픈 '-', 마침표 '.' 또는 at 기호 '@'로 시작할 수 없습니다.
4. 액세스에서에서 생성한 IAM 역할을 선택합니다 [IAM 역할 및 정책 생성](#). 이 IAM 역할에는 Amazon S3 버킷에 액세스할 수 있는 권한과 AWS Transfer Family 서비스와의 신뢰 관계가 포함된 IAM 정책이 포함됩니다. 예 설명된 절차는 적절한 신뢰 관계를 설정하는 방법을 [신뢰 관계를 구축하기 위해](#) 보여줍니다.
5. 정책에서 없음을 선택합니다.
6. 홈 디렉터리에서 전송할 데이터를 저장할 Amazon S3 버킷을 선택합니다 AWS Transfer Family. home 디렉터리의 경로를 입력합니다. 사용자가 클라이언트를 사용하여 로그인할 때 표시되는 디렉터리입니다.

세션 정책을 사용할 수 있는 옵션이 있도록 사용자 이름이 포함된 디렉터리 경로를 사용하는 것이 좋습니다. 세션 정책은 Amazon S3 버킷의 사용자 액세스를 해당 사용자의 home 디렉터리로 제한합니다. 세션 정책 사용에 대한 자세한 내용은 [섹션을 참조하세요](#) [세션 정책의 작동 방식](#).

원하는 경우 이 파라미터를 비워 두고 Amazon S3 버킷의 root 디렉터를 사용할 수 있습니다. 이 옵션을 선택하는 경우 IAM 역할이 디렉터리에 대한 액세스를 제공하는지 root 확인합니다.

7. 제한된 확인란을 선택하여 사용자가 home 디렉터리 외부의 모든 항목에 액세스하지 못하도록 합니다. 또한 사용자가 Amazon S3 버킷 이름 또는 폴더 이름을 볼 수 없습니다.
8. SSH 퍼블릭 키의 경우 SSH 키 페어의 퍼블릭 SSH 키 부분을 `ssh-rsa <string>` 형식으로 입력합니다.

새 사용자를 추가하려면 먼저 서비스에서 키를 검증해야 합니다. SSH 키 페어를 생성하는 방법에 대한 자세한 내용은 [섹션을 참조하세요](#) [서비스 관리 사용자를 위한 SSH 키 생성](#).

9. (옵션) 키와 값에 하나 이상의 태그를 키-값 쌍으로 입력하고 태그 추가를 선택합니다.
10. 추가를 선택해 새 사용자를 원하는 서버에 추가합니다.

새 사용자는 서버 세부 정보 페이지의 사용자 섹션에 나타납니다.

4단계: 클라이언트를 사용하여 파일 전송

클라이언트에서 전송 작업을 지정하여 AWS Transfer Family 서비스를 통해 파일을 전송합니다.는 여러 클라이언트를 AWS Transfer Family 지원합니다. 자세한 내용은 [클라이언트를 사용하여 서버 엔드 포인트를 통한 파일 전송](#)을 참조하세요.

이 섹션에는 Cyberduck과 OpenSSH를 사용하기 위한 절차가 수록되어 있습니다.

주제

- [Cyberduck 사용하기](#)
- [OpenSSH 사용](#)

Cyberduck 사용하기

Cyberduck을 AWS Transfer Family 사용하여를 통해 파일을 전송하려면

1. [Cyberduck](#) 클라이언트를 엽니다.
2. 연결 열기를 선택합니다.

3. 연결 열기 대화 상자에서 SFTP(SSH File Transfer 프로토콜)를 선택합니다.
4. 서버에 서버 엔드포인트를 입력합니다. 서버 엔드포인트는 서버 세부 정보 페이지에 있습니다. [SFTP, FTPS 및 FTP 서버 세부 정보 보기](#)를 참조하세요.
5. 포트 번호에는 SFTP에 **22**를 입력합니다.
6. 사용자 이름에 [서버 엔드포인트에 대한 사용자 관리](#)에서 생성한 사용자 이름을 입력합니다.
7. SSH 프라이빗 키에는 SSH 프라이빗 키를 선택 또는 입력합니다.
8. 연결을 선택합니다.
9. 파일 전송을 실행합니다.

파일 위치에 따라, 다음 중 하나를 수행하세요.

- 로컬 디렉터리(소스)에서 전송할 파일을 선택하고, Amazon S3 디렉터리(대상)로 끌어다 놓습니다.
- Amazon S3 디렉터리(소스)에서 전송할 파일을 선택하고, 로컬 디렉터리(대상)로 끌어다 놓습니다.

OpenSSH 사용

아래의 지침을 이용해 OpenSSH를 이용하는 명령줄로 파일을 전송하세요.

Note

이 클라이언트는 SFTP 지원 서버에서만 작동합니다.

OpenSSH 명령줄 유틸리티를 AWS Transfer Family 사용하여를 통해 파일을 전송하려면

1. Linux나 Macintosh에서 명령 터미널을 엽니다.
2. 프롬프트에서 다음 명령을 입력합니다. `% sftp -i transfer-key sftp_user@service_endpoint`

앞의 명령에서 `sftp_user`는 사용자 이름이고 `transfer-key`는 SSH 프라이빗 키입니다. 여기서 `service_endpoint`는 선택한 서버의 AWS Transfer Family 콘솔에 표시된 서버의 엔드포인트입니다.

`sftp` 프롬프트가 나타날 것입니다.

3. (옵션) 사용자의 홈 디렉터리를 보려면 `sftp` 프롬프트에 다음 명령을 입력합니다. `sftp> pwd`
4. 다음 줄에 다음 텍스트를 입력합니다. `sftp> cd /amzn-s3-demo-bucket/home/sftp_user`

본 시작하기 실습에서는 이 Amazon S3 버킷이 파일 전송 대상입니다.

5. 다음 줄에 다음 명령을 입력합니다. `sftp> put filename.txt`

`put` 명령은 파일을 Amazon S3 버킷으로 전송합니다.

파일 전송이 진행 중이거나 완료되었음을 나타내는, 다음과 비슷한 메시지가 표시됩니다.

```
Uploading filename.txt to /amzn-s3-demo-bucket/home/sftp_user/
filename.txt
```

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

파일 복호화를 위한 관리형 워크플로 설정

이 자습서에서는 복호화 단계가 포함된 관리형 워크플로를 설정하는 방법을 설명합니다. 또한 자습서에서는 Amazon S3 버킷에 암호화된 파일을 업로드한 다음 동일한 버킷에서 복호화된 파일을 보는 방법을 보여줍니다.

Note

AWS 스토리지 블로그에는 Transfer Family 관리형 워크플로, [PGP 및를 사용한 암호화 및 암호화 해제를 사용하여 코드를 작성하지 않고 파일을 간단하게 해독하는 방법을 설명하는 게시물이 있습니다 AWS Transfer Family](#).

주제

- [1단계: 실행 역할 구성](#)
- [2단계: 관리형 워크플로 생성](#)
- [3단계: 서버에 워크플로 추가 및 사용자 생성](#)
- [4단계: PGP 키 쌍 생성](#)
- [5단계:에 PGP 프라이빗 키 저장 AWS Secrets Manager](#)
- [6단계: 파일 암호화](#)

- [7단계: 워크플로 실행 및 결과 보기](#)

1단계: 실행 역할 구성

Transfer Family가 워크플로를 시작하는 데 사용할 수 있는 AWS Identity and Access Management (IAM) 실행 역할을 생성합니다. 실행 역할을 생성하는 프로세스는 [워크플로에 대한 IAM 정책](#)에 설명되어 있습니다.

Note

[신뢰 관계를 구축하기 위해](#)에 설명된 대로 실행 역할을 생성하는 과정에서 실행 역할과 Transfer Family 사이에 신뢰 관계를 확실히 구축해야 합니다.

다음 실행 역할 정책에는 이 자습서에서 생성한 워크플로를 시작하는 데 필요한 모든 권한이 포함되어 있습니다. 이 정책 예를 사용하려면 *user input placeholders*를 실제 정보로 바꾸세요. `amzn-s3-demo-bucket`를 암호화된 파일을 업로드하는 Amazon S3 버킷의 이름으로 바꿉니다.

Note

모든 워크플로에 이 예에 나열된 모든 권한이 필요한 것은 아닙니다. 특정 워크플로의 단계 유형에 따라 권한을 제한할 수 있습니다. 사전 정의된 각 단계 타입에 필요한 권한은 [사전 정의된 단계 사용](#)에 설명되어 있습니다. 사용자 지정 단계에 필요한 권한은 [사용자 지정 단계 IAM 권한](#)에 설명되어 있습니다.

```
{
  "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:::amzn-s3-demo-bucket/*",
        "arn:aws:s3:::amzn-s3-demo-bucket"
    ]
},
{
    "Sid": "DecryptSecret",
    "Effect": "Allow",
    "Action": [
        "secretsmanager:GetSecretValue"
    ],
    "Resource": "arn:aws:secretsmanager:us-east-1:123456789012:secret:aws/transfer/*"
}
]
}

```

2단계: 관리형 워크플로 생성

이제 복호화 단계가 포함된 워크플로를 만들어야 합니다.

복호화 단계가 포함된 워크플로를 만들려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 워크플로를 선택한 다음, 워크플로 생성을 선택합니다.
3. 다음 세부 정보를 입력합니다.
 - 설명을 입력합니다. 예를 들어, **Decrypt workflow example**.
 - 공칭 단계 섹션에서 단계 추가를 선택합니다.
4. 단계 타입 선택에서 파일 복호화를 선택한 후 다음을 선택합니다.
5. 파라미터 구성 대화 상자에서 다음 사항을 지정합니다.
 - 설명이 포함된 단계 이름을 입력합니다. 예: **decrypt-step**. 단계 이름에는 공백은 허용되지 않습니다.

- 복호화된 파일의 대상으로 Amazon S3을 선택합니다.
- 대상 버킷 이름으로는 1단계에서 생성한 IAM 정책에서 amzn-s3-demo-bucket로 지정한 것과 동일한 Amazon S3 버킷을 선택합니다.
- 대상 키 접두사에는 대상 버킷에 복호화된 파일을 저장할 접두사(폴더)의 이름을 입력합니다.
예: **decrypted-files/**.

 Note

접두사에는 반드시 후행 슬래시(/)를 추가해야 합니다.

- 이 자습서에서는 기존 항목 덮어쓰기를 선택 해제해 두세요. 이 설정을 지우면 기존 파일과 이름이 같은 파일을 해독하려고 하면 워크플로 처리가 중지되고 새 파일은 처리되지 않습니다.

다음을 선택하여 검토 화면으로 이동합니다.

6. 해당 단계의 세부 정보를 검토합니다. 모든 내용이 정확하면 단계 생성을 선택합니다.
7. 워크플로에는 단일 복호화 단계만 필요하므로 추가로 구성할 단계가 없습니다. 워크플로 생성을 선택하여 새 워크플로를 생성합니다.

새 워크플로의 워크플로 ID를 기록해 둡니다. 다음 단계에서 이 ID가 필요합니다. 이 자습서에서는 *w-1234abcd5678efghi*를 예 워크플로 ID로 사용합니다.

3단계: 서버에 워크플로 추가 및 사용자 생성

이제 복호화 단계가 포함된 워크플로를 Transfer Family 서버에 연결해야 합니다. 이 자습서에서는 기존 Transfer Family 서버에 워크플로를 연결하는 방법을 보여줍니다. 또는 워크플로에 사용할 새 서버를 만들 수 있습니다.

워크플로를 서버에 연결한 후에는 SFTP를 통해 서버에 연결하고 워크플로가 실행되도록 트리거할 수 있는 사용자를 만들어야 합니다.

워크플로를 실행하도록 Transfer Family 서버를 구성하려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 서버를 선택한 다음 목록에서 서버 하나를 선택합니다. 이 서버가 SFTP 프로토콜을 지원하는지 확인하세요.
3. 서버의 세부 정보 페이지에서 추가 세부 정보 섹션까지 아래로 스크롤한 다음 편집을 선택합니다.

4. 추가 세부 정보 편집 페이지의 관리형 워크플로 섹션에서 워크플로를 선택하고 해당하는 실행 역할을 선택합니다.
 - 전체 파일 업로드를 위한 워크플로의 경우 [2단계: 관리형 워크플로 생성](#)에서 만든 워크플로를 선택합니다. 예: **w-1234abcd5678efghi**.
 - 관리형 워크플로 실행 역할의 경우 [1단계: 실행 역할 구성](#)에서 생성한 IAM 역할을 선택합니다.
5. 페이지 맨 아래로 스크롤하고 저장을 선택하여 변경 내용을 저장합니다.

사용 중인 서버의 ID를 기록해 둡니다. PGP 키를 저장하는 데 사용하는 AWS Secrets Manager 보안 암호의 이름은 부분적으로 서버 ID를 기반으로 합니다.

워크플로를 트리거할 수 있는 사용자를 추가하려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 서버를 선택한 다음 복호화 워크플로에 사용할 서버를 선택합니다.
3. 서버 세부 정보 페이지에서 사용자 섹션으로 스크롤하여 사용자 추가를 선택합니다.
4. 새 사용자의 경우 다음 세부 정보를 입력합니다.
 - 사용자 이름에 **decrypt-user**를 입력합니다.
 - 역할에서 서버에 액세스할 수 있는 사용자 역할을 선택합니다.
 - 홈 디렉터리의 경우 이전에 사용한 Amazon S3 버킷을 선택합니다. 예: **amzn-s3-demo-bucket**.
 - SSH 퍼블릭 키의 경우 보유한 프라이빗 키에 해당하는 퍼블릭 키를 붙여넣습니다. 자세한 내용은 [서비스 관리 사용자를 위한 SSH 키 생성](#)을 참조하세요.
5. 추가를 선택하여 새 사용자를 저장합니다.

이 서버의 Transfer Family 사용자 이름을 기록해 둡니다. 암호는 부분적으로 사용자 이름을 기반으로 합니다. 단순화를 위해 이 자습서에서는 서버의 모든 사용자가 사용할 수 있는 기본 암호를 사용합니다.

4단계: PGP 키 쌍 생성

[지원되는 PGP 클라이언트](#) 중 하나를 사용하여 PGP키 쌍을 생성합니다. 이 과정은 [PGP 키 생성](#)에 자세히 설명되어 있습니다.

PGP 키 쌍 생성

1. 이 자습서에서는 gpg(GnuPG) 버전 2.0.22 클라이언트를 사용하여 RSA를 암호화 알고리즘으로 사용하는 PGP 키 쌍을 생성할 수 있습니다. 이 클라이언트의 경우 다음 명령을 실행하고 이메일 주소와 암호를 입력합니다. 원하는 이름이나 이메일 주소를 사용할 수 있습니다. 자습서 뒷부분에서 값을 입력해야 하므로 사용하는 값을 기억해 두세요.

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

Note

GnuPG 버전 2.3.0 이상을 사용하는 경우 `gpg --full-gen-key`를 실행해야 합니다. 생성할 키 타입을 묻는 메시지가 표시되면 RSA 또는 ECC를 선택합니다. ECC를 선택하는 경우 타원 곡선에 Curve25519 대해 NIST BrainPool 및 중에서 선택할 수 있습니다.

2. 다음 명령을 실행하여 프라이빗 키를 내보냅니다. `user@example.com`을 키를 생성할 때 사용한 이메일 주소로 바꿉니다.

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

이 명령은 프라이빗 키를 **workflow-tutorial-key.pgp** 파일로 내보냅니다. 출력 파일의 이름을 원하는 대로 지정할 수 있습니다. 프라이빗 키를 AWS Secrets Manager에 추가한 후 프라이빗 키 파일을 삭제할 수도 있습니다.

5단계:에 PGP 프라이빗 키 저장 AWS Secrets Manager

매우 구체적인 방식으로 Secrets Manager에 프라이빗 키를 저장해야 워크플로가 업로드된 파일에 대한 복호화 단계를 실행할 때 워크플로가 프라이빗 키를 찾을 수 있습니다.

Note

Secrets Manager에 암호를 저장하면에 요금이 AWS 계정 발생합니다. 요금에 대한 자세한 내용은 [AWS Secrets Manager 요금](#)을 참조하세요.

Secrets Manager에 PGP 프라이빗 키를 저장하려면

1. 에 로그인 AWS Management Console 하고 <https://console.aws.amazon.com/secretsmanager/> AWS Secrets Manager 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 암호를 선택합니다.
3. 암호 페이지에서 새 암호 저장을 선택합니다.
4. 암호 선택 페이지의 암호 타입에서 다른 타입의 암호를 선택합니다.
5. 키/값 쌍 섹션에서 키/값 탭을 선택합니다.
 - 키 — **PGPPrivateKey**를 입력합니다.
 - 값 — 프라이빗 키의 텍스트를 값 필드에 붙여넣습니다.
6. 행 추가를 선택하고 키/값 쌍 섹션에서 키/값 탭을 선택합니다.
 - 키 — **PGPPassphrase**를 입력합니다.
 - 값 — [4단계: PGP 키 쌍 생성](#)에서 PGP 키 쌍을 생성할 때 사용한 암호를 입력합니다.
7. 다음을 선택합니다.
8. 암호 구성 페이지에 암호를 위한 명칭과 설명을 입력합니다. 특정 사용자 또는 모든 사용자가 사용할 수 있는 보안 암호를 생성할 수 있습니다. 서버 ID가 인 경우 다음과 같이 보안 암호의 이름을 **s-11112222333344445** 지정합니다.
 - 모든 사용자의 기본 보안 암호를 생성하려면 보안 암호의 이름을 로 지정합니다 **aws/transfer/s-11112222333344445/@pgp-default**.
 - 이전에 생성한 사용자에게만 암호만 생성하려면 암호 이름을 **aws/transfer/s-11112222333344445/decrypt-user**로 지정합니다.
9. 다음을 선택한 후 교체 구성 페이지의 기본값을 그대로 사용합니다. 그 다음 다음을 선택합니다.
10. 검토 페이지에서 저장을 선택하여 암호를 만들고 저장합니다.

Secrets Manager에 PGP 프라이빗 키를 추가하는 방법에 대한 자세한 내용은 [PGP 키를 저장하는 AWS Secrets Manager 데 사용을 참조하세요](#).

6단계: 파일 암호화

gpg 프로그램을 사용하여 워크플로에 사용할 파일을 암호화합니다. 파일을 암호화하려면 다음 명령을 실행합니다.

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

이 자동화를 실행하기 전에 다음 사항에 유의하세요.

- `-r` 인수의 경우 *marymajor@example.com*을 PGP 키 쌍을 만들 때 사용한 이메일 주소로 바꿉니다.
- `--openpgp` 플래그는 옵션입니다. 이 플래그는 암호화된 파일이 [OpenPGP RFC4880](#) 표준을 준수하도록 합니다.
- 이 명령은 **testfile.txt**와 같은 위치에 이름이 **testfile.txt.gpg**라고 지정된 파일을 만듭니다.

Important

AWS Transfer Family 워크플로에 사용할 파일을 암호화할 때는 항상 `-r` 파라미터를 사용하여 비익명 수신자를 지정해야 합니다. 익명 암호화(수신자를 지정하지 않음)는 시스템이 복호화에 사용할 키를 식별할 수 없기 때문에 워크플로에서 복호화 실패를 일으킬 수 있습니다. 이 문제에 대한 디버깅 정보에서 확인할 수 있습니다 [익명 수신자 암호화 문제 해결](#).

7단계: 워크플로 실행 및 결과 보기

워크플로를 실행하려면 3단계에서 만든 사용자를 사용하여 Transfer Family 서버에 연결합니다. 그런 다음 [2.5단계에서 지정한 Amazon S3 버킷을 살펴보고 복호화된 파일을 볼 수 있도록 대상 파라미터를 구성](#)할 수 있습니다.

복호화 워크플로를 실행하려면

1. 명령 터미널을 엽니다.
2. 실제 엔드포인트를 *your-endpoint*로 대체하고 *transfer-key*를 사용자의 SSH 프라이빗 키로 대체하여 다음 명령을 실행합니다.

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

예를 들어 프라이빗 키가 `~/.ssh/decrypt-user`에 저장되어 있고 엔드포인트가 `s-11112222333344445.server.transfer.us-east-2.amazonaws.com`인 경우, 명령은 다음과 같습니다.

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

3. `pwd` 명령을 실행합니다. 이 명령이 제대로 실행되면 다음과 같은 응답을 반환합니다.

```
Remote working directory: /amzn-s3-demo-bucket/decrypt-user
```

디렉터리에는 Amazon S3 버킷 이름이 반영됩니다.

4. 다음 명령을 실행하여 파일을 업로드하고 워크플로를 실행하도록 트리거합니다.

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

5. 복호화된 파일의 대상 폴더는 워크플로를 만들 때 지정한 `decrypted-files/` 폴더입니다. 이제 해당 폴더로 이동하여 내용을 나열할 수 있습니다.

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

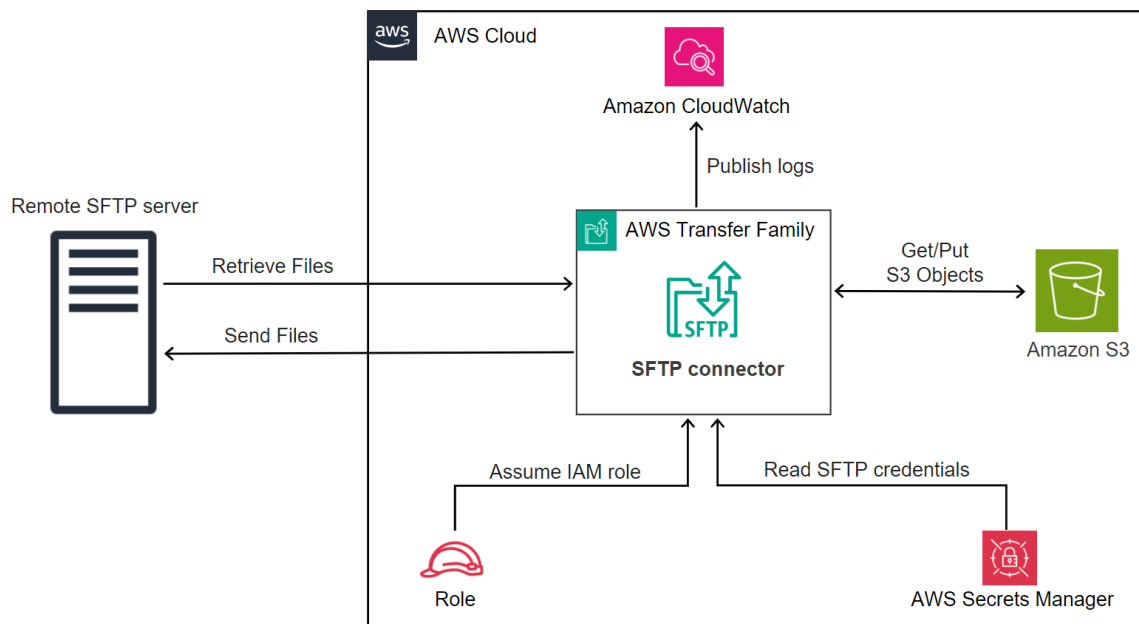
성공하면 `ls` 명령이 `testfile.txt` 파일을 나열합니다. 이 파일을 다운로드하여 이전에 암호화한 원본 파일과 동일한지 확인할 수 있습니다.

SFTP 커넥터 설정 및 사용

커넥터의 목적은 AWS 스토리지와 파트너의 SFTP 서버 간에 관계를 설정하는 것입니다. Amazon S3에서 파트너 소유의 외부 대상으로 파일을 보낼 수 있습니다. SFTP 커넥터를 사용하여 파트너의 SFTP 서버에서 파일을 검색할 수도 있습니다.

이 자습서에서는 서비스 관리형 송신 유형과 VPC_LATTICE 송신 유형으로 SFTP 커넥터를 설정한 다음 Amazon S3 스토리지와 SFTP 서버 간에 파일을 전송하는 방법을 보여줍니다.

SFTP 커넥터는에서 SFTP 자격 증명을 검색 AWS Secrets Manager 하여 원격 SFTP 서버에 인증하고 연결을 설정합니다. 커넥터는 원격 서버로 파일을 보내거나 원격 서버에서 파일을 검색하고 Amazon S3에 파일을 저장합니다. 서비스 관리형 송신(AWS 관리형 인프라 사용) 또는 VPC 송신(VPC 간 리소스 액세스를 사용하여 VPC를 통해 라우팅) 중에서 선택할 수 있습니다. IAM 역할은 Amazon S3 버킷 및 Secrets Manager에 저장된 자격 증명에 대한 액세스를 허용하는 데 사용됩니다. 또한 Amazon CloudWatch에 로그인할 수 있습니다.



다음 블로그 게시물은 SFTP 커넥터를 사용하여 원격 SFTP 서버로 전송하기 전에 PGP를 사용한 파일 암호화를 포함하여 [AWS Transfer Family SFTP 커넥터를 사용하여 MFT 워크플로를 빌드하기 위한 참조 아키텍처를 제공합니다. SFTP 커넥터 및 PGP 암호화를 사용하여 안전하고 규정을 준수하는 관리형 파일 전송 설계.](#)

커넥터 송신 유형

SFTP 커넥터는 커넥터가 트래픽을 원격 SFTP 서버로 라우팅하는 방법을 결정하는 두 가지 송신 유형을 지원합니다.

- **SERVICE_MANAGED(기본값):** 아웃바운드 연결에 AWS 고정 IP 주소가 있는 Transfer Family 관리형 인프라를 사용합니다.
- **VPC:** VPC 간 리소스 액세스를 사용하여 VPC를 통해 트래픽을 라우팅하여 프라이빗 엔드포인트 연결 및 자체 NAT 게이트웨이 사용을 활성화합니다.

이 자습서에서는 두 송신 유형을 모두 다룹니다. 필요한 경우 VPC 송신 유형을 선택합니다.

- VPC의 프라이빗 SFTP 서버에 연결(프라이빗 IP 주소)
- Direct Connect 또는 VPN을 통해 온프레미스 SFTP 서버에 연결
- 보안 제어를 위해 VPC를 통해 퍼블릭 엔드포인트 트래픽 라우팅
- 아웃바운드 연결에 자체 탄력적 IP 주소 사용

주제

- [1단계: 필요한 지원 리소스 생성](#)
- [2단계: SFTP 커넥터 생성 및 테스트](#)
- [3단계: SFTP 커넥터를 사용하여 파일 전송 및 검색](#)
- [원격 SFTP 서버로 사용할 Transfer Family 서버를 생성하는 절차](#)

1단계: 필요한 지원 리소스 생성

SFTP 커넥터를 사용하여 Amazon S3와 원격 SFTP 서버 간에 파일을 복사할 수 있습니다. 이 자습서에서는 AWS Transfer Family 서버를 원격 SFTP 서버로 사용합니다. 다음 리소스를 생성하고 구성해야 합니다.

- Amazon S3 버킷을 생성하여 AWS 환경에 파일을 저장하고 원격 SFTP 서버에서 파일을 전송하고 검색합니다 [Amazon S3 버킷 생성](#).
- Secrets Manager에서 Amazon S3 스토리지 및 보안 암호에 액세스하기 위한 AWS Identity and Access Management 역할을 생성합니다 [필요한 권한을 가진 IAM 역할 생성](#).
- SFTP 프로토콜을 사용하는 Transfer Family 서버와 SFTP 커넥터를 사용하여 SFTP 서버로 또는 SFTP 서버에서 파일을 전송하는 서비스 관리형 사용자를 생성합니다 [Transfer Family SFTP 서버 및 사용자 생성](#).
- SFTP 커넥터가 원격 SFTP 서버에 로그인하는 데 사용하는 자격 증명을 저장하는 AWS Secrets Manager 보안 암호를 생성합니다 [에서 보안 암호 생성 및 저장 AWS Secrets Manager](#).

VPC 송신 유형 커넥터의 경우 다음 사항도 필요합니다.

- 적절한 서브넷 및 보안 그룹이 있는 VPC
- Resource Gateway(최소 2개의 가용 영역 필요): [Resource Gateway 생성\(VPC 송신 유형만 해당\)](#).
- SFTP 서버를 가리키는 리소스 구성: [리소스 구성 생성\(VPC 송신 유형만 해당\)](#). 자세한 내용은 Amazon VPC Lattice 사용 설명서의 [Resource configurations](#) 섹션을 참조하시기 바랍니다.

Amazon S3 버킷 생성

Amazon S3 버킷을 생성하려면

1. <https://console.aws.amazon.com/s3/> AWS Transfer Family 콘솔에 로그인합니다.
2. 리전을 선택하고 이름을 입력합니다.

이 자습서에서는 버킷이 **US East (N. Virginia) us-east-1** 있으며 이름은 **sftp-server-storage-east**.

3. 기본값을 수락하고 버킷 생성을 선택합니다.

Amazon S3 버킷 생성에 대한 자세한 내용은 [S3 버킷을 생성하려면 어떻게 해야 하나요?](#)를 참조하세요. Amazon Simple Storage Service 사용 설명서의 .

필요한 권한을 가진 IAM 역할 생성

액세스 역할의 경우 다음 권한이 있는 정책을 생성합니다.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowListingOfUserFolder",
      "Action": [
        "s3:ListBucket",
        "s3:GetBucketLocation"
      ],
      "Effect": "Allow",
      "Resource": [
        "arn:aws:s3:::amzn-s3-demo-bucket"
      ]
    },
    {
      "Sid": "HomeDirObjectAccess",
      "Effect": "Allow",
      "Action": [
        "s3:PutObject",
        "s3:GetObject",
        "s3:DeleteObject",
        "s3:DeleteObjectVersion",
        "s3:GetObjectVersion",
        "s3:GetObjectACL",
        "s3:PutObjectACL"
      ],
      "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/*"
    }
  ]
}
```

```

    },
    {
      "Sid": "GetConnectorSecretValue",
      "Effect": "Allow",
      "Action": [
        "secretsmanager:GetSecretValue"
      ],
      "Resource": "arn:aws:secretsmanager:us-west-2:111122223333:secret:aws/transfer/SecretName-6RandomCharacters"
    }
  ]
}

```

다음과 같이 항목을 바꿉니다.

- *amzn-s3-demo-bucket*의 경우 자습서에서는를 사용합니다**sftp-server-storage-east**.
- *##*의 경우 자습서에서는를 사용합니다**us-east-1**.
- *account-id*의 경우 ID를 AWS 계정 사용합니다.
- *SecretName-6RandomCharacters*의 경우 이름은 입니다(보안 암호**using sftp-connector1**에 대해 6개의 무작위 문자가 있음).

또한이 역할에 사용자의 전송 요청을 처리할 때 커넥터가 리소스에 액세스할 수 있도록 허용하는 신뢰 관계가 포함되어 있는지 확인해야 합니다. 신뢰 관계 설정에 대한 자세한 내용은 [신뢰 관계를 구축하기 위해](#)을 참조하세요.

Note

자습서에서 사용 중인 역할에 대한 세부 정보를 보려면 섹션을 참조하세요 [사용자 및 액세스 역할 결합](#).

에서 보안 암호 생성 및 저장 AWS Secrets Manager

SFTP 커넥터의 사용자 자격 증명을 저장하려면 Secrets Manager에 암호를 저장해야 합니다. 암호, SSH 프라이빗 키 또는 둘 다를 사용할 수 있습니다. 자습서에서는 프라이빗 키를 사용합니다.

Note

Secrets Manager에 암호를 저장하면에 요금이 AWS 계정 발생합니다. 요금에 대한 자세한 내용은 [AWS Secrets Manager 요금](#)을 참조하세요.

보안 암호를 저장하는 절차를 시작하기 전에 프라이빗 키를 검색하고 형식을 지정합니다. 프라이빗 키는 원격 SFTP 서버의 사용자에게 대해 구성된 퍼블릭 키와 일치해야 합니다. 자습서의 경우 프라이빗 키는 원격 서버로 사용 중인 Transfer Family SFTP 서버의 테스트 사용자에게 대해 저장된 퍼블릭 키와 일치해야 합니다.

이렇게 하려면 다음 명령을 실행하세요.

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

예를 들어 프라이빗 키 파일이 있는 경우 `~/.ssh/sftp-testuser-privatekey` 명령은 다음과 같습니다.

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

그러면 키가 올바른 형식(줄 바꿈 문자 포함)으로 표준 출력으로 출력됩니다. 다음 절차(6단계)에 붙여넣어야 하므로이 텍스트를 어딘가에 복사합니다.

SFTP 커넥터의 사용자 자격 증명을 Secrets Manager에 저장하려면

1. 에 로그인 AWS Management Console 하고 <https://console.aws.amazon.com/secretsmanager/> AWS Secrets Manager 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 암호를 선택합니다.
3. 암호 페이지에서 새 암호 저장을 선택합니다.
4. 암호 선택 페이지의 암호 타입에서 다른 타입의 암호를 선택합니다.
5. 키/값 쌍 섹션에서 키/값 탭을 선택합니다.
 - 키 -를 입력합니다 **Username**.
 - value - 사용자 이름을 입력합니다 **sftp-testuser**.
6. 키를 입력하려면 일반 텍스트 탭을 사용하는 것이 좋습니다.
 - a. 행 추가를 선택한 다음을 입력합니다 **PrivateKey**.
 - b. 일반 텍스트 탭을 선택합니다. 필드에는 이제 다음 텍스트가 포함됩니다.

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

- c. 빈 큰따옴표("") 사이에 프라이빗 키(이전에 저장됨)의 텍스트를 붙여 넣습니다.

화면은 다음과 같아야 합니다(키 데이터는 회색으로 표시됨).



7. 다음을 선택합니다.
8. 보안 암호 구성 페이지에서 보안 암호의 이름을 입력합니다. 이 자습서에서는 보안 암호의 이름을 **aws/transfer/sftp-connector1**로 지정합니다.
9. 다음을 선택한 후 교체 구성 페이지의 기본값을 그대로 사용합니다. 그 다음 다음을 선택합니다.
10. 검토 페이지에서 저장을 선택하여 암호를 만들고 저장합니다.

Resource Gateway 생성(VPC 송신 유형만 해당)

VPC 송신 유형 커넥터의 경우 VPC에서 Resource Gateway를 생성해야 합니다. Resource Gateway는 VPC 간 리소스 액세스의 진입점 역할을 합니다.

Resource Gateway를 생성하려면

1. 다음 명령을 실행하여 Resource Gateway를 생성합니다(VPC ID 및 서브넷 IDs를 값으로 대체).

```
aws vpc-lattice create-resource-gateway \
```

```
--name my-sftp-resource-gateway \  
--vpc-identifier vpc-12345678 \  
--subnet-ids subnet-12345678 subnet-87654321
```

Note

Resource Gateway에는 최소 2개의 가용 영역에 서브넷이 필요합니다.

2. 다음 단계에서 사용할 응답의 Resource Gateway ID를 기록해 둡니다.

리소스 구성 생성(VPC 송신 유형만 해당)

SFTP 서버를 가리키는 리소스 구성을 생성합니다. VPC에 있는 서버의 프라이빗 IP 주소 또는 외부 서버의 퍼블릭 DNS 이름일 수 있습니다. 리소스 구성에 대한 자세한 내용은 Amazon VPC Lattice 사용 설명서의 [리소스 구성](#)을 참조하세요.

리소스 구성을 생성하려면

1. 프라이빗 SFTP 서버의 경우 다음을 실행합니다.

```
aws vpc-lattice create-resource-configuration \  
  --name my-sftp-resource-config \  
  --port-ranges 22 \  
  --type SINGLE \  
  --resource-gateway-identifier rgw-12345678 \  
  --resource-configuration-definition ipResource={ipAddress="10.0.1.100"}
```

2. 퍼블릭 SFTP 서버(DNS 이름만 해당)의 경우 다음을 실행합니다.

```
aws vpc-lattice create-resource-configuration \  
  --name my-public-sftp-resource-config \  
  --port-ranges 22 \  
  --type SINGLE \  
  --resource-gateway-identifier rgw-12345678 \  
  --resource-configuration-definition dnsResource={domainName="sftp.example.com"}
```

Note

퍼블릭 엔드포인트는 IP 주소가 아닌 DNS 이름을 사용해야 합니다.

3. 커넥터를 생성할 때 사용할 응답의 리소스 구성 ARN을 기록해 둡니다.

2단계: SFTP 커넥터 생성 및 테스트

이 섹션에서는 앞서 생성한 모든 리소스를 사용하는 SFTP 커넥터를 생성합니다. 자세한 내용은 [SFTP 커넥터 생성](#) 섹션을 참조하세요.

SFTP 커넥터를 생성하려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 SFTP 커넥터를 선택한 다음 SFTP 커넥터 생성을 선택합니다.
3. 송신 유형에서 다음 중 하나를 선택합니다.
 - 서비스 관리형(기본값): 아웃바운드 연결에 고정 IP 주소가 있는 AWS Transfer Family 관리형 인프라를 사용합니다.
 - VPC Lattice: VPC 간 리소스 액세스를 사용하여 VPC를 통해 트래픽을 라우팅합니다. 프라이빗 엔드포인트 연결 또는 자체 NAT 게이트웨이를 사용하려면이 옵션을 선택합니다.

Important

커넥터를 생성한 후에는 송신 유형을 변경할 수 없습니다. 연결 요구 사항에 따라 신중하게 선택합니다.

4. 커넥터 구성 섹션에서 다음 정보를 제공합니다:
 - URL에 원격 SFTP 서버의 URL을 입력합니다. 자습서에서는 원격 SFTP 서버로 사용 중인 Transfer Family 서버의 URL을 입력합니다.

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

1111aaaa2222bbbb3을 Transfer Family 서버 ID로 바꿉니다.

- 액세스 역할에 앞서 생성한 역할인을 입력합니다 **sftp-connector-role**.
- 리소스 구성 ARN(VPC Lattice 송신 유형만 해당)에 앞서 생성한 리소스 구성의 ARN을 입력합니다.

```
arn:aws:vpc-lattice:us-east-1:account-id:resourceconfiguration/rcfg-12345678
```

- 로깅 역할의 경우 보안 주체 요소에 있는 신뢰 정책 `transfer.amazonaws.com`이 포함된 역할을 선택합니다.

팁: Transfer Family를 신뢰할 수 있는 엔터티로 추가하는 것 외에도 역할에 `AWSTransferLoggingAccess` AWS 관리형 정책을 추가할 수 있습니다. 이 정책은 [AWSTransferLoggingAccess](#)에 자세히 설명되어 있습니다.

Connector configuration

Egress type [Info](#)
Choose your SFTP connector's egress type: either directly over public internet, or via your VPC environment

☒ **Service managed**
Connect to public endpoints over the internet. The connector will present service-provided static IP addresses to remote server.

☐ **VPC Lattice**
Connect to public or private endpoints through your VPC. The connector will present elastic IP address from your VPC's CIDR range to remote server.

URL
Enter the URL of remote server to which you need to connect using the SFTP connector

 URL of the remote (target) SFTP server

Access role [Info](#)
IAM Role for Amazon S3 access and AWS Secrets Manager access
 [Refresh](#)

Logging role - optional [Info](#)
IAM role for the connector to push events to your CloudWatch logs
 [Refresh](#)

5. SFTP 구성 섹션에 다음 정보를 제공합니다.

- 커넥터 자격 증명에서 SFTP 자격 증명이 포함된 Secrets Manager 리소스의 이름을 선택합니다. 자습서에서 선택합니다 `aws/transfer/sftp-connector1`.
- 신뢰할 수 있는 호스트 키의 경우 호스트 키의 퍼블릭 부분에 붙여 넣습니다. SFTP 서버에 `ssh-keyscan` 대해를 실행하여 이 키를 검색할 수 있습니다. 신뢰할 수 있는 호스트 키를 포맷하고 저장하는 방법에 대한 자세한 내용은 [SftpConnectorConfig](#) 데이터 형식 설명서를 참조하세요.
- 최대 동시 연결에서 1~5의 정수 값을 선택합니다. 기본값은 5입니다.

SFTP configuration [Info](#)

Connector credentials [Info](#)
 Select a secret from AWS Secrets Manager containing the username and SSH private key, or username and password, to be used to connect to the remote server

Trusted host keys - optional [Info](#)
 Enter the public portion of the host key(s) that is used to identify the remote server you need to connect to. You can enter the trusted host key(s) now, or add them after creating the connector by using the host key information returned by the TestConnection action.

Connections to the remote server are established only if it's SSH fingerprint matches one of the provided trusted host key(s).

Maximum concurrent connections [Info](#)
 Select the maximum number of concurrent connections that your connector can create to the remote server

6. 모든 설정을 확인한 후 커넥터 생성을 선택하여 SFTP 커넥터를 생성합니다.

를 사용하여 커넥터를 생성할 수도 있습니다 AWS Command Line Interface.

- 서비스 관리형 송신을 사용하여 SFTP 커넥터를 생성하려면 다음 명령을 실행합니다.

```
aws transfer create-connector \
  --url "sftp://s-1111aaaa2222bbbb3.server.transfer.us-east-1.amazonaws.com" \
  --access-role "arn:aws::iam::account-id:role/sftp-connector-role" \
  --sftp-config UserSecretId="aws/transfer/sftp-connector1",TrustedHostKeys="ssh-rsa AAAAB3NzaC..."
```

- VPC 기반 송신을 사용하여 SFTP 커넥터를 생성하려면 다음 명령을 실행합니다.

```
aws transfer create-connector \
  --url "sftp://my.sftp.server.com:22" \
  --access-role "arn:aws::iam::account-id:role/sftp-connector-role" \
  --sftp-config UserSecretId="aws/transfer/sftp-connector1",TrustedHostKeys="ssh-rsa AAAAB3NzaC..." \
  --egress-config VpcLattice={ResourceConfigurationArn="arn:aws:vpc-lattice:us-east-1:account-id:resourceconfiguration/rcfg-12345678",PortNumber=22}
```

SFTP 커넥터를 만든 후에는 새 커넥터를 사용하여 파일을 전송하기 전에 테스트해 보는 것이 좋습니다.

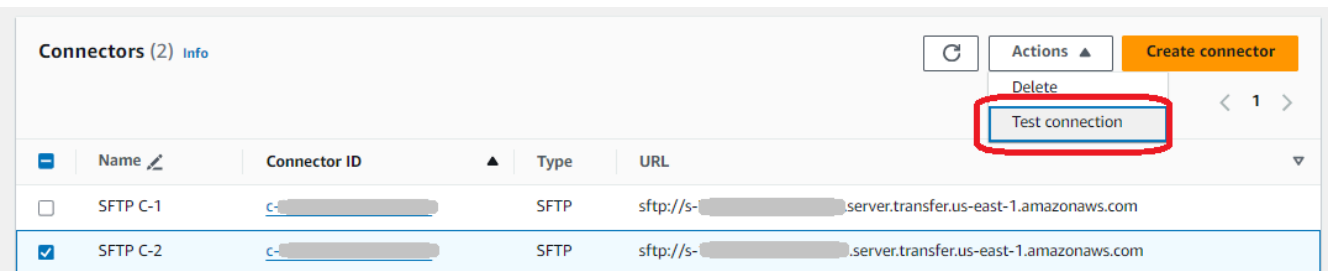
Note

VPC 송신 유형 커넥터의 경우 DNS 확인은 생성 후 몇 분 정도 걸릴 수 있습니다. 이 시간 동안 커넥터 상태는 PENDING 되고 "Connector not available"가 반환됩니다. 파일 전송을 시도하기 ACTIVE 전에 상태가 될 때까지 기다립니다.

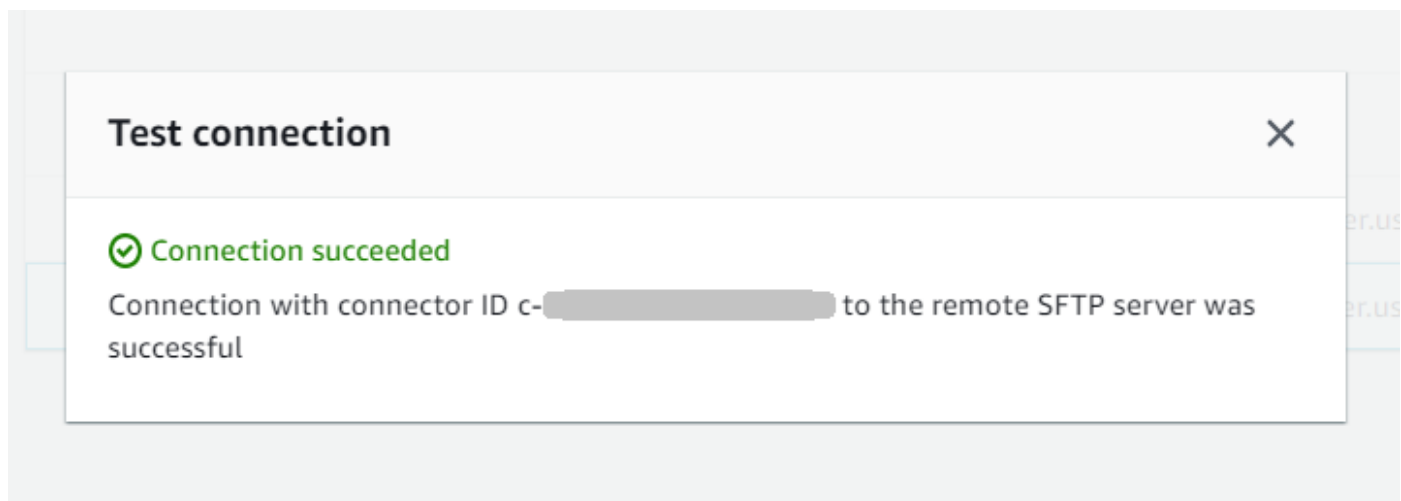
Test a connector using the console

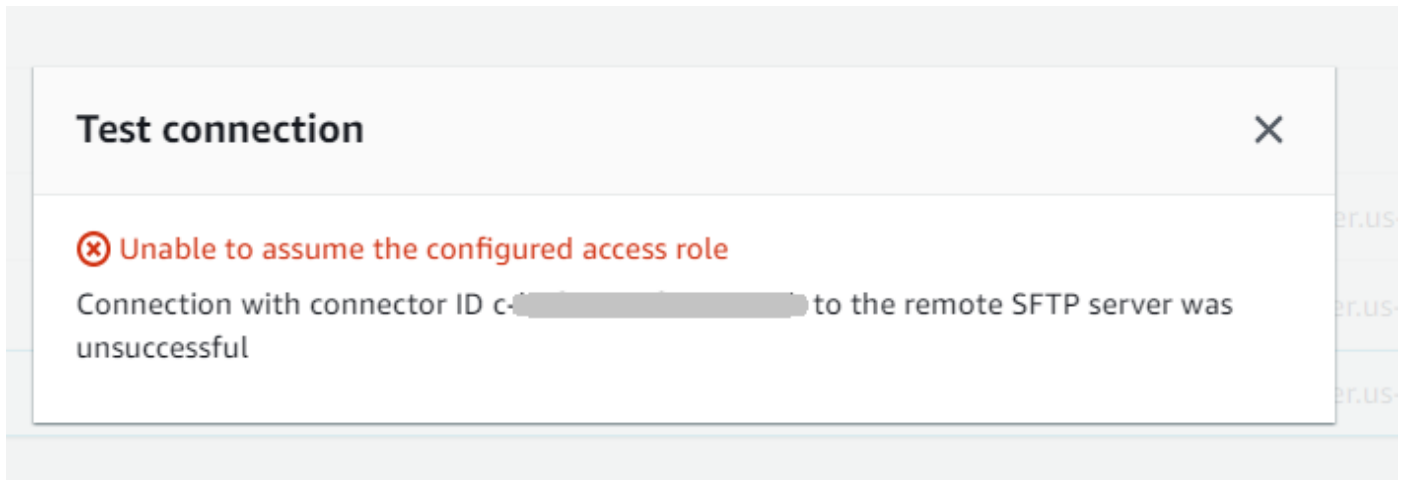
SFTP 커넥터를 테스트하려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 SFTP 커넥터를 선택하고 커넥터를 선택합니다.
3. 작업 메뉴에서 테스트 연계를 선택합니다.



시스템은 테스트의 통과 또는 실패 여부를 나타내는 메시지를 반환합니다. 테스트가 실패하면 시스템은 테스트 실패 이유를 기반으로 오류 메시지를 제공합니다.





Test a connector using the CLI

를 사용하여 커넥터를 테스트하려면 명령 프롬프트에서 다음 명령을 AWS Command Line Interface 실행합니다(connector-*id*를 실제 커넥터 ID로 대체).

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

테스트가 성공하면 다음 줄이 반환됩니다.

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

테스트가 실패하면 다음과 같은 설명이 포함된 오류 메시지가 표시됩니다.

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

VPC 송신 유형 커넥터를 설명하면 응답에 새 필드가 포함됩니다.

```
{
  "Connector": {
    "AccessRole": "arn:aws:iam::219573224423:role/sftp-connector-role",
    "Arn": "arn:aws:transfer:us-east-1:219573224423:connector/c-5dfa309ccabf40759",
    "ConnectorId": "c-5dfa309ccabf40759",

```

```

    "Status": "ACTIVE",
    "EgressConfig": {
      "ResourceConfigurationArn": "arn:aws:vpc-lattice:us-east-1:025066256552:resourceconfiguration/rcfg-079259b27a357a190"
    },
    "EgressType": "VPC",
    "ServiceManagedEgressIpAddresses": null,
    "SftpConfig": {
      "TrustedHostKeys": [ "ssh-rsa AAAAB3NzaC..." ],
      "UserSecretId": "aws/transfer/sftp-connector1"
    },
    "Url": "sftp://my.sftp.server.com:22"
  }
}

```

트래픽ServiceManagedEgressIpAddresses은 AWS 관리형 인프라 대신 VPC를 통해 라우팅되므로 VPC 송신 유형 커넥터의 경우는 null입니다.

3단계: SFTP 커넥터를 사용하여 파일 전송 및 검색

간소화를 위해 Amazon S3 버킷에 파일이 이미 있다고 가정합니다.

Note

이 자습서에서는 소스 및 대상 스토리지 위치 모두에 Amazon S3 버킷을 사용하고 있습니다. SFTP 서버가 Amazon S3 스토리지를 사용하지 않는 경우 다음 명령 `sftp-server-storage-east`에 표시되는 모든 위치에서 경로를 SFTP 서버에서 액세스할 수 있는 파일 위치의 경로로 바꿀 수 있습니다.

- Amazon S3 스토리지 `SEND-to-SERVER.txt`에서 SFTP 서버로 라는 파일을 전송합니다.
- SFTP 서버에서 Amazon S3 스토리지로 `RETRIEVE-to-S3.txt` 라는 파일을 검색합니다.

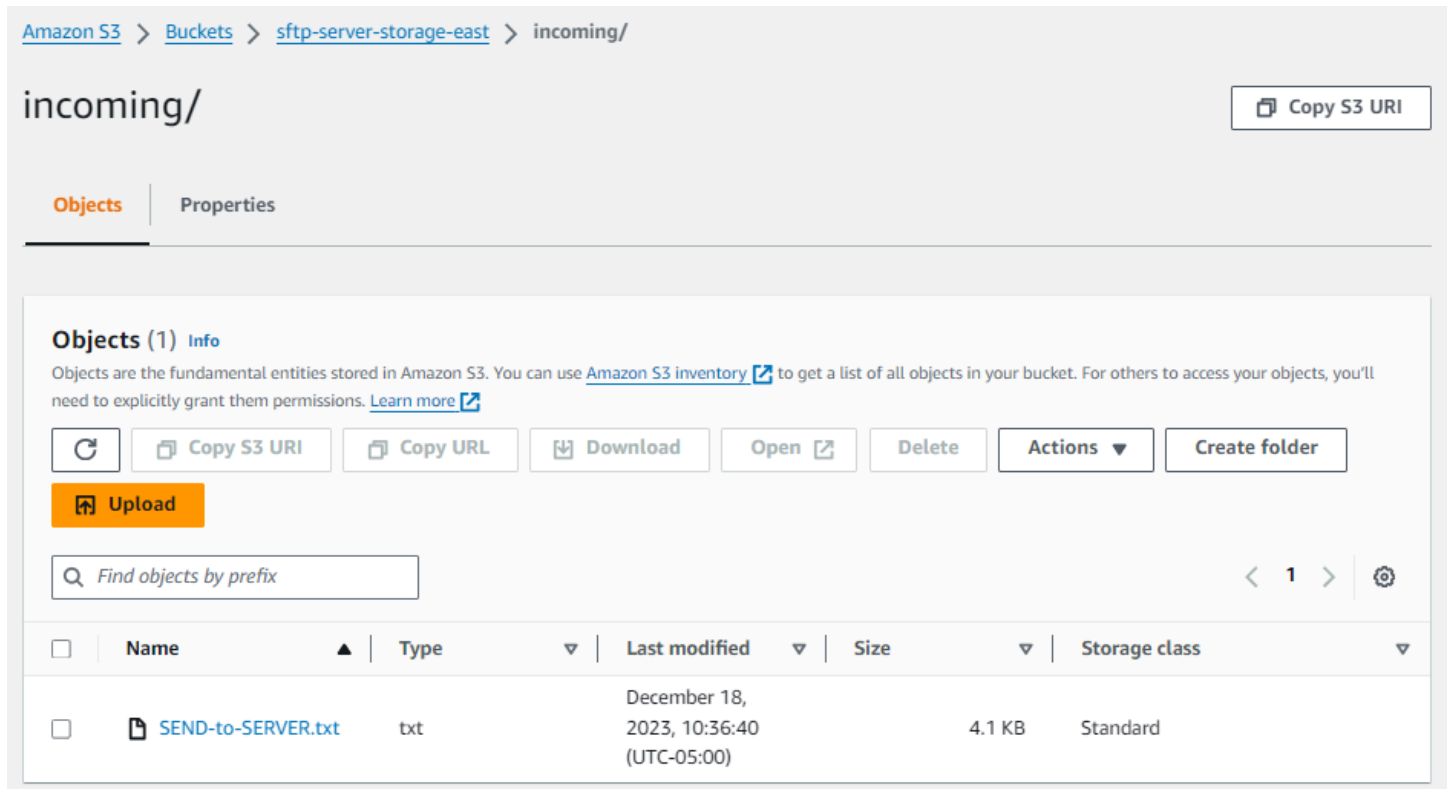
Note

다음 명령에서 *connector-id*를 커넥터 ID로 바꿉니다.

먼저 Amazon S3 버킷에서 원격 SFTP 서버로 파일을 보냅니다. 명령 프롬프트에서 다음 명령을 실행합니다.

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

이제 sftp-server-storage-east 버킷이 다음과 같아야 합니다.



파일이 예상대로 표시되지 않으면 CloudWatch 로그를 확인합니다.

CloudWatch 로그를 확인하려면

1. <https://console.aws.amazon.com/cloudwatch/> Amazon CloudWatch 콘솔을 엽니다.
2. 왼쪽 탐색 메뉴에서 로그 그룹을 선택합니다.
3. 검색 창에 커넥터 ID를 입력하여 로그를 찾습니다.
4. 검색에서 반환되는 로그 스트림을 선택합니다.
5. 최신 로그 항목을 확장합니다.

성공하면 로그 항목은 다음과 같습니다.

```
{
  "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": "/sftp-server-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": "account-id",
    "connector-arn": "arn:aws:transfer:us-east-1:account-id:connector/connector-id",
    "remote-directory-path": "/sftp-server-storage-east/incoming"
  }

```

파일 전송에 실패한 경우 로그 항목에 문제를 지정하는 오류 메시지가 포함됩니다. 오류의 일반적인 원인은 IAM 권한 문제와 잘못된 파일 경로입니다.

다음으로 SFTP 서버에서 Amazon S3 버킷으로 파일을 검색합니다. 명령 프롬프트에서 다음 명령을 실행합니다.

```

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

```

전송에 성공하면 Amazon S3 버킷에 다음과 같이 전송된 파일이 포함됩니다.

Amazon S3 > Buckets > sftp-server-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](#)

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

성공하면 로그 항목은 다음과 같습니다.

```
{
  "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": "account-id",
  "connector-arn": "arn:aws:transfer:us-east-1:account-id:connector/c-connector-id",
  "local-directory-path": "/sftp-server-storage-east/incoming"
}
```

VPC 송신 유형 커넥터 문제 해결

VPC 송신 유형 커넥터에 문제가 있는 경우 다음을 확인하세요.

- 커넥터 상태는 PENDING: VPC 커넥터의 DNS 확인에는 몇 분 정도 걸릴 수 있습니다. 연결을 시도하기 전에 상태가 ACTIVE가 될 때까지 기다립니다.
- 연결 제한 시간: 보안 그룹이 Resource Gateway 서브넷과 대상 SFTP 서버 간의 포트 22 트래픽을 허용하는지 확인합니다.
- 리소스 구성 오류: 리소스 구성이 올바른 IP 주소 또는 DNS 이름을 가리키고 Resource Gateway가 SFTP 서버(프라이빗 엔드포인트용)와 동일한 VPC에 있는지 확인합니다. 자세한 내용은 Amazon VPC Lattice 사용 설명서의 [Resource configurations](#) 섹션을 참조하시기 바랍니다.
- 퍼블릭 엔드포인트 문제: 퍼블릭 엔드포인트의 경우 리소스 구성에서 IP 주소가 아닌 DNS 이름을 사용하고 있는지 확인합니다. VPC에 아웃바운드 인터넷 액세스를 위한 NAT 게이트웨이가 있는지 확인합니다.
- AZ 가용성: Resource Gateway에는 2개 이상의 가용 영역에 서브넷이 필요합니다. 모든 AZs VPC Lattice를 지원하는 것은 아닙니다. 해당 리전에서 지원되는 AZs 확인하세요.

VPC 송신 유형에 대한 비용 고려 사항:

- VPC Lattice는 리소스 공급자로서 데이터 처리에 대해 0.006/GB의 요금을 부과합니다(VPC Lattice에서 직접 청구).
- AWS Transfer Family는 0.01 USD/GB 리소스 소비자 비용(첫 1PB)을 흡수합니다.
- VPC를 통한 퍼블릭 엔드포인트의 경우 추가 NAT 게이트웨이 및 데이터 전송 요금이 적용될 수 있습니다.
- 표준 0.40 USD/GB 데이터 처리 요금을 초과하는 추가 Transfer Family 요금 없음

원격 SFTP 서버로 사용할 Transfer Family 서버를 생성하는 절차

다음은이 자습서에서 원격 SFTP 서버 역할을 하는 Transfer Family 서버를 생성하는 단계를 간략하게 설명합니다. 다음 사항에 유의하세요.

- Transfer Family 서버를 사용하여 원격 SFTP 서버를 나타냅니다. 일반적인 SFTP 커넥터 사용자에게는 자체 원격 SFTP 서버가 있습니다. [Transfer Family SFTP 서버 및 사용자 생성을\(를\)](#) 참조하세요.
- Transfer Family 서버를 사용하기 때문에 서비스 관리형 SFTP 사용자도 사용합니다. 또한 간소화를 위해이 사용자가 Transfer Family 서버에 액세스하는 데 필요한 권한과 커넥터를 사용하는 데 필요한 권한을 결합했습니다. 다시 말하지만 대부분의 SFTP 커넥터 사용 사례에는 Transfer Family 서버와 연결되지 않은 별도의 SFTP 사용자가 있습니다. [Transfer Family SFTP 서버 및 사용자 생성을\(를\)](#) 참조하세요.

- 자습서에서는 원격 SFTP 서버에 Amazon S3 스토리지를 사용하므로 한 버킷에서 다른 버킷으로 파일을 전송할 수 **sftp-server-storage-east** 있도록 두 번째 버킷인을 생성해야 합니다.

Transfer Family SFTP 서버 및 사용자 생성

사용자가 있는 SFTP 서버가 이미 있으므로 대부분의 사용자는 Transfer Family SFTP 서버와 사용자를 생성할 필요가 없으며 서버를 사용하여 파일을 주고받을 수 있습니다. 그러나 이 자습서에서는 간소화를 위해 Transfer Family 서버를 사용하여 원격 SFTP 서버로 작동합니다.

에 설명된 절차에 따라 [SFTP 지원 서버 생성](#) 서버를 생성하고 사용자를 [3단계: 서비스 관리 사용자 추가](#) 추가합니다. 다음은 자습서에서 사용 중인 사용자 세부 정보입니다.

- 서비스 관리형 사용자를 생성합니다 `sftp-testuser`.
 - 홈 디렉터리를 로 설정 `/sftp-server-storage-east/sftp-testuser`
 - 사용자를 생성할 때 퍼블릭 키를 저장합니다. 나중에 Secrets Manager에서 보안 암호를 생성할 때 해당 프라이빗 키를 제공해야 합니다.
- 역할: `sftp-connector-role`. 자습서에서는 SFTP 사용자와 SFTP 커넥터 액세스 모두에 동일한 IAM 역할을 사용합니다. 조직의 커넥터를 생성할 때 별도의 사용자 및 액세스 역할이 있을 수 있습니다.
- 서버 호스트 키: 커넥터를 생성할 때 서버 호스트 키를 사용해야 합니다. 서버에 `ssh-keyscan` 대행을 실행하여 키를 검색할 수 있습니다. 예를 들어 서버 ID가 `s-1111aaaa2222bbbb3`이고 엔드포인트가 있는 경우 `us-east-1` 다음 명령은 서버 호스트 키를 검색합니다.

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

[2단계: SFTP 커넥터 생성 및 테스트](#) 절차에 붙여 넣어야 하므로 이 텍스트를 어딘가에 복사합니다.

사용자 및 액세스 역할 결합

자습서에서는 결합된 단일 역할을 사용합니다. 이 역할은 SFTP 사용자뿐만 아니라 커넥터에 대한 액세스에도 사용됩니다. 다음 예제에는 자습서에서 작업을 수행하려는 경우 이 역할에 대한 세부 정보가 포함되어 있습니다.

다음 예제에서는 Amazon S3의 두 버킷과 Secrets Manager에 `aws/transfer/sftp-connector1` 저장된 이라는 보안 암호에 액세스하는 데 필요한 권한을 부여합니다. 자습서의 경우 이 역할의 이름은 `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:::sftp-server-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:::sftp-server-storage-east/*"
      ]
    },
    {
      "Sid": "GetConnectorSecretValue",
      "Effect": "Allow",
      "Action": [
        "secretsmanager:GetSecretValue"
      ],
      "Resource": "arn:aws:secretsmanager:us-east-1:111122223333:secret:aws/transfer/sftp-connector1-6RandomCharacters"
    }
  ]
}

```

}

Transfer Family에 대한 역할 생성에 대한 전체 세부 정보에는 설명된 절차에 따라 역할을 [사용자 역할 생성](#) 생성합니다.

Amazon API Gateway 메서드를 사용자 지정 자격 증명 공급자로 설정

이 자습서에서는 Amazon API Gateway 메서드를 설정하고 사용자 지정 자격 증명 공급자로 사용하여 AWS Transfer Family 서버에 파일을 업로드하는 방법을 보여줍니다. 이 자습서에서는 [기본 스택 템플릿](#)과 기타 기본 기능을 예로만 사용합니다.

주제

- [사전 조건](#)
- [1단계: CloudFormation 스택 생성](#)
- [2단계: 서버의 API Gateway 메서드 구성을 확인합니다.](#)
- [3단계: Transfer Family 서버 세부 정보 보기](#)
- [4단계: 사용자가 서버에 연결할 수 있는지 테스트](#)
- [5단계: SFTP 연결 및 파일 전송 테스트](#)
- [6단계: 버킷에 대한 액세스 제한](#)
- [Amazon EFS를 사용하는 경우 Lambda 업데이트](#)

사전 조건

에서 Transfer Family 리소스를 생성하기 전에 스토리지와 사용자 역할을 CloudFormation 생성합니다.

스토리지를 지정하고 사용자 역할을 생성하려면

1. 사용 중인 스토리지에 따라 다음 설명서를 참조하세요.
 - Amazon S3 버킷을 생성하려면 Amazon Simple Storage Service 사용 설명서의 [S3 버킷을 생성하려면 어떻게 해야 하나요?](#) 단원을 참조하세요.
 - Amazon EFS 파일 시스템을 생성하려면 섹션을 참조하세요 [Amazon EFS 파일 시스템 구성](#).
2. 사용자 역할을 생성하려면 [IAM 역할 및 정책 생성](#) 단원을 참조하세요.

다음 섹션에서 CloudFormation 스택을 생성할 때 스토리지의 세부 정보와 사용자 역할을 입력합니다.

1단계: CloudFormation 스택 생성

제공된 템플릿에서 AWS CloudFormation 스택을 생성하려면

1. <https://console.aws.amazon.com/cloudformation> CloudFormation 콘솔을 엽니다.
2. 스택 생성을 선택한 다음 새 리소스 사용(표준)을 선택합니다.
3. 사전 조건 - 템플릿 준비 창에서 기존 템플릿 선택을 선택합니다.
4. [기본 스택 템플릿](#)인 이 링크를 복사하여 Amazon S3 URL 필드에 붙여넣습니다.
5. 다음을 클릭합니다.
6. 스택 이름을 비롯한 파라미터를 지정합니다. 다음 작업을 수행합니다.
 - UserName 및 UserPassword의 기본값을 바꿉니다.
 - UserHomeDirectory의 경우 이전에 생성한 스토리지(Amazon S3 버킷 또는 Amazon EFS 파일 시스템)의 세부 정보를 입력합니다.
 - 기본 UserRoleArn을 이전에 생성한 사용자 역할로 바꿉니다. AWS Identity and Access Management (IAM) 역할에는 적절한 권한이 있어야 합니다. IAM 역할 및 버킷 정책 예는 [6단계: 버킷에 대한 액세스 제한](#) 단원을 참조하세요.
 - 암호 대신 퍼블릭 키를 사용하여 인증하려면 UserPublicKey1 필드에 퍼블릭 키를 입력합니다. SFTP를 사용하여 서버에 처음 연결할 때 암호 대신 프라이빗 키를 입력합니다.
7. 다음을 선택한 후 스택 옵션 구성 페이지에서 다시 다음을 선택합니다.
8. 생성 중인 스택의 세부 정보를 검토한 다음 스택 생성을 선택합니다.

Note

페이지 하단의 기능에서 CloudFormation 이 IAM 리소스를 생성할 수도 있음을 인정해야 합니다.

2단계: 서버의 API Gateway 메서드 구성을 확인합니다.

Note

보안을 강화하기 위해 웹 애플리케이션 방화벽을 구성할 수 있습니다. AWS WAF 은(는) Amazon API Gateway에 전달되는 HTTP 및 HTTPS 요청을 모니터링할 수 있게 해주는 웹 애플리케이션 방화벽입니다. 자세한 내용은 [웹 애플리케이션 방화벽 추가](#)를 참조하세요.

⚠ API Gateway 캐싱을 활성화하지 않음

Transfer Family의 사용자 지정 자격 증명 공급자로 API Gateway 메서드를 사용할 때는 캐싱을 활성화하지 마십시오. 캐싱은 다음과 같은 이유로 인증 요청에 부적절하고 유효하지 않습니다.

- 각 인증 요청은 고유하며 캐시된 응답이 아닌 라이브 응답이 필요합니다.
- Transfer Family는 중복되거나 반복적인 요청을 API Gateway로 보내지 않으므로 캐싱은 이 점을 제공하지 않습니다.
- 캐싱을 활성화하면 API Gateway가 일치하지 않는 데이터로 응답하여 인증 요청에 유효하지 않은 응답이 발생합니다.

서버의 API Gateway 메서드 구성을 확인하고 배포하려면

1. <https://console.aws.amazon.com/apigateway/>에서 Amazon API Gateway 콘솔을 엽니다.
2. 템플릿이 생성한 사용자 지정 자격 증명 공급자 기본 템플릿 전송 API를 선택합니다.
CloudFormation
3. 리소스 창에서 GET을 선택한 후 메서드 요청을 선택합니다.
4. 작업 및 API 배포를 선택합니다. 배포 단계에서 prod를 선택한 다음 배포를 선택합니다.

API Gateway 메서드가 성공적으로 배포되면 단계 편집기 섹션에서 성능을 확인합니다.

Note

페이지 상단에 표시되는 URL 호출 주소를 복사합니다. 이 정보는 다음 단계에 필요합니다.

3단계: Transfer Family 서버 세부 정보 보기

템플릿을 사용하여 CloudFormation 스택을 생성하면 Transfer Family 서버가 자동으로 생성됩니다.

Transfer Family 서버 세부 정보 보기

1. <https://console.aws.amazon.com/cloudformation> CloudFormation 콘솔을 엽니다.
2. 생성한 스택을 선택합니다.
3. 리소스 탭을 선택합니다.

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	

서버 ARN은 전송 서버 행의 물리적 ID 열에 표시됩니다. 서버 ID는 ARN에 포함되어 있습니다(예: s-11112222333344445).

4. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 열고 서버 페이지에서 새 서버를 선택합니다.

서버 ID는 TransferServer 리소스에 대해 표시된 ID와 일치합니다 CloudFormation.

4단계: 사용자가 서버에 연결할 수 있는지 테스트

Transfer Family 콘솔을 사용하여 사용자가 서버에 연결할 수 있는지 테스트하려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 서버 페이지에서 새 서버를 선택하고 작업을 선택한 다음 테스트를 선택합니다.
3. 사용자 이름 필드와 암호 필드에 로그인 자격 증명의 텍스트를 입력합니다. CloudFormation 스택을 배포할 때 설정하는 값은 다음과 같습니다.
4. 서버 프로토콜에 대해 SFTP를 선택하고 소스 IP에 **127.0.0.1**을 입력합니다.
5. 테스트를 선택합니다.

사용자 인증에 성공하면 테스트에서 StatusCode: 200 HTML 응답과 사용자 역할 및 권한의 세부 정보가 포함된 JSON 개체가 반환됩니다. 예제:

```
{
  "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\"
}
```

테스트가 실패하면 API Gateway AWS 관리형 정책 중 하나를 API에 사용 중인 역할에 추가합니다.

5단계: SFTP 연결 및 파일 전송 테스트

SFTP 연결을 테스트하려면

1. Linux 또는 macOS 장치에서 명령 터미널을 엽니다.
2. 인증에 암호를 사용하는지 또는 키 쌍을 사용하는지 여부에 따라 다음 명령 중 하나를 입력합니다.

- 암호를 사용하는 경우 이 명령 입력:

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

메시지가 표시되면 암호를 입력합니다.

- 키 쌍을 사용하는 경우 이 명령 입력:

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

Note

이러한 sftp 명령에 Transfer Family 서버가 위치해 있는 AWS 리전 에 해당하는 코드를 입력합니다. 예를 들어 서버가 미국 동부(오하이오)에 있는 경우 **us-east-2**를 입력합니다.

3. sftp> 프롬프트에서 디렉터리와 파일(pwd 및 ls)을 업로드(put)하고 다운로드(get)하고 볼 수 있는지 확인합니다.

6단계: 버킷에 대한 액세스 제한

특정 Amazon S3 버킷에 액세스할 수 있는 사용자를 제한할 수 있습니다. 다음 예는 CloudFormation 스택과 사용자에 대해 선택한 정책에서 사용할 설정을 보여줍니다.

이 예제에서는 CloudFormation 스택에 대해 다음 파라미터를 설정합니다.

- CreateServer: true
- UserHomeDirectory: /amzn-s3-demo-bucket1
- UserName: myuser
- UserPassword: MySuperSecretPassword

Important

다음은 암호의 예시입니다. API Gateway 메서드를 구성할 때는 강력한 암호를 입력해야 합니다.

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

UserPublicKey1은 퍼블릭/프라이빗 키 쌍의 일부로 생성한 퍼블릭 키입니다.

*role-id*은(는) 생성한 사용자 역할에 따라 다릅니다. myuser-api-gateway-role에 연결된 정책은 다음과 같습니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": "s3:ListBucket",
      "Resource": "arn:aws:s3:::amzn-s3-demo-bucket1"
    },
    {
      "Sid": "VisualEditor1",
      "Effect": "Allow",
      "Action": [
        "s3:PutObject",
        "s3:GetObjectAcl",
        "s3:GetObject",
        "s3:DeleteObjectVersion",
        "s3:DeleteObject",
        "s3:PutObjectAcl",
        "s3:GetObjectVersion"
      ],
      "Resource": "arn:aws:s3:::amzn-s3-demo-bucket1/*"
    }
  ]
}
```

SFTP를 사용하여 서버에 연결하려면 프롬프트에 다음 명령 중 하나를 입력합니다.

- 암호를 사용하여 인증하는 경우 실행할 명령:

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

메시지가 표시되면 암호를 입력합니다.

- 키 쌍을 사용하여 인증하는 경우 실행할 명령:

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

Note

이러한 sftp 명령의 경우 Transfer Family 서버가 AWS 리전 위치의 ID를 사용합니다. 예를 들어 서버가 미국 동부(오하이오)에 있는 경우 us-east-2를 사용하세요.

sftp 프롬프트가 표시되면 pwd 명령을 실행하여 홈 디렉터리를 볼 수 있습니다. 예제:

```
sftp> pwd
Remote working directory: /amzn-s3-demo-bucket1
```

사용자는 홈 디렉터리 상위 디렉터리는 일체 볼 수 없습니다. 예제:

```
sftp> pwd
Remote working directory: /amzn-s3-demo-bucket1
sftp> cd ..
sftp> ls
Couldn't read directory: Permission denied
```

Amazon EFS를 사용하는 경우 Lambda 업데이트

Transfer Family 서버의 스토리지 옵션으로 Amazon EFS를 선택한 경우 스택의 람다 함수를 편집해야 합니다.

Lambda 함수에 Posix 프로파일을 추가하려면

1. <https://console.aws.amazon.com/lambda/>에서 Lambda 콘솔을 엽니다.
2. 이전에 생성한 Lambda 함수를 선택합니다. Lambda 함수의 형식은 **stack-name-GetUserConfigLambda-lambda-identifier**입니다. 여기서 **stack-name**은 CloudFormation 스택 이름이고 **lambda-identifier**는 함수의 식별자입니다.
3. 코드 탭에서 index.js를 선택하여 함수에 대한 코드를 표시합니다.

4. response에서 Policy과(와) HomeDirectory 사이에 다음 줄을 추가합니다.

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

여기서 *uid-value*와 *gid-value*는 각각 사용자 ID와 그룹 ID를 나타내는 0 이상의 정수입니다.

예를 들어, POSIX 프로필을 추가한 후 응답 필드는 다음과 같을 수 있습니다.

```
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 '/'
};
```

AS2 구성 설정

이 자습서에서는를 사용하여 적용성 문 2(AS2) 구성을 설정하는 방법을 안내합니다 AWS Transfer Family. 여기에 설명된 단계를 완료하면 샘플 거래 파트너(sample trading partner)의 AS2 메시지를 수락할 준비가 된 AS2 지원 서버를 갖게 됩니다. 샘플 거래 파트너에게 AS2 메시지를 보내는 데 사용할 수 있는 커넥터도 있습니다.

Note

예제 설정의 일부 부분에서는 AWS Command Line Interface ()를 사용합니다AWS CLI. 아직을 설치하지 않은 경우 AWS Command Line Interface 사용 설명서의 [의 최신 버전 설치 또는 업데이트를 AWS CLI](#) AWS CLI참조하세요.

1. 사용자 자신과 사용자의 거래 파트너를 위한 인증서를 만듭니다. 사용할 수 있는 기존 인증서가 있다면 이 단계를 건너뛰어도 됩니다.

이 과정은 [1단계: AS2용 인증서 생성](#)에 설명되어 있습니다.

2. 1단계에서 생성한 인증서를 가져옵니다.

이 과정은 [2단계: 인증서를 Transfer Family 인증서 리소스로 가져오기](#)에 설명되어 있습니다.

3. 거래 파트너를 설정하려면 로컬 프로필과 파트너 프로필을 생성하세요.

이 과정은 [3단계: 사용자와 거래 파트너를 위한 프로필 생성](#)에 설명되어 있습니다.

4. AS2 프로토콜을 사용하는 AWS Transfer Family 서버를 생성합니다. 선택적으로 서버에 탄력적 IP 주소를 추가하여 인터넷에 연결되도록 할 수 있습니다.

이 과정은 [4단계: AS2 프로토콜을 사용하는 Transfer Family 서버 생성](#)에 설명되어 있습니다.

Note

인바운드 전송에만 사용할 Transfer Family 서버를 생성해야 합니다. 아웃바운드 전송만 수행하는 경우에는 Transfer Family 서버가 필요하지 않습니다.

5. 거래 파트너와 계약을 체결합니다.

이 과정은 [5단계: 사용자와 사용자의 파트너 간의 계약서 작성](#)에 설명되어 있습니다.

Note

인바운드 전송에 대한 계약만 작성해야 합니다. 아웃바운드 전송만 수행하는 경우에는 계약이 필요하지 않습니다.

6. 사용자와 사용자의 거래 파트너와 계약을 체결합니다.

이 과정은 [6단계: 사용자와 사용자의 거래 파트너 간의 커넥터를 생성합니다](#)에 설명되어 있습니다.

Note

아웃바운드 전송 전용 커넥터를 생성해야 합니다. 인바운드 전송만 수행하는 경우 커넥터가 필요하지 않습니다.

7. AS2 파일 교환을 테스트해 보세요.

이 과정은 [7단계: Transfer Family를 사용하여 AS2를 통한 파일 교환 테스트](#)에 설명되어 있습니다.

이 단계를 완료한 후 다음을 수행할 수 있습니다.

- Transfer Familystart-file-transfer AWS Command Line Interface (AWS CLI) 명령을 사용하여 원격 AS2-enabled 파트너 서버로 파일을 전송합니다.

- Virtual Private Cloud(VPC) 엔드포인트를 통해 포트 5080에서 원격 AS2 지원 파트너 서버로부터 파일을 수신합니다.

1단계: AS2용 인증서 생성

AS2 거래소의 양 당사자는 모두 X.509 인증서가 필요합니다. 원하는 방식으로 이러한 인증서를 생성할 수 있습니다. 이 주제에서는 명령줄에서 [OpenSSL](#)을 사용하여 루트 인증서를 만든 다음 하위 인증서에 서명하는 방법을 설명합니다. 양 당사자는 자체 인증서를 생성해야 합니다.

Note

AS2 인증서의 키 길이는 최소 2,048비트, 최대 4,096비트여야 합니다.

파트너와 파일을 전송하려면 다음 사항에 유의하세요.

- 프로필에 인증서를 첨부할 수 있습니다. 인증서에는 퍼블릭 키 또는 프라이빗 키가 포함됩니다.
- 사용자의 거래 파트너가 사용자에게 퍼블릭 키를 보내면, 사용자는 그에게 사용자의 퍼블릭 키를 보냅니다.
- 거래 파트너는 퍼블릭 키로 메시지를 암호화하고 프라이빗 키로 서명합니다. 정반대로, 사용자는 거래 파트너의 퍼블릭 키로 메시지를 암호화하고 그 메시지에 사용자 프라이빗 키로 서명합니다.

Note

GUI로 키를 관리하려는 경우, 사용할 수 있는 옵션 중 하나가 [Portecle](#)입니다.

예 인증서를 생성하려면

Important

파트너에게 사용자 프라이빗 키를 보내지 마세요. 이 예시에서, 사용자는 한 당사자를 위해 자체 서명된 퍼블릭 키와 프라이빗 키 세트를 생성합니다. 테스트 목적으로 두 거래 파트너 역할을 모두 수행하려는 경우 이 지침을 반복하여 거래 파트너당 하나씩 총 두 세트의 키를 생성할 수 있습니다. 이 경우 두 개의 루트 인증 기관(CA)을 생성할 필요가 없습니다.

1. 다음 명령을 실행하여 2,048비트 길이의 모듈러스가 포함된 RSA 프라이빗 키를 생성합니다.

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

2. 다음 명령을 실행하여 사용자의 root-ca-key.pem파일로 자체 서명된 인증서를 생성합니다.

```
/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 아규먼트는 다음 두 값으로 구성됩니다.

	이름	설명
C	국가 코드	사용자 조직이 위치한 국가를 나타내는 2자리 코드입니다.
ST	주, 지역 또는 지방	해당 조직이 위치한 주 또는 지방의 이름. (이 경우 지역은 사용자의 AWS 리전을 의미하지 않습니다.)
L	시 이름	해당 조직이 위치한 시의 이름.
O	조직 이름	LLC, Corp 등과 같은 접미사를 포함한 사용자 조직의 전체 법적 이름.
OU	조직 단위 이름	조직 내에서 이 인증서를 다루는 부서.
CN	일반적인 이름 또는 완전한 자격을 갖춘 도메인 이름 (FQDN)	이 예에서는 루트 인증서를 생성하므로 값은 R00TCA입니다. 이 예에서는 인증서의

	이름	설명
		용도를 설명하는데 CN를 사용합니다.

3. 로컬 프로필을 위한 서명 키와 암호화 키를 생성하세요.

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

Note

OpenAS2와 같은 일부 AS2 지원 서버에서는 서명과 암호화 모두에 동일한 인증서를 사용해야 합니다. 이 경우 두 가지 용도로 동일한 프라이빗 키와 인증서를 가져올 수 있습니다. 이렇게 하려면 이전 명령 두 개 대신 다음 명령을 실행합니다.

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

4. 다음 명령을 실행하여 루크 키를 위한 인증서 서명 요청(CSR)을 생성하여 서명합니다.

```
/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. 다음으로, signing-cert.conf 파일과 encryption-cert.conf 파일을 만들어야 합니다.

- 텍스트 편집기를 사용하여 다음 내용을 포함하는 signing-cert.conf 파일을 생성합니다.

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

- 텍스트 편집기를 사용하여 다음 내용을 포함하는 encryption-cert.conf 파일을 생성합니다.

```
authorityKeyIdentifier=keyid,issuer
```

```
keyUsage = dataEncipherment
```

6. 마지막으로 다음 명령을 실행하여 서명된 인증서를 생성합니다.

```
/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
```

2단계: 인증서를 Transfer Family 인증서 리소스로 가져오기

이 절차에서는 AWS CLI를 사용하여 인증서를 가져오는 방법을 설명합니다. 이 방법 대신 Transfer Family 콘솔을 사용하려면 [the section called “AS2 인증서 가져오기”](#) 섹션을 참조하세요.

1단계에서 만든 서명 및 암호화 인증서를 가져오려면 다음 `import-certificate` 명령을 실행합니다. 암호화와 서명에 동일한 인증서를 사용하는 경우 동일한 인증서를 두 번 가져오세요 (한 번은 SIGNING 사용량과 함께, 다른 한 번은 ENCRYPTION 사용량과 함께).

Note

인증서와 해당 체인이 모두 포함된 파일이 있는 경우 `certificate` 파라미터만 사용하여 해당 파일을 `import-certificate` 명령에 제공할 수 있습니다. 예제:

```
aws transfer import-certificate --usage ENCRYPTION --certificate
file:///combined-cert-and-chain-file.pem
```

`certificate` 파라미터를 사용하여 인증서와 해당 체인을 모두 업로드하는 경우 `certificate-chain` 파라미터를 사용하지 마십시오.

인증서와 체인을 결합하면 키는 64자마다 줄 바꿈 "\n"이 포함된 기존 PEM 표준을 사용하여 형식이 지정됩니다. 저장된 인증서는 사용자가 업로드한 인증서와 기능적으로 동일하며,를 통한 `DescribeCertificate` 응답 AWS CLI 에는 이러한 줄 바꿈 문자가 포함됩니다.

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


이 명령은 사용자 서명 CertificateId를 반환합니다. 다음 섹션에서는 이 인증서 ID를 *my-signing-cert-id*라고 합니다.

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

이 명령은 암호화 CertificateId를 반환합니다. 다음 섹션에서는 이 인증서 ID를 *my-encrypt-cert-id*라고 합니다.

그런 다음, 다음 명령을 실행하여 파트너의 암호화 및 서명 인증서를 가져옵니다.

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

이 명령은 파트너의 암호화 CertificateId를 반환합니다. 다음 섹션에서는 이 인증서 ID를 *partner-encrypt-cert-id*라고 합니다.

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

이 명령은 파트너의 서명 CertificateId를 반환합니다. 다음 섹션에서는 이 인증서 ID를 *partner-signing-cert-id*라고 합니다.

3단계: 사용자와 거래 파트너를 위한 프로파일 생성

이 절차에서는를 사용하여 AS2 프로파일을 생성하는 방법을 설명합니다 AWS CLI. 이 방법 대신 Transfer Family 콘솔을 사용하려면 [the section called “AS2 프로파일 생성”](#) 섹션을 참조하세요.

다음 명령을 실행하여 로컬 AS2 프로파일을 생성합니다. 이 명령은 퍼블릭 키와 프라이빗 키가 포함된 인증서를 참조합니다.

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

이 명령은 프로파일 ID를 반환합니다. 다음 섹션에서는 이 ID를 *my-profile-id*라고 합니다.

이제 다음 명령을 실행하여 파트너 프로필을 생성합니다. 이 명령어는 파트너의 퍼블릭 키 인증서만 사용합니다. 이 명령을 사용하려면 *user input placeholders*를 사용자 자신(예: 파트너의 AS2 이름 및 인증서 ID)의 정보로 바꾸세요.

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

이 명령은 파트너의 프로필 ID를 반환합니다. 다음 섹션에서는 이 ID를 *partner-profile-id*라고 합니다.

Note

이전 명령에서 *MYCORP*를 조직 이름으로 바꾸고 *PARTNER-COMPANY*를 사용자의 거래 파트너의 조직 이름으로 바꾸세요.

4단계: AS2 프로토콜을 사용하는 Transfer Family 서버 생성

이 절차에서는 Transfer Family AWS CLI를 사용하여 AS2 지원 서버를 생성하는 방법을 설명합니다.

Note

많은 예 단계에서 파일에서 파라미터를 로드하는 명령을 사용합니다. 파일을 사용하여 파라미터를 로드하는 방법에 대한 자세한 내용은 [파일에서 파라미터를 로드하는 방법](#)을 참조하세요.

이 방법 대신 콘솔을 사용하려면 [Transfer Family 콘솔을 사용하여 AS2 서버 생성](#) 섹션을 참조하세요.

SFTP 또는 FTPS AWS Transfer Family 서버를 생성하는 방법과 마찬가지로 `create-server` AWS CLI 명령의 `--protocols AS2` 파라미터를 사용하여 AS2-enabled 서버를 생성합니다. 현재 Transfer Family는 VPC 엔드포인트 타입과 AS2 프로토콜을 사용하는 Amazon S3 스토리지만 지원합니다.

`create-server` 명령을 사용하여 Transfer Family용 AS2 지원 서버를 생성하면 VPC 엔드포인트가 자동으로 생성됩니다. 이 엔드포인트는 AS2 메시지를 받아들일 수 있도록 TCP 포트 5080을 노출합니다.

VPC 엔드포인트를 인터넷에 공개하려는 경우 탄력적 IP 주소를 VPC 엔드포인트와 연결할 수 있습니다.

이러한 지침을 사용하려면 다음이 필요합니다.

- 사용자 VPC의 ID (예: vpc-abcdef01).
- 사용자 VPC 서브넷의 ID (예: 서브넷-abcdef01, 서브넷-서브넷-abcdef01, 서브넷-021345ab).
- 거래 파트너로부터 TCP 포트 5080을 통해 들어오는 트래픽을 허용하는 보안 그룹의 ID 하나 이상 (예: sg-1234567890abcdef0 및 sg-abcdef01234567890)
- (옵션) VPC 엔드포인트와 연결하려는 탄력적 IP 주소.
- 거래 파트너가 VPN을 통해 VPC에 연결되지 않은 경우 인터넷 게이트웨이가 필요합니다. 자세한 내용은 Amazon VPC 사용 설명서의 [인터넷 게이트웨이를 사용한 인터넷 연결](#)을 참조하세요.

AS2 지원 서버를 만들려면

1. 다음 명령을 실행합니다. 각 *user input placeholder*를 사용자의 정보로 바꿉니다.

```
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. (옵션) VPC 엔드포인트를 퍼블릭으로 만들 수 있습니다. update-server 작업을 통해서만 Transfer Family 서버에 탄력적 IP 주소를 연결할 수 있습니다. 다음 명령은 서버를 중지하고 탄력적 IP 주소로 업데이트한 다음 다시 시작합니다.

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

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

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

이 start-server 명령은 사용자 서버의 퍼블릭 IP 주소가 포함된 DNS 레코드를 자동으로 생성합니다. 거래 파트너에게 서버 액세스 권한을 부여하려면 거래 파트너에게 다음 정보를 제공해야 합니다. 이 경우에는 *your-region*는 귀하의 AWS 리전을 참조합니다.

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

거래 파트너에게 제공하는 전체 URL은 다음과 같습니다.

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

3. AS2 지원 서버의 액세스 가능 여부를 테스트하려면 다음 명령을 사용하세요. VPC 엔드포인트의 프라이빗 DNS 주소 또는 퍼블릭 엔드포인트(탄력적 IP 주소를 엔드포인트와 연결한 경우)를 통해 서버에 액세스할 수 있는지 확인하세요.

서버가 올바르게 구성되어 있으면 연결이 성공합니다. 하지만 유효한 AS2 메시지를 보내고 있지 않기 때문에 HTTP 상태 코드 400(잘못된 요청) 응답을 받게 됩니다.

- 퍼블릭 엔드포인트의 경우 (이전 단계에서 탄력적 IP 주소를 연결한 경우) 서버 ID와 지역을 대체하여 다음 명령을 실행합니다.

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

- VPC 내에서 연결하는 경우 다음 명령을 실행하여 VPC 엔드포인트의 프라이빗 DNS 이름을 조회합니다.

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

이 `describe-server` 명령은 `VpcEndpointId` 파라미터에 VPC 엔드포인트 ID를 반환합니다. 이 값을 사용하여 다음 명령을 실행합니다.

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

이 `describe-vpc-endpoints` 명령은 `DNSEntries` 배열을 반환하며, 여기에는 여러 `DnsName` 파라미터가 포함됩니다. 다음 명령에서 지역 DNS 이름(가용 영역을 포함하지 않는 이름)을 사용합니다.

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

예를 들어, 다음 명령은 이전 명령의 자리 표시자에 대한 샘플 값을 보여줍니다.

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

4. (옵션) 로깅 역할을 구성합니다. Transfer Family는 구조화된 JSON 형식으로 송수신된 메시지의 상태를 Amazon CloudWatch Logs에 기록합니다. Transfer Family에게 계정의 CloudWatch 로그에 대한 액세스 권한을 제공하려면 서버에서 로깅 역할을 구성해야 합니다.

를 신뢰하는 AWS Identity and Access Management (IAM) 역할을 생성하고 AWSTransferLoggingAccess 관리형 정책을 transfer.amazonaws.com에 연결합니다. 자세한 내용은 [IAM 역할 및 정책 생성](#)을 참조하세요. 방금 생성한 IAM 역할의 Amazon 리소스 이름 (ARN)을 메모하고 다음 update-server 명령을 실행하여 서버에 연결합니다.

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

Note

로깅 역할은 옵션이지만 메시지 상태를 확인하고 구성 문제를 해결할 수 있도록 로깅 역할을 설정하는 것이 좋습니다.

5단계: 사용자와 사용자의 파트너 간의 계약서 작성

이 절차에서는 AWS CLI를 사용하여 AS2 계약서를 생성하는 방법에 대해 설명합니다. 이 방법 대신 Transfer Family 콘솔을 사용하려면 [the section called “AS2 계약 생성”](#) 섹션을 참조하세요.

계약에는 두 프로필 (로컬 및 파트너), 인증서, 그리고 두 당사자 간의 인바운드 AS2 전송을 허용하는 서버 구성이 함께 제공됩니다. 다음 명령을 실행하여 항목을 나열할 수 있습니다.

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

이 단계에는 버킷에 대한 읽기/쓰기 액세스 권한이 있는 Amazon S3 버킷 및 IAM 역할이 필요합니다. 이 역할을 생성하는 지침은 Transfer Family SFTP, FTP 및 FTPS 프로토콜의 지침과 동일하며 [IAM 역할 및 정책 생성](#)에서 확인할 수 있습니다.

계약을 생성하려면 다음 항목이 필요합니다.

- AS2 파일을 저장할 Amazon S3 버킷 이름(및 지정된 경우 객체 접두사)입니다. 파일 유형에 따라 별도의 디렉터리를 지정하는 것이 좋습니다.
- 버킷에 액세스할 수 있는 IAM 역할의 ARN입니다.
- 사용자의 Transfer Family 서버 ID
- 사용자의 프로필 ID 및 사용자 파트너의 프로필 ID

와 같은 파일에 다음 코드를 저장합니다 `agreementDetails.json`. 각 *user input placeholder*를 사용자의 정보로 바꿉니다.

```
{
  "Description": "ExampleAgreementName",
  "ServerId": "your-server-id",
  "LocalProfileId": "your-profile-id",
  "PartnerProfileId": "your-partner-profile-id",
  "AccessRole": "arn:aws:iam::111111111111:role/TransferAS2AccessRole",
  "Status": "ACTIVE",
  "PreserveFilename": "ENABLED",
  "EnforceMessageSigning": "ENABLED",
  "CustomDirectories": {
    "FailedFilesDirectory": "/amzn-s3-demo-destination-bucket/AS2-failed",
    "MdnFilesDirectory": "/amzn-s3-demo-destination-bucket/AS2-mdn",
    "PayloadFilesDirectory": "/amzn-s3-demo-destination-bucket/AS2-payload",
    "StatusFilesDirectory": "/amzn-s3-demo-destination-bucket/AS2-status",
    "TemporaryFilesDirectory": "/amzn-s3-demo-destination-bucket/AS2-temp"
  }
}
```

Note

별도의 디렉터리 대신 단일 기본 디렉터를 사용하려면 이전 코드에서 `CustomDirectories` 줄과 개별 디렉터리 줄을 제거하고 대신 다음 파라미터를 사용합니다.

`"BaseDirectory": "/amzn-s3-demo-destination-bucket/AS2-inbox"`

기본 디렉터리와 별도의 디렉터리 파라미터를 모두 사용하지 마십시오. 그렇지 않으면 명령이 실패합니다.

그리고 다음 명령을 실행합니다.

```
aws transfer create-agreement --cli-input-json file://agreementDetails.json
```

성공한 경우 계약의 ID가 반환됩니다. 그리고 다음 명령을 사용하여 계약의 세부 정보를 볼 수 있습니다.

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

6단계: 사용자와 사용자의 거래 파트너 간의 커넥터를 생성합니다.

이 절차에서는 AWS CLI를 사용하여 AS2 커넥터를 생성하는 방법에 대해 설명합니다. 이 방법 대신 Transfer Family 콘솔을 사용하려면 [the section called “AS2 커넥터 구성”](#) 섹션을 참조하세요.

StartFileTransfer API 작업을 사용하면 커넥터를 사용하여 Amazon S3에 저장된 파일을 거래 파트너의 AS2 엔드포인트로 전송할 수 있습니다. 다음 명령을 실행하여 이전에 생성한 프로필을 찾을 수 있습니다.

```
aws transfer list-profiles
```

커넥터를 생성할 때 파트너의 AS2 서버 URL을 제공해야 합니다. 다음 텍스트를 testAS2Config.json라는 파일에 복사합니다.

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

Note

의 경우 DES_EDE3_CBC 알고리즘이 필요한 레거시 클라이언트를 지원해야 하는 경우가 아니면 알고리즘을 지정하지 EncryptionAlgorithm하십시오. 취약한 암호화 알고리즘이기 때문입니다.

이제 다음 명령을 실행하여 커넥터를 생성합니다.

```
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
```

7단계: Transfer Family를 사용하여 AS2를 통한 파일 교환 테스트

거래 파트너로부터 파일 받기

퍼블릭 탄력적 IP 주소를 VPC 엔드포인트와 연결한 경우 Transfer Family는 퍼블릭 IP 주소를 포함하는 DNS 이름을 자동으로 생성합니다. 하위 도메인은 AWS Transfer Family 서버 ID(형식)입니다 `s-1234567890abcdef0`. 거래 파트너에게 다음과 같은 형식으로 서버 URL을 입력합니다.

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

퍼블릭 탄력적 IP 주소를 VPC 엔드포인트와 연결하지 않은 경우, 포트 5080에서 거래 파트너가 보내는 HTTP POST를 통해 AS2 메시지를 받아들일 수 있는 VPC 엔드포인트의 호스트 이름을 찾아보세요. VPC 엔드포인트 세부 정보를 검색하려면 다음 명령을 사용합니다.

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

예를 들어 위의 명령이 `vpce-1234abcd5678efghi`의 VPC 엔드포인트 ID를 반환한다고 가정해 보겠습니다. 그런 다음, 다음 명령을 사용하여 DNS 이름을 검색합니다.

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

이 명령은 다음 명령을 실행하는 데 필요한 VPC 엔드포인트의 모든 세부 정보를 반환합니다.

DNS 이름은 `DnsEntries` 배열에 나열됩니다. 거래 파트너가 VPC 내에 있어야 VPC 엔드포인트에 액세스할 수 있습니다(예: AWS PrivateLink 또는 VPN을 통해). 거래 파트너에게 다음과 같은 형식으로 서버 VPC 엔드포인트 URL을 제공합니다.

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

예를 들어, 다음 URL은 이전 명령의 자리 표시자에 대한 샘플 값을 보여줍니다.


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

이 예에서 성공적인 전송은 지정한 base-directory 파라미터에 [5단계: 사용자와 사용자의 파트너 간의 계약서 작성](#)의 지정된 위치에 저장됩니다. 이름이 myfile1.txt 및 myfile2.txt 인 파일이 성공적으로 수신되면 파일은 */path-defined-in-the-agreement/processed/original_filename.messageId.original_extension*로 저장됩니다. 여기서 파일은 /amzn-s3-demo-destination-bucket/AS2-inbox/processed/myfile1.*messageId*.txt 및 /amzn-s3-demo-destination-bucket/AS2-inbox/processed/myfile2.*messageId*.txt로 저장됩니다.

Transfer Family 서버를 생성할 때 로깅 역할을 구성한 경우 CloudWatch 로그에서 AS2 메시지 상태를 확인할 수도 있습니다.

거래 파트너에게 파일을 보내세요.

Transfer Family를 사용하여 다음 start-file-transfer AWS Command Line Interface (AWS CLI) 명령과 같이 커넥터 ID와 파일 경로를 참조하여 AS2 메시지를 보낼 수 있습니다.

```
aws transfer start-file-transfer --connector-id c-1234567890abcdef0 \
--send-file-paths "/amzn-s3-demo-source-bucket/myfile1.txt" "/amzn-s3-demo-source-bucket/myfile2.txt"
```

커넥터의 세부 정보를 가져오려면 다음 명령을 실행합니다.

```
aws transfer list-connectors
```

이 list-connectors 명령은 커넥터의 커넥터 ID, URL 및 Amazon 리소스 이름(ARN)을 반환합니다.

특정 커넥터의 속성을 반환하려면 사용하려는 ID로 다음 명령을 실행합니다.

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

이 describe-connector 명령은 커넥터의 URL, 역할, 프로필, mDNS (메시지 처리 알림), 태그 및 모니터링 메트릭을 포함하여 커넥터의 모든 속성을 반환합니다.

JSON 및 MDN 파일을 보면 파트너가 파일을 성공적으로 수신했는지 확인할 수 있습니다. 이러한 파일 이름은 [파일 이름 및 위치](#)에 설명된 규칙에 따라 지정됩니다. 커넥터를 생성할 때 로깅 역할을 구성한 경우, CloudWatch 로그에서 AS2 메시지 상태를 확인할 수도 있습니다.

자습서: 기본 Transfer Family 웹 앱 설정

이 자습서에서는 Transfer Family 웹 앱을 설정하는 방법을 안내합니다. Transfer Family 웹 앱을 사용하면 웹 브라우저를 통해 Amazon S3와 데이터를 주고 받을 수 있는 간단한 인터페이스를 사용할 수 있습니다. 이 기능에 대한 자세한 설명서는 섹션을 참조하세요 [Transfer Family 웹 앱](#).

웹 앱 자습서: 사전 조건

- 의 계정 인스턴스 또는 조직 인스턴스를 생성합니다 AWS IAM Identity Center. 자세한 내용은 [Transfer Family 웹 앱에 대한 자격 증명 공급자 구성](#)을 참조하세요.

IAM Identity Center를 자격 증명 공급자로 사용하지 않는 경우는 대체(이 경우 Okta) 자격 증명 공급자를 사용하는 방법을 [Okta를 웹 앱의 자격 증명 공급자로 통합](#) 보여줍니다.

- Transfer Family 웹 앱과 상호 작용하는 데 사용할 Amazon S3 버킷이 필요합니다. 자세한 내용은 [Amazon S3 버킷 구성](#) 섹션을 참조하세요.

Note

이 자습서에서는 ID 제공업체에 IAM Identity Center 디렉터리를 사용하고 있다고 가정합니다. 그렇지 않은 경우 이 자습서를 진행하기 [Transfer Family 웹 앱에 대한 자격 증명 공급자 구성](#) 전에 섹션을 참조하세요.

자습서를 완료한 후 사용자는 로그인하고 생성한 웹 앱과 상호 작용할 수 있습니다.

1단계: 필요한 지원 리소스 생성

IAM Identity Center 디렉터리에 사용자를 추가해야 합니다. 또한 두 가지 역할이 필요합니다. 하나는 웹 앱의 자격 증명 보유자 역할로 사용하고 다른 하나는 Amazon S3 액세스 권한 부여를 구성하는 데 사용합니다. 자습서에서는 AWS 서비스에서 이러한 역할을 생성할 수 있도록 허용합니다.

사용자를 추가하려면

1. 에 로그인 AWS Management Console 하고 <https://console.aws.amazon.com/singlesignon/> AWS IAM Identity Center 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 사용자를 선택합니다.
3. 사용자 추가를 선택하고 사용자 세부 정보를 지정합니다.

사용자 이름, 이메일 주소 및 기타 필수 정보를 지정합니다. 암호 설정 지침이 포함된 이메일을 사용자에게 보내거나 일회용 암호를 생성하여 사용자에게 공유할 수 있습니다.

4. 다음을 선택하고 선택적으로 새 사용자를 하나 이상의 그룹에 할당합니다.
5. 다음을 선택하고 선택 사항을 검토합니다.

모든 것이 양호한 경우 사용자 추가를 선택하여 지정한 세부 정보로 새 사용자를 생성합니다.

이 자습서의 예제 사용자는 Bob Stiles, username bobstiles 및 이메일 주소 bobstiles@example.com입니다.

2단계: Transfer Family 웹 앱 생성

Transfer Family 웹 앱을 생성하려면

1. 에 로그인 AWS Management Console 하고 <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 웹 앱을 선택합니다.
3. 웹 앱 생성을 선택합니다.

인증 액세스의 경우 서비스는 사전 조건으로 설정한 AWS IAM Identity Center 인스턴스를 자동으로 찾습니다.

4. 권한 유형 창에서 새 서비스 역할 생성 및 사용을 선택합니다. 서비스가 자격 증명 보유자 역할을 생성합니다. 자격 증명 보유자 역할에는 인증된 사용자의 자격 증명이 세션에 포함됩니다.
5. 웹 앱 단위 창에서 기본값인 1을 수락하거나 필요한 경우 더 높은 값으로 조정합니다.
6. 웹 앱을 구성하는 데 도움이 되는 태그를 추가합니다. 자습서의 경우 키의 이름과 값에 대한 자습서 웹 앱을 입력합니다.

Tip

웹 앱 이름을 생성한 후 웹 앱 목록 페이지에서 직접 편집할 수 있습니다.

7. 다음을 선택하여 웹 앱 설계 페이지를 엽니다. 이 화면에서 다음 정보를 제공합니다.

선택적으로 웹 앱의 제목을 제공할 수 있습니다. 로고 및 파비콘의 이미지 파일을 업로드할 수도 있습니다.

- 페이지 제목의 경우 사용자가 웹 앱에 연결할 때 표시되는 브라우저 탭의 제목을 사용자 지정합니다. 페이지 제목에 아무 것도 입력하지 않으면 기본적으로 웹 앱 전송으로 설정됩니다.
- 로고의 경우 이미지 파일을 업로드합니다. 로고 이미지의 최대 파일 크기는 50KB입니다.
- favicon의 경우 이미지 파일을 업로드합니다. 파비콘의 최대 파일 크기는 20KB입니다.

8. 다음을 선택한 다음 웹 앱 생성을 선택합니다.

브랜드 경험을 제공하기 위해 사용자가 Transfer Family 웹 앱에 액세스할 수 있는 사용자 지정 URL을 제공할 수 있습니다. 자세한 내용은 [사용자 지정 URL로 액세스 엔드포인트 업데이트](#)를 참조하세요.

3단계: 버킷에 대한 교차 오리진 리소스 공유(CORS) 구성

웹 앱에서 사용하는 모든 버킷에 대해 교차 오리진 리소스 공유(CORS)를 설정해야 합니다. CORS 구성은가 버킷에 액세스하도록 허용할 오리진을 식별하는 규칙을 정의하는 문서입니다. CORS에 대한 자세한 내용은 [CORS\(Cross-Origin Resource Sharing\) 구성](#)을 참조하세요.

Amazon S3 버킷에 대한 교차 오리진 리소스 공유(CORS)를 설정하려면

1. 에 로그인 AWS Management Console 하고 <https://console.aws.amazon.com/s3/> Amazon S3 콘솔을 엽니다.
2. 왼쪽 탐색 패널에서 버킷을 선택하고 검색 대화 상자에서 버킷을 검색한 다음 권한 탭을 선택합니다.
3. CORS(Cross-Origin Resource Sharing)에서 편집을 선택하고 다음 코드에 붙여 넣습니다.
AccessEndpoint를 웹 앱의 실제 액세스 엔드포인트로 바꿉니다. 사용자가 웹 앱에 로그인하려고 할 때 오류가 발생하므로 후행 슬래시를 입력하지 마십시오.
 - 잘못된 예: <https://webapp-c7bf3423.transfer-webapp.us-east-2.on.aws/>
 - 올바른 예: <https://webapp-c7bf3423.transfer-webapp.us-east-2.on.aws>

버킷을 여러 웹 앱에 재사용하는 경우 해당 웹 앱 액세스 엔드포인트를 AllowedOrigins 목록에 추가합니다.

```
[
  {
    "AllowedHeaders": [
      "*"
    ],
```

```

    "AllowedMethods": [
        "GET",
        "PUT",
        "POST",
        "DELETE",
        "HEAD"
    ],
    "AllowedOrigins": [
        "https://AccessEndpoint"
    ],
    "ExposeHeaders": [
        "last-modified",
        "content-length",
        "etag",
        "x-amz-version-id",
        "content-type",
        "x-amz-request-id",
        "x-amz-id-2",
        "date",
        "x-amz-cf-id",
        "x-amz-storage-class",
        "access-control-expose-headers"
    ],
    "MaxAgeSeconds": 3000
}
]

```

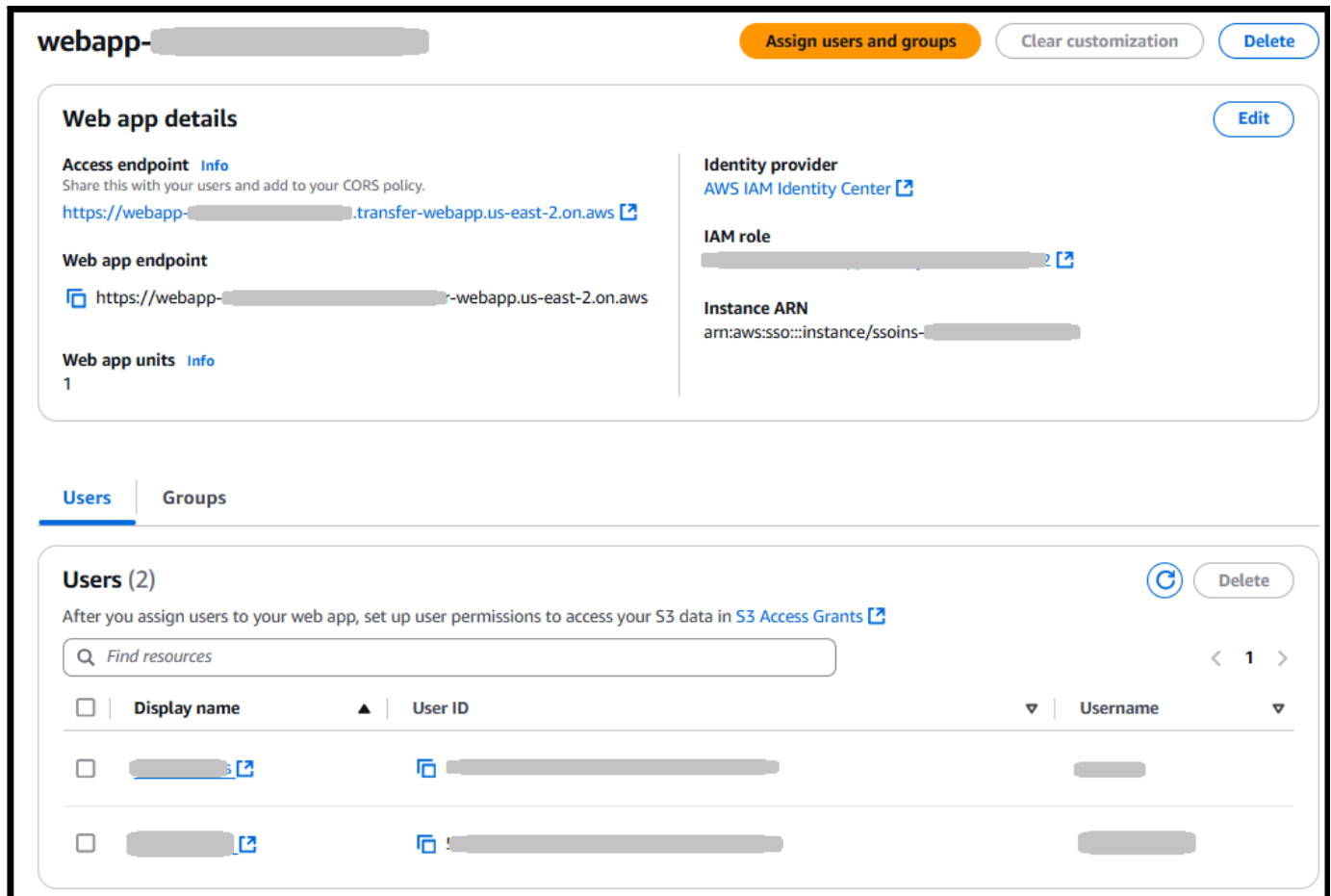
4. 변경 사항 저장을 선택하여 CORS를 업데이트합니다.

4단계: Transfer Family 웹 앱에 사용자 추가

IAM Identity Center에서 이전에 생성한 사용자를 추가합니다.

Transfer Family 웹 앱에 사용자를 할당하려면

1. 이전에 생성한 웹 앱으로 이동합니다.
2. 사용자 및 그룹 할당을 선택합니다.



3. IAM Identity Center에서 이전에 생성한 사용자를 할당하려면 기존 사용자 및 그룹 할당을 선택하고 다음을 선택합니다.
 - a. 표시 이름으로 사용자를 검색합니다. 검색 기준을 입력하기 전까지는 사용자가 표시되지 않습니다. 를 추가하려면 검색 상자에 bob을 **Bob Stiles**입력합니다. 사용자를 찾을 수 없는 경우 IAM Identity Center 관리 콘솔로 이동하여 사용자를 찾은 다음 여기에 표시 이름을 복사하여 붙여 넣습니다.
 - b. **Bob Stiles** 사용자를 선택한 다음 할당을 선택합니다.

5단계: Amazon S3에 위치 등록 및 액세스 권한 생성

웹 앱에 사용자를 할당한 후에는 버킷을 등록하고 해당 사용자에게 대한 액세스 권한을 생성해야 합니다.

Note

계속하려면 먼저 S3 Access Grants 인스턴스가 있어야 합니다. 자세한 내용은 Amazon Simple Storage Service 사용 설명서의 [S3 Access Grants 인스턴스 생성](#)을 참조하세요.

위치를 등록하고 액세스 권한을 생성하려면

1. 에 로그인 AWS Management Console 하고 <https://console.aws.amazon.com/s3/> Amazon S3 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 Access Grants를 선택합니다.
3. 세부 정보 보기를 선택하여 S3 Access Grants 인스턴스의 세부 정보를 확인합니다.
4. 위치 탭을 선택한 다음 위치 등록을 선택합니다.
5. 다음 정보를 입력합니다.
 - 범위에 버킷을 찾아보거나 버킷 이름과 선택적으로 접두사를 입력합니다. 범위는 문자열로 시작합니다 **s3://**.
 - IAM 역할의 경우 새 역할 생성을 선택하여 Amazon S3가 역할을 생성하도록 합니다. 이 역할은 S3 Access Grants가 지정된 위치 범위에 액세스할 수 있도록 허용합니다.

The screenshot shows the 'Register location' page in the AWS Management Console. The breadcrumb navigation at the top reads: Amazon S3 > Access Grants > US East (N. Virginia) us-east-1: default > Register location. The page title is 'Register location' with an 'Info' link. Below the title, a message states: 'Registering a location for S3 your S3 Access Grants instance assigns an IAM role the permission to get temporary credentials to access resources within the specified scope.'

The main content area is divided into two sections: 'Location' and 'IAM role', each with an 'Info' link.

Location section:

- Location scope:** A message says 'You can't edit the location scope after registering the location.'
- Scope:** A text input field contains 's3://'. To the right are 'View' and 'Browse S3' buttons.
- Format instructions:**
 - Format for all buckets: s3://
 - Format for bucket: s3://<bucket>
 - Format for prefix: s3://<bucket>/<prefix-with-path>

Permissions section:

- A message states: 'This IAM role allows S3 Access Grants to access your specified location scope.'
- A blue box contains a note: 'To ensure the role has proper permissions to access the location scope, consult the [list of permissions](#)'.

IAM role section:

- A message states: 'This IAM role allows S3 Access Grants to access your specified location scope.'
- There are three radio button options:
 - ☒ Create new role
 - ☐ Choose from existing IAM roles
 - ☐ Enter IAM role ARN

At the bottom right, there are 'Cancel' and 'Register location' buttons.

계속하려면 위치 등록을 선택합니다.

6. 권한 부여 탭을 선택한 다음 권한 부여 생성을 선택하고 다음 세부 정보를 제공합니다.
 - 위치에서 위치 찾아보기를 선택하고 이전 단계에서 등록한 위치를 선택합니다.
 - 하위 접두사에 *를 입력하여 액세스 권한이 전체 버킷에 적용됨을 나타냅니다.
 - 권한에서 읽기 및 쓰기를 선택합니다.
 - 피부여자 유형에서 IAM Identity Center의 디렉터리 자격 증명을 선택합니다.
 - 디렉터리 자격 증명 유형에서 사용자를 선택합니다.
 - IAM Identity Center 사용자/ID에서의 사용자 ID를 복사하여 붙여 넣습니다 **Bob Stiles**. 이 ID는 Transfer Family 웹 앱의 사용자 창에서 사용할 수 있습니다.
7. 권한 부여 생성을 선택합니다.

액세스 권한이 생성됩니다.

6단계: 사용자로 Transfer Family 웹 앱에 액세스

이제 웹 앱의 URL로 이동하여 앞서 할당한 사용자로 로그인합니다.

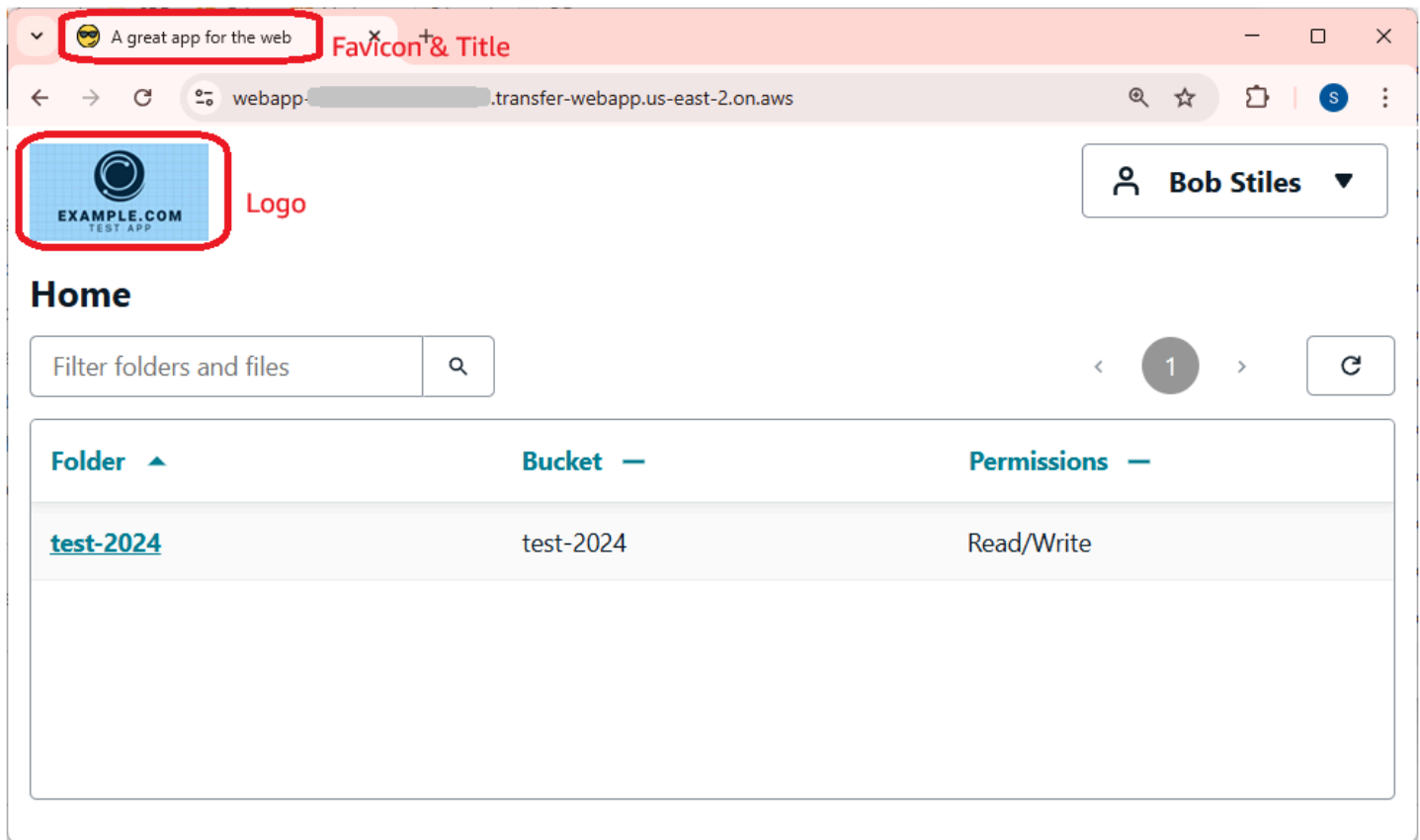
Transfer Family 웹 앱에 로그인하려면

1. 웹 앱으로 이동
2. 웹 앱 세부 정보 창에서 액세스 엔드포인트를 선택합니다.

The screenshot shows the AWS IAM console interface for a web application. At the top, there are buttons for 'Assign users and groups', 'Clear customization', and 'Delete'. The main section is titled 'webapp-'. Below this, there are two tabs: 'Users' and 'Groups'. The 'Users' tab is selected, showing a list of users. The 'Access endpoint' is highlighted with a red box, showing the URL 'https://webapp-...transfer-webapp.us-east-2.on.aws'. The 'Web app endpoint' is also highlighted, showing the same URL. The 'IAM role' is 'AWSTransferWebAppIdentityBearer-us-east-2'. The 'Instance ARN' is 'arn:aws:sso::instance/ssoins-...'. In the 'Users' section, a user named 'Bob Stiles' is listed with the username 'bobstiles', and this row is highlighted with a red box.

3. 로그인 화면에서 생성한 사용자를 입력한 후 다음을 **bobstiles** 선택합니다.
4. 생성 시 시스템이 사용자에게 할당한 암호를 입력하고 다음을 선택합니다.
5. 조직에 다중 인증(MFA)이 필요한 경우 지금 설정해야 합니다. 그렇지 않은 경우 6단계로 건너뛰니다.
 - a. MFA 디바이스를 등록하는 화면이 표시됩니다. 사용 가능한 옵션 중 하나를 선택하고 다음을 선택합니다.
 - b. 이 사용자에게 대해 MFA를 구성하는 데 필요한 단계를 수행합니다. 단계는 선택한 MFA 옵션에 따라 다릅니다.
 - c. 사용자에게 대해 새 암호를 설정해야 할 수 있습니다. 필요한 경우 지금 설정해야 합니다. 시스템에서 구성한 새 MFA 자격 증명을 사용하여 다시 로그인해야 할 수도 있습니다.

사용 시 다음과 비슷한 화면이 표시되어야 합니다. 이 스크린샷에는 파비콘 및 로고에 대한 사용자 지정이 포함되어 있습니다.



다음 단계

표준 S3 버킷 액세스를 기본 Transfer Family 웹 앱을 성공적으로 설정했습니다. 사용자가 한 버킷에서 다운로드하여 다른 버킷으로 업로드할 수 있도록 허용하는 등 버킷 권한을 더 세밀하게 제어해야 하는 경우 섹션을 참조하세요 [자습서: 선택적 다중 버킷 액세스를 사용하여 AWS Transfer Family 웹 앱 설정](#).

Okta를 웹 앱의 자격 증명 공급자로 통합

외부 자격 증명 공급자를 Transfer Family 웹 앱과 통합할 수 있습니다. 이 섹션에서는 Okta를 자격 증명 공급자로 설정하는 방법을 설명합니다.

1. Okta에서 사용자, 그룹 및 애플리케이션을 생성합니다. 이 작업을 수행하는 방법에 대한 자세한 내용은 [Okta 및 IAM Identity Center를 사용하여 SAML 및 SCIM 구성](#)을 참조하세요.

AWS IAM Identity Center

General Sign On Provisioning Import Assignments **Push Groups**

Push Groups to AWS IAM Identity Center

Push Groups Refresh App Groups Bulk Edit Search...

Pushed Groups	Group in Okta	Group in AWS IAM Identity Center	Last Push	Push Status
All	Examplegroup No description	Examplegroup No description	March 4, 2025 at 3:02:08 PM GMT+9	Active
Errors				
By name				
By rule				

2. Okta를 연결하고 Okta에서 로 사용자 및 그룹을 가져옵니다 AWS IAM Identity Center. [Okta 및 IAM Identity Center](#)를 사용하여 SAML 및 SCIM 구성의 1~4단계를 따릅니다.

IAM Identity Center

Managing Instance
ssols-6684735b83d16df8

Dashboard
Users
Groups
Settings

Multi-account permissions
AWS accounts
Permission sets

Your identity source is currently configured as 'External identity provider'. To add new users or edit their attributes, you must do this using your external identity provider (IdP).

Users (2) Delete users

Users listed here can sign in to the AWS access portal to access AWS accounts and assigned cloud applications. [Learn more](#)

Username Find users < 1 >

	Username	Display name	Status	Created by
☐	Exampleuser	Example User	Enabled	SCIM
☐	johnstiles	John Stiles	Enabled	Manual

IAM Identity Center

Managing Instance
ssols-6684735b83d16df8

Dashboard
Users
Groups
Settings

Multi-account permissions
AWS accounts
Permission sets

Your identity source is currently configured as 'External identity provider'. To add new groups or edit their memberships, you must do this using your external identity provider.

Groups (2) Delete group

With groups, you can grant or deny permissions to groups of workforce users, rather than having to apply those permissions to each user. [Learn more](#)

Find groups by group name < 1 >

	Group name	Users	Description	Created by
☐	Examplegroup	1 user	-	SCIM
☐	dev	1 user	-	Manual

3. IAM Identity Center의 ID 소스가 SAML 2.0인지 확인합니다.

Settings

Details

Configure your identity source and multi-factor authentication settings for use when managing access to your AWS accounts, resources, and cloud applications.

Instance name - Edit 	Instance ID ssoins-	Organization ID AWS Organizations not enabled
Region US East (Ohio) us-east-2	Date created Thursday, November 14, 2024 at 4:11:19 PM EST	Instance ARN arn:aws:sso::instance/ssoins-

[Identity source](#) | [Authentication](#) | [Management](#) | [Tags](#)

Identity source

Choose the directory where you want to manage your users and groups. [Learn More](#)

Identity source
External identity provider

Authentication method SAML 2.0	Provisioning method SCIM

AWS access portal URL
[https://.awsapps.com/start](#)

Identity store ID
d-

Issuer URL
[https://identitycenter.amazonaws.com/ssoins-](#)

[Actions](#) ▼

4. 에 설명된 대로 사용자와 그룹을 할당합니다 [4단계: Transfer Family 웹 앱에 사용자 추가](#).

5. 웹 앱에 로그인할 때 사용자가 MFA를 사용할 필요가 없도록 하려면 Okta에서 다음 단계를 수행합니다.

- Okta 관리자 콘솔에서 [애플리케이션] - [애플리케이션]에 AWS IAM Identity Center 액세스하고 애플리케이션을 선택합니다.
- 로그인 탭에서 [사용자 인증] - 편집을 선택합니다.
- 암호만 선택합니다.

자습서의 다른 모든 단계를 수행한 후 사용자는 웹 브라우저에서 웹 앱의 액세스 엔드포인트로 이동하여 Transfer Family 웹 앱에 액세스할 수 있어야 합니다.

자습서: 선택적 다중 버킷 액세스를 사용하여 AWS Transfer Family 웹 앱 설정

이 자습서에서는 단일 사용자에게 대한 특정 Amazon S3 버킷 권한으로 Transfer Family 웹 앱을 구성하는 방법을 안내합니다. 보안을 유지하면서 사용자가 한 버킷에서 다운로드하여 다른 버킷으로 업로드할 수 있는 솔루션을 설정하는 방법을 알아봅니다. 이 자습서는 기본 자습서에서 다루는 개념을 기반으로 하는 고급 자습서입니다. AWS Transfer Family 웹 앱을 처음 사용하는 경우로 시작하는 것이 좋습니다. [자습서: 기본 Transfer Family 웹 앱 설정](#).

사전 조건

이 튜토리얼을 시작하려면 다음이 필요합니다.

- AWS Transfer Family 웹 앱과 동일한 리전에 구성된 IAM Identity Center. 는 AWS 모든 리전에 대해 AWS 계정당 하나의 IAM Identity Center 인스턴스만 허용합니다.
- IAM Identity Center에 구성된 사용자가 한 명 이상 있습니다.
- S3 버킷 2개: 다운로드용 버킷 1개와 업로드용 버킷 1개.

Note

이 자습서에서는 기본 웹 앱 자습서와 많은 사전 조건을 공유합니다. IAM Identity Center 설정 및 사용자 생성에 대한 자세한 내용은 섹션을 참조하세요. [자습서: 기본 Transfer Family 웹 앱 설정](#).

1단계: Transfer Family 웹 앱 생성

Transfer Family 웹 앱을 생성하려면

1. 에 로그인 AWS Management Console 하고 <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 웹 앱을 선택합니다.
3. 웹 앱 생성을 선택합니다.

인증 액세스의 경우 서비스는 사전 조건으로 설정한 AWS IAM Identity Center 인스턴스를 자동으로 찾습니다.

4. 권한 유형 창에서 새 서비스 역할 생성 및 사용을 선택합니다. 서비스가 자격 증명 보유자 역할을 생성합니다. 자격 증명 보유자 역할에는 인증된 사용자의 자격 증명이 세션에 포함됩니다.
5. 웹 앱 단위 창에서 기본값인 1을 수락하거나 필요한 경우 더 높은 값으로 조정합니다.
6. 웹 앱을 구성하는 데 도움이 되는 태그를 추가합니다. 자습서의 경우 키의 이름과 값에 대한 자습서 웹 앱을 입력합니다.

 Tip

웹 앱 이름을 생성한 후 웹 앱 목록 페이지에서 직접 편집할 수 있습니다.

7. 다음을 선택하여 웹 앱 설계 페이지를 엽니다. 이 화면에서 다음 정보를 제공합니다.

선택적으로 웹 앱의 제목을 제공할 수 있습니다. 로고 및 파비콘의 이미지 파일을 업로드할 수도 있습니다.

- 페이지 제목의 경우 사용자가 웹 앱에 연결할 때 표시되는 브라우저 탭의 제목을 사용자 지정합니다. 페이지 제목에 아무 것도 입력하지 않으면 기본적으로 웹 앱 전송으로 설정됩니다.
- 로고의 경우 이미지 파일을 업로드합니다. 로고 이미지의 최대 파일 크기는 50KB입니다.
- favicon의 경우 이미지 파일을 업로드합니다. 파비콘의 최대 파일 크기는 20KB입니다.

8. 다음을 선택한 다음 웹 앱 생성을 선택합니다.

2단계: S3 액세스를 위한 IAM 역할 구성

두 개의 IAM 역할을 생성해야 합니다. 하나는 첫 번째 버킷에 대한 다운로드 전용 액세스 권한이 있고 다른 하나는 두 번째 버킷에 대한 업로드 전용 액세스 권한이 있습니다.

두 역할에 대한 신뢰 정책

두 IAM 역할에 대해 다음 신뢰 정책을 사용합니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AccessGrantsTrustPolicy",
      "Effect": "Allow",
      "Principal": {
        "Service": "access-grants.s3.amazonaws.com"
      }
    }
  ]
}
```

```

    },
    "Action": [
        "sts:AssumeRole",
        "sts:SetSourceIdentity",
        "sts:SetContext"
    ]
}
]
}

```

다운로드 버킷에 대한 IAM 정책

다운로드 버킷에 대한 읽기 전용 액세스를 위해 다음 정책을 사용하여 IAM 역할을 생성합니다.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ObjectLevelReadPermissions",
      "Effect": "Allow",
      "Action": [
        "s3:GetObject",
        "s3:GetObjectVersion",
        "s3:GetObjectAcl",
        "s3:GetObjectVersionAcl",
        "s3:ListMultipartUploadParts",
        "s3:ListBucket"
      ],
      "Resource": [
        "arn:aws:s3:::amzn-s3-demo-bucket1/*",
        "arn:aws:s3:::amzn-s3-demo-bucket1"
      ]
    }
  ]
}

```

Important

amzn-s3-demo-bucket1을 다운로드 버킷의 실제 이름으로 바꿉니다.

버킷 업로드를 위한 IAM 정책

업로드 버킷에 대한 쓰기 액세스를 위해 다음 정책을 사용하여 다른 IAM 역할을 생성합니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ObjectLevelWritePermissions",
      "Effect": "Allow",
      "Action": [
        "s3:PutObject",
        "s3:PutObjectAcl",
        "s3:PutObjectVersionAcl",
        "s3:DeleteObject",
        "s3:DeleteObjectVersion",
        "s3:AbortMultipartUpload",
        "s3:ListBucket"
      ],
      "Resource": [
        "arn:aws:s3:::amzn-s3-demo-bucket2/*",
        "arn:aws:s3:::amzn-s3-demo-bucket2"
      ]
    }
  ]
}
```

Important

amzn-s3-demo-bucket2를 업로드 버킷의 실제 이름으로 바꿉니다.

3단계: S3 Access Grants 설정

1. 에서 S3 콘솔을 엽니다 <https://console.aws.amazon.com/s3/>.
2. 탐색 창에서 Access Grants를 선택합니다.
3. S3 Access Grants 인스턴스 생성을 클릭합니다.
4. IAM Identity Center 인스턴스 추가 옵션을 선택하고 ID 센터 인스턴스 ARN을 입력합니다.

5. 다음을 클릭한 다음 취소를 클릭하여 추가 단계를 진행하지 않고 S3 Access Grants 인스턴스 생성을 완료합니다.

이 단계에서는 S3 Access Grants 인스턴스를 생성합니다. 이제 위치를 등록하고 액세스 권한을 생성합니다.

4단계: S3 버킷 위치 등록

두 S3 버킷을 S3 Access Grants의 위치로 등록합니다.

1. S3 Access Grants 콘솔에서 위치로 이동하여 위치 등록을 클릭합니다.
2. 위치 범위에서 다운로드할 특정 S3 버킷(amzn-s3-demo-bucket1)을 선택합니다.
3. IAM 역할을 선택하라는 메시지가 표시되면 이전에 생성한 IAM 역할 다운로드를 선택합니다.
4. 등록 프로세스를 완료합니다.
5. 이 프로세스를 반복하여 업로드 버킷(amzn-s3-demo-bucket2)을 등록하고 메시지가 표시되면 업로드 IAM 역할을 선택합니다.

5단계: 액세스 권한 부여 생성

등록된 각 위치에 대해 하나씩 두 개의 권한 부여를 생성합니다.

1. S3 Access Grants 콘솔에서 권한 부여로 이동하여 권한 부여 생성을 클릭합니다.
2. 위치에서 위치 찾아보기를 클릭하고 다운로드 버킷 위치(amzn-s3-demo-bucket1)를 선택합니다.
3. 하위 접두사(선택 사항)에서 *를 입력하여 전체 버킷에 대한 액세스를 허용하거나와 같은 경로를 지정folder1/folder2/*하여 특정 접두사에 대한 액세스를 제한합니다.

를 사용하면 권한 부여 범위가 로 * 설정s3://bucket-name/*되므로 전체 버킷에 액세스할 수 있습니다. 특정 접두사에 대한 액세스만 허용하려면 권한 부여 범위를 로 folder1/folder2/* 설정하는와 같은 경로를 입력합니다s3://bucket-name/folder1/folder2/*.

4. 권한 및 액세스에서 다운로드 버킷에 대한 읽기를 선택합니다.
5. 피부여자 유형에서 IAM Identity Center의 디렉터리 자격 증명을 선택합니다.
6. IAM 보안 주체 유형에서 사용자를 선택하고 IAM Identity Center 사용자의 사용자 ID를 입력합니다.
7. 권한 부여 생성 프로세스를 완료합니다.

- 이 프로세스를 반복하여 업로드 버킷(amzn-s3-demo-bucket2)에 대한 권한 부여를 생성하되 권한에 대해 읽기-쓰기를 선택합니다.

6단계: S3 버킷에 대한 CORS 정책 구성

WebApp을 통한 액세스를 허용하도록 두 S3 버킷에 대한 CORS 정책을 구성합니다. AWS Transfer Family

- S3 콘솔을 열고 다운로드 버킷(amzn-s3-demo-bucket1)으로 이동합니다.
- 권한 탭을 선택합니다.
- CORS(Cross-Origin Resource Sharing) 섹션으로 스크롤하여 편집을 클릭합니다.
- 다음 CORS 구성을 추가하여 *WebAppEndpoint*를 실제 WebApp 엔드포인트 URL로 바꿉니다.

AWS Transfer Family 콘솔의 WebApps. `https://webapp-*****.transfer-webapp.us-west-2.on.aws` 비슷하게 보일 것입니다.

```
[
  {
    "AllowedHeaders": [
      "*"
    ],
    "AllowedMethods": [
      "GET",
      "PUT",
      "POST",
      "DELETE",
      "HEAD"
    ],
    "AllowedOrigins": [
      "https://WebAppEndpoint"
    ],
    "ExposeHeaders": [
      "last-modified",
      "content-length",
      "etag",
      "x-amz-version-id",
      "content-type",
      "x-amz-request-id",
      "x-amz-id-2",
      "date",
```

```
        "x-amz-cf-id",  
        "x-amz-storage-class",  
        "access-control-expose-headers"  
    ],  
    "MaxAgeSeconds": 3000  
}  
]
```

5. 변경 사항 저장을 클릭합니다.
6. 업로드 버킷(amzn-s3-demo-bucket2)에 대해 프로세스를 반복합니다.

7단계: 구성 테스트

1. AWS Transfer Family 웹 앱 URL을 엽니다. AWS Transfer Family 콘솔의 액세스 엔드포인트 필드의 WebApps에서이 URL을 찾을 수 있습니다.
2. 액세스 권한 부여로 구성된 IAM Identity Center 사용자 자격 증명을 사용하여 로그인합니다.
3. 로그인한 후 홈 페이지에 두 S3 위치가 모두 표시됩니다.
4. 다운로드 버킷(amzn-s3-demo-bucket1)으로 이동하여 파일을 다운로드할 수 있지만 업로드할 수 없는지 확인합니다.
5. 업로드 버킷(amzn-s3-demo-bucket2)으로 이동하여 파일을 업로드할 수 있는지 확인합니다.

결론

단일 사용자에게 대한 선택적 S3 버킷 액세스를 AWS Transfer Family WebApp을 성공적으로 구성했습니다. 이 설정을 통해 사용자는 IAM 역할 및 S3 Access Grants를 통해 보안을 유지하면서 한 버킷에서 다운로드하고 다른 버킷으로 업로드할 수 있습니다.

이 접근 방식은 각 사용자에게 대해 S3 Access Grants에 추가 권한을 생성하여 여러 사용자에게 확장할 수 있으므로 버킷 액세스 권한을 세밀하게 제어할 수 있습니다. 기본 웹 앱 설정에 대한 자세한 내용은 [섹션을 참조하세요](#) [자습서: 기본 Transfer Family 웹 앱 설정](#).

SFTP, FTPS 또는 FTP 서버 엔드포인트 구성

이 주제에서는 하나 이상의 SFTP, FTPS 및 FTP 프로토콜을 사용하는 AWS Transfer Family 서버 엔드포인트를 생성하고 사용하는 방법에 대한 세부 정보를 제공합니다.

주제

- [자격 증명 공급자 옵션](#)
- [AWS Transfer Family 엔드포인트 유형 매트릭스](#)
- [SFTP, FTPS 또는 FTP 서버 엔드포인트 구성](#)
- [FTP 및 FTPS Network Load Balancer 고려 사항](#)
- [클라이언트를 사용하여 서버 엔드포인트를 통한 파일 전송](#)
- [서버 엔드포인트에 대한 사용자 관리](#)
- [논리적 디렉터리를 사용하여 Transfer Family 디렉터리 구조를 단순화합니다.](#)
- [Transfer Family를 사용하여 FSx for NetApp ONTAP 파일 시스템에 액세스](#)

자격 증명 공급자 옵션

AWS Transfer Family 는 사용자를 인증하고 관리하기 위한 몇 가지 방법을 제공합니다. 다음 표에서는 Transfer Family와 함께 사용할 수 있는 ID 공급자를 비교합니다.

작업	AWS Transfer Family 서비스 관리형	AWS Managed Microsoft AD	Amazon API Gateway	AWS Lambda
지원되는 프로토콜	SFTP	SFTP, FTPS, FTP	SFTP, FTPS, FTP	SFTP, FTPS, FTP
키 기반 인증	예	아니요	예	예
암호 인증	아니요	예	예	예
AWS Identity and Access Management (IAM) 및 POSIX	예	예	예	예

작업	AWS Transfer Family 서비스 관리형	AWS Managed Microsoft AD	Amazon API Gateway	AWS Lambda
논리적 홈 디렉터리	예	예	예	예
파라미터화된 액세스(사용자 이름 기반)	예	예	예	예
임시 액세스 구조	예	아니요	예	예
AWS WAF	아니요	아니요	예	아니요

참고:

- IAM은 Amazon S3 지원 스토리지에 대한 액세스를 통제하는 데 사용되고 POSIX는 Amazon EFS에 사용됩니다.
- 임시는 런타임에 사용자 프로필을 전송할 수 있는 기능을 의미합니다. 예컨대, 사용자 이름을 변수로 전달하여 사용자를 홈 디렉터리에 연결할 수 있습니다.
- 에 대한 자세한 내용은 단원을 AWS WAF참조하십시오 [웹 애플리케이션 방화벽 추가](#).
- Microsoft Entra ID(이전 Azure AD)와 통합된 Lambda 함수를 Transfer Family 자격 증명 공급자로 사용하는 방법을 설명하는 블로그 게시물이 있습니다. 자세한 내용은 [Azure Active Directory AWS Transfer Family 를 사용하여 인증 및 섹션을 AWS Lambda](#)참조하세요.
- 사용자 지정 자격 증명 공급자를 사용하는 Transfer Family 서버를 빠르게 배포하는 데 도움이 되는 여러 CloudFormation 템플릿을 제공합니다. 자세한 내용은 [Lambda 함수 템플릿](#)을 참조하세요.

다음 절차에서 SFTP 지원 서버, FTPS 지원 서버, FTP 지원 서버 또는 AS2 지원 서버를 생성할 수 있습니다.

다음 단계

- [SFTP 지원 서버 생성](#)
- [FTPS 지원 서버 생성](#)
- [FTP 지원 서버 생성](#)

- [AS2 구성](#)

AWS Transfer Family 엔드포인트 유형 매트릭스

Transfer Family 서버를 생성하는 경우 사용할 엔드포인트 타입을 선택합니다. 다음 표에서는 각 엔드포인트 타입의 특성에 대해 설명합니다.

엔드포인트 타입 매트릭스

기능	퍼블릭	VPC - 인터넷	VPC - 내부형	VPC_엔드포인트 (폐지됨)
지원되는 프로토콜	SFTP	SFTP, FTPS, AS2	SFTP, FTP, FTPS, AS2	SFTP
액세스	인터넷을 통해 이 엔드포인트 타입에는 VPC에 특별한 구성이 필요하지 않습니다.	인터넷을 통해, 그리고 또는 Direct Connect VPN을 통한 온프레미스 데이터 센터와 같은 VPC 및 VPC 연결 환경 내에서.	Direct Connect 또는 VPN을 통한 온프레미스 데이터 센터와 같은 VPC 및 VPC 연결 환경 내에서.	Direct Connect 또는 VPN을 통한 온프레미스 데이터 센터와 같은 VPC 및 VPC 연결 환경 내에서.
고정 IP 주소	고정 IP 주소는 연결할 수 없습니다.는 변경될 수 있는 IP 주소를 AWS 제공합니다.	탄력적 IP 주소를 엔드포인트에 연결할 수 있습니다. 이는 AWS에서 소유한 IP 주소일 수도 있고 자체 IP 주소일 수도 있습니다(자체 IP 주소 사용). 엔드포인트에 연결된 탄력적 IP 주소는 변경되지 않습니다.	엔드포인트에 연결된 사설 IP 주소는 변경되지 않습니다.	엔드포인트에 연결된 사설 IP 주소는 변경되지 않습니다.

기능	퍼블릭	VPC - 인터넷	VPC - 내부형	VPC_엔드포인트 (폐지됨)
		서버에 연결된 사 설 IP 주소도 변 경되지 않습니다.		

기능	퍼블릭	VPC - 인터넷	VPC - 내부형	VPC_엔드포인트 (폐지됨)
소스 IP 허용 목록	이 엔드포인트 타입은 소스 IP 주소별 허용 목록을 지원하지 않습니다. 엔드포인트는 공개적으로 액세스할 수 있으며 포트 22를 통해 트래픽을 수신합니다.  Note VPC 호스팅 엔드포인트의 경우 SFTP Transfer Family 서버는 포트 22(기본값), 2222, 2223 또는 22000을 통해 작동할 수 있습니다.	소스 IP 주소별 액세스를 허용하려면 서버 엔드포인트에 연결된 보안 그룹과 엔드포인트가 있는 서브넷에 연결된 네트워크 ACL을 사용할 수 있습니다.	소스 IP 주소별 액세스를 허용하려면 서버 엔드포인트에 연결된 보안 그룹과 엔드포인트가 있는 서브넷에 연결된 네트워크 ACL(액세스 통제 목록)을 사용할 수 있습니다.	소스 IP 주소별 액세스를 허용하려면 서버 엔드포인트에 연결된 보안 그룹과 엔드포인트가 있는 서브넷에 연결된 네트워크 ACL을 사용할 수 있습니다.

기능	퍼블릭	VPC - 인터넷	VPC - 내부형	VPC_엔드포인트 (폐지됨)
클라이언트 방화벽 허용 목록	서버의 DNS 이름을 허용해야 합니다. IP 주소는 변경될 수 있으므로 클라이언트 방화벽 허용 목록에 IP 주소를 사용하지 마세요.	서버의 DNS 이름 또는 서버에 연결된 탄력적 IP 주소를 허용할 수 있습니다.	엔드포인트의 프라이빗 IP 주소 또는 DNS 이름을 허용할 수 있습니다.	엔드포인트의 프라이빗 IP 주소 또는 DNS 이름을 허용할 수 있습니다.
IP 주소 유형	IPv4(기본값) 또는 듀얼 스택 (IPv4 및 IPv6)	IPv4 전용(듀얼 스택은 지원되지 않음)	IPv4(기본값) 또는 듀얼 스택 (IPv4 및 IPv6)	IPv4 전용(듀얼 스택은 지원되지 않음)

Note

VPC_ENDPOINT 엔드포인트 타입은 이제 더 이상 사용되지 않으며 새 서버를 만드는 데 사용할 수 없습니다. 를 사용하는 대신 이전 표에 설명된 대로 내부 또는 인터넷 연결로 사용할 수 있는 VPC 엔드포인트 유형(EndpointType=VPC)을 EndpointType=VPC_ENDPOINT 사용합니다.

- 사용 중단에 대한 자세한 내용은 섹션을 참조하세요 [VPC_ENDPOINT 사용 중단](#).
- VPC 엔드포인트 권한 관리에 대한 자세한 내용은 섹션을 참조하세요 [Transfer Family 서버에 대한 VPC 엔드포인트 액세스 제한](#).

AWS Transfer Family 서버의 보안 상태를 강화하려면 다음 옵션을 고려해 보세요.

- 내부 액세스 권한이 있는 VPC 엔드포인트를 사용하면 서버는 Direct Connect 또는 VPN을 통한 온 프레미스 데이터 센터와 같은 VPC 또는 VPC 연결 환경 내의 클라이언트만 액세스할 수 있습니다.
- 클라이언트가 인터넷을 통해 엔드포인트에 액세스하도록 허용하고 서버를 보호하려면 인터넷 연결 액세스가 가능한 VPC 엔드포인트를 사용하세요. 그런 다음 사용자 클라이언트를 호스팅하는 특정 IP 주소로부터의 트래픽만 허용하도록 VPC의 보안 그룹을 수정하세요.

- 암호 기반 인증이 필요하고 서버에서 맞춤 ID 공급자를 사용하는 경우 암호 정책을 통해 사용자가 취약한 암호를 만들지 못하도록 하고 로그인 시도 실패 횟수를 제한하는 것이 좋습니다.
- AWS Transfer Family 는 관리형 서비스이므로 셸 액세스를 제공하지 않습니다. Transfer Family 서버에서는 기본 SFTP 서버에 직접 액세스하여 OS 기본 명령을 실행할 수 없습니다.
- 내부 액세스가 가능한 VPC 엔드포인트 앞에서 Network Load Balancer를 사용하세요. 로드 밸런서의 리스너 포트를 포트 22에서 다른 포트로 변경합니다. 이렇게 하면 포트 스캐너와 봇이 서버를 탐색할 위험을 줄일 수 있지만 제거하지는 못합니다. 포트 22가 스캔에 가장 많이 사용되기 때문입니다. 자세한 내용은 [이제 Network Load Balancer가 보안 그룹을 지원한다는 블로그 게시물을 참조하세요](#).

Note

Network Load Balancer를 사용하는 경우 AWS Transfer Family CloudWatch 로그에는 실제 클라이언트 IP 주소가 아닌 NLB의 IP 주소가 표시됩니다.

SFTP, FTPS 또는 FTP 서버 엔드포인트 구성

AWS Transfer Family 서비스를 사용하여 파일 전송 서버를 생성할 수 있습니다. 다음 파일 전송 프로토콜을 사용할 수 있습니다.

- Secure Shell(SSH) 파일 전송 프로토콜(SFTP) - SSH를 통한 파일 전송. 자세한 내용은 [the section called “SFTP 지원 서버 생성”](#)을 참조하세요.

Note

SFTP Transfer Family 서버를 생성하는 AWS CDK 예제를 제공합니다. 이 예는 TypeScript를 사용하는 데, [여기](#)의 GitHub에서 입수할 수 있습니다.

- FTPS(파일 전송 프로토콜 보안) - TLS 암호화를 사용한 파일 전송. 자세한 내용은 [the section called “FTPS 지원 서버 생성”](#)를 참조하세요.
- FTP(파일 전송 프로토콜) - 암호화되지 않은 파일 전송. 자세한 내용은 [the section called “FTP 지원 서버 생성”](#)를 참조하세요.
- 적용성 보고서 2(AS2) - 구조화된 B2B 데이터 전송을 위한 파일 전송. 자세한 내용은 [the section called “AS2 구성”](#)을 참조하세요. AS2의 경우 데모용으로 CloudFormation 스택을 빠르게 생성할 수 있습니다. 이 절차는 [템플릿을 사용하여 데모 Transfer Family AS2 스택을 생성하세요](#)에 설명되어 있습니다.

여러 프로토콜로 서버를 만들 수 있습니다.

Note

같은 서버 엔드포인트에 여러 프로토콜을 사용하도록 설정한 상태에서 여러 프로토콜에서 동일한 사용자 이름을 사용하여 액세스를 제공하려는 경우 ID 공급자에 각 프로토콜별 보안 인증이 설정되어 있으면 그렇게 할 수 있습니다. FTP의 경우 SFTP 및 FTPS와 별도의 자격 증명을 유지하는 것이 좋습니다. 이는 SFTP 및 FTPS와 달리 FTP는 자격 증명을 일반 텍스트로 전송하기 때문입니다. FTP 자격 증명을 SFTP 또는 FTPS에서 분리하면 FTP 자격 증명에 공유되거나 노출되더라도 SFTP 또는 FTPS를 사용하는 워크로드의 보안을 유지할 수 있습니다.

서버를 생성할 때 특징을 선택하여 해당 서버에 할당된 사용자의 파일 작업 요청을 AWS 리전 수행합니다. 서버에 하나 이상의 프로토콜을 할당하는 동시에 다음 ID 공급자 타입 중 하나를 할당합니다:

- SSH 키를 사용한 서비스 관리. 자세한 내용은 [서비스 관리형과 작업](#)을 참조하세요.
- AWS Directory Service for Microsoft Active Directory (AWS Managed Microsoft AD) 이 방법을 사용하면 Microsoft Active Directory 그룹을 통합하여 Transfer Family 서버에 대한 액세스를 제공할 수 있습니다. 자세한 내용은 [Microsoft Active Directory용 AWS Directory Service 사용](#)을 참조하세요.
- 사용자 지정 자격 증명 공급자입니다. Transfer Family는 [사용자 지정 자격 증명 공급자와 작업](#) 주제에 설명된 대로 사용자 지정 자격 증명 공급자를 사용하기 위한 몇 가지 옵션을 제공합니다.

또한 기본 서버 엔드포인트를 사용하여 엔드포인트 타입(공개적으로 액세스 가능 또는 VPC 호스팅)과 호스트 이름을 서버에 할당하거나, Amazon Route 53 서비스를 사용하거나 선택한 도메인 이름 시스템(DNS) 서비스를 사용하여 맞춤 호스트 이름을 할당합니다. 서버 호스트 이름은 서버 호스트가 생성된 AWS 리전 에서 고유해야 합니다.

또한 Amazon CloudWatch 로깅 역할을 할당하여 CloudWatch 로그에 이벤트를 푸시하고, 서버에서 사용할 수 있도록 활성화된 암호화 알고리즘이 포함된 보안 정책을 선택하고, 카-값 쌍인 태그의 형태로 서버에 메타데이터를 추가할 수 있습니다.

Important

인스턴스화한 서버와 데이터 전송에 대한 비용이 발생합니다. Transfer Family 사용 비용 견적을 AWS Pricing Calculator 얻는 데 사용할 요금 및에 대한 자세한 내용은 [AWS Transfer Family 요금](#)을 참조하세요.

SFTP 지원 서버 생성

Secure Shell(SSH) File Transfer 프로토콜(SFTP)은 인터넷을 통한 안전한 데이터 전송에 사용되는 네트워크 프로토콜입니다. 이 프로토콜은 SSH의 전체 보안 및 인증 기능을 지원합니다. 금융 서비스, 의료, 소매 및 광고와 같은 다양한 산업의 비즈니스 파트너들 사이의 민감한 정보를 비롯한 데이터를 교환하는 데 널리 사용됩니다.

다음 사항에 유의하세요.

- Transfer Family용 SFTP 서버는 포트 22를 통해 작동합니다. VPC 호스팅 엔드포인트의 경우 SFTP Transfer Family 서버는 포트 2222, 2223 또는 22000을 통해 작동할 수도 있습니다. 자세한 내용은 [Virtual Private Cloud\(VPC\)에 서버 생성](#)을 참조하세요.
- 퍼블릭 엔드포인트는 보안 그룹을 통한 트래픽을 제한할 수 없습니다. Transfer Family 서버에서 보안 그룹을 사용하려면 설명된 대로 가상 프라이빗 클라우드(VPC) 내에서 서버의 엔드포인트를 호스팅해야 합니다. [Virtual Private Cloud\(VPC\)에 서버 생성](#).

참고 항목

- SFTP Transfer Family 서버를 생성하는 AWS CDK 예제를 제공합니다. 이 예는 TypeScript를 사용하는 데, [여기](#)의 GitHub에서 입수할 수 있습니다.
- VPC 내부에 Transfer Family 서버를 배포하는 방법에 대한 연습은 [IP 허용 목록을 사용하여 AWS Transfer Family 서버 보안을 참조하세요](#).

SFTP 지원 서버를 생성하려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 열고 탐색 창에서 서버를 선택한 다음 서버 생성을 선택합니다.
2. 프로토콜 선택에서 SFTP를 선택하고 다음을 선택합니다.
3. 자격 증명 공급자 선택에서 사용자 액세스를 관리하는 데 사용할 자격 증명 공급자를 선택합니다. 다음과 같은 옵션이 있습니다:
 - 서비스 관리형 - 사용자 ID와 키를 저장합니다 AWS Transfer Family.
 - AWS Directory Service for Microsoft Active Directory - 엔드포인트에 액세스할 수 있는 Directory Service 디렉터리를 제공합니다. 그러면 Active Directory에 저장된 자격 증명을 사용하여 사용자를 인증할 수 있습니다. 자격 AWS Managed Microsoft AD 증명 공급자 작업에 대한 자세한 내용은 섹션을 참조하세요 [Microsoft Active Directory용 AWS Directory Service 사용](#).

 Note

- 교차 계정 및 공유 디렉터리는에서 지원되지 않습니다 AWS Managed Microsoft AD.
- Directory Service를 자격 증명 공급자로 사용하여 서버를 설정하려면 몇 가지 Directory Service 권한을 추가해야 합니다. 자세한 내용은 [사용을 시작하기 전에 AWS Directory Service for Microsoft Active Directory](#)을 참조하세요.

- 맞춤 ID 공급자 – 다음 옵션 중 하나를 선택합니다.
- AWS Lambda 를 사용하여 자격 증명 공급자 연결 - Lambda 함수가 지원하는 기존 자격 증명 공급자를 사용할 수 있습니다. 귀하는 Lambda 함수의 명칭을 제공합니다. 자세한 내용은 [AWS Lambda 를 사용하여 자격 증명 공급자 통합](#)를 참조하세요.
- Amazon API Gateway를 사용하는 자격 증명 공급자 연결 - Lambda 함수가 지원하는 API Gateway 방법을 생성하여 자격 증명 공급자로 사용할 수 있습니다. 사용자는 Amazon API Gateway URL과 호출 역할을 제공합니다. 자세한 내용은 [Amazon API Gateway를 ID 제공자 통합에 사용](#)을 참조하세요.

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

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

4. 다음을 선택합니다.

5. 엔드포인트 선택에서 다음을 수행합니다.

- a. 엔드포인트 타입으로는 공적으로 액세스 가능 엔드포인트 타입을 선택합니다. VPC 호스팅 엔드포인트에 대한 내용은 [Virtual Private Cloud\(VPC\)에 서버 생성](#) 섹션을 참조하세요.
- b. IP 주소 유형에서 이전 버전과의 호환성을 위해 IPv4(기본값)를 선택하거나 엔드포인트에 대한 IPv4 및 IPv6 연결을 모두 활성화하려면 듀얼 스택을 선택합니다.

Note

듀얼 스택 모드를 사용하면 Transfer Family 엔드포인트가 IPv4 및 IPv6 지원 클라이언트와 모두 통신할 수 있습니다. 따라서 한 번에 모두 전환할 필요 없이 IPv4 기반 시스템에서 IPv6 기반 시스템으로 점진적으로 전환할 수 있습니다.

- c. (옵션) 맞춤 호스트 이름의 경우 없음을 선택합니다.

에서 제공하는 서버 호스트 이름을 가져옵니다 AWS Transfer Family. 서버 호스트 이름은 `serverId.server.transfer.regionId.amazonaws.com` 형식을 취합니다.

맞춤 호스트 이름의 경우 서버 엔드포인트의 맞춤 별칭을 지정합니다. 사용자 지정 호스트 이름 사용에 대한 자세한 내용은 [사용자 지정 호스트 이름으로 작업](#)를 참조하세요.

- d. (옵션) FIPS 지원의 경우 FIPS 지원 엔드포인트 확인란을 선택하여 엔드포인트가 미연방 정보 처리 표준(FIPS)을 준수하는지 확인합니다.

Note

FIPS 지원 엔드포인트는 북미 AWS 리전에서만 사용할 수 있습니다. 사용할 수 있는 리전은 AWS 일반 참조의 [AWS Transfer Family 엔드포인트 및 할당량](#)을 참조하세요. FIPS에 대한 자세한 설명은 [미연방 정보 처리 표준\(FIPS\) 140-2](#)를 참조하세요.

- e. 다음을 선택합니다.

6. 도메인 선택 페이지에서 선택한 프로토콜을 통해 데이터를 저장하고 액세스하는 데 사용할 AWS 스토리지 서비스를 선택합니다.

- 선택한 프로토콜을 통해 파일을 객체로 저장하고 액세스하려면 Amazon S3를 선택하세요.
- 선택한 프로토콜을 통해 Amazon EFS 파일 시스템에 파일을 저장하고 액세스하려면 Amazon EFS를 선택합니다.

다음을 선택합니다.

7. 추가 상세 정보 구성에서 다음을 수행합니다.

- a. 로깅의 경우 기존 로그 그룹을 지정하거나 새 로그 그룹을 생성합니다(기본 옵션). 기존 로그 그룹을 선택하는 경우와 연결된 로그 그룹을 선택해야 합니다 AWS 계정.

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/ [dropdown] [refresh] [Create 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.

로그 그룹 생성을 선택하면 CloudWatch 콘솔 (<https://console.aws.amazon.com/cloudwatch/>)에서 로그 그룹 생성 페이지가 열립니다. 자세한 내용은 [CloudWatch Logs의 로그 그룹 생성](#)을 참조하세요.

- b. (옵션) 관리형 워크플로의 경우 워크플로를 실행할 때 Transfer Family가 맡아야 하는 워크플로 ID(및 해당 역할)를 선택합니다. 업로드 완료 시 실행할 워크플로 하나와 부분 업로드 시 실행할 워크플로 하나를 선택할 수 있습니다. 관리형 워크플로를 사용하여 파일을 처리하는 방법에 대한 자세한 내용은 [AWS Transfer Family 관리형 워크플로](#)를 참조하세요.

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. 암호화 알고리즘 옵션의 경우 서버에서 사용하도록 설정된 암호화 알고리즘이 포함된 보안 정책을 선택합니다. 최신 보안 정책은 기본값입니다. 자세한 내용은 [AWS Transfer Family 서버에 대한 보안 정책](#)을 참조하세요.

- d. (옵션) 서버 호스트 키의 경우 클라이언트가 SFTP를 통해 서버에 연결할 때 서버를 식별하는데 사용할 RSA, ED25519 또는 ECDSA 프라이빗 키를 입력하세요. (옵션) 설명을 추가하여 여러 서버 호스트 키를 구분합니다.

서버를 생성한 후 호스트 키를 추가할 수 있습니다. 키를 교체하거나 RSA 키와 ECDSA 키 같은 다양한 타입의 키를 사용하려는 경우 호스트 키가 여러 개 있으면 유용합니다.

Note

서버 호스트 키 섹션은 기존 SFTP 지원 서버에서 사용자를 마이그레이션하는 데만 사용됩니다.

- e. (옵션) 태그의 키 및 값에서 하나 이상의 태그를 키-값 쌍을 입력한 다음 태그 추가를 선택합니다.
- f. 다음을 선택합니다.
- g. Amazon S3 디렉터리의 성능을 최적화할 수 있습니다. 예를 들어 홈 디렉터리로 이동하여 하위 디렉터리가 10,000개 있다고 가정해 보겠습니다. 즉, Amazon S3 버킷에는 10,000개의 폴더가 있습니다. 이 시나리오에서 `ls (list)` 명령을 실행하면 목록 작업이 6~8분 정도 걸립니다. 그러나 디렉터리를 최적화하는 경우이 작업은 몇 초 밖에 걸리지 않습니다.

콘솔을 사용하여 서버를 생성하면 최적화된 디렉터리가 기본적으로 활성화됩니다. API를 사용하여 서버를 생성하는 경우이 동작은 기본적으로 활성화되지 않습니다.

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

- h. (선택 사항) 조직 정책 또는 이용 약관과 같은 사용자 지정 메시지를 최종 사용자에게 표시하도록 AWS Transfer Family 서버를 구성합니다. 디스플레이 배너의 경우 사전 인증 디스플레이 배너 텍스트 상자에 사용자가 인증하기 전에 표시할 텍스트 메시지를 입력합니다.
- i. (옵션) 다음 추가 옵션을 구성할 수 있습니다.

- SetStat 옵션: 이 옵션을 사용하여 Amazon S3 버킷에 업로드하는 파일에 대해 클라이언트가 SETSTAT를 사용하려고 할 때 생성되는 오류를 무시합니다. 자세한 설명은 [ProtocolDetails](#) 주제의 SetStatOption 설명서를 참조하세요.
- TLS 세션 재개: 이 옵션은 FTPS를 이 서버의 프로토콜 중 하나로 활성화한 경우에만 사용할 수 있습니다.
- 패시브 IP: 이 옵션은 FTPS 또는 FTP를 이 서버의 프로토콜 중 하나로 활성화한 경우에만 사용할 수 있습니다.

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

i 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

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

8. 검토 및 생성에서 옵션을 검토합니다.

- 모든 항목을 편집하려면 단계 옆에 있는 편집을 선택합니다.

Note

편집하기로 선택한 단계 이후의 각 단계를 검토해야 합니다.

- 변경 사항이 없는 경우 서버 생성을 선택하여 서버를 생성하세요. 서버 페이지로 이동하고, 새 서버가 나열되는 다음 화면이 표시됩니다.

새 서버 상태가 온라인으로 변경되기까지 몇 분 정도 걸릴 수 있습니다. 이때 서버에서 파일 작업을 수행할 수 있지만 먼저 사용자를 만들어야 합니다. 사용자 생성에 대한 자세한 내용은 섹션을 참조하세요 [서버 엔드포인트에 대한 사용자 관리](#).

FTPS 지원 서버 생성

SSL을 통한 파일 전송 프로토콜(FTPS)은 FTP의 확장입니다. 전송 계층 보안(TLS) 및 Sockets Layer(SSL) 암호화 프로토콜을 사용하여 트래픽을 암호화합니다. FTPS를 사용하면 통제 및 데이터 채널 연결을 동시에 또는 개별적으로 암호화할 수 있습니다.

Note

Network Load Balancer에 대한 중요한 고려 사항은 섹션을 참조하세요 [NLBs 및 NATs 앞에 배치하지 마세요. AWS Transfer Family](#).

FTPS 지원 서버를 만들려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 열고 탐색 창에서 서버를 선택한 다음 서버 생성을 선택합니다.
2. 프로토콜 선택에서 FTPS를 선택합니다.

서버 인증서에서 클라이언트가 FTPS를 통해 연결할 때 서버를 식별하는 데 사용할 AWS Certificate Manager (ACM)에 저장된 인증서를 선택한 후 다음을 선택합니다.

새 퍼블릭 인증서를 요청하려면 AWS Certificate Manager 사용 설명서의 [퍼블릭 인증서 요청](#)을 참조하세요.

기존 인증서를 ACM으로 가져오려면 AWS Certificate Manager 사용 설명서의 [ACM으로 인증서 가져오기](#)를 참조하세요.

프라이빗 IP 주소를 통해 FTPS를 사용하도록 프라이빗 인증서를 요청하려면 AWS Certificate Manager 사용 설명서의 [프라이빗 인증서 요청](#)을 참조하세요.

다음 암호화 알고리즘 및 키 크기를 사용하는 인증서가 지원됩니다:

- 2048비트 RSA(RSA_2048)
- 4096비트 RSA(RSA_4096)
- 타원 프라임 곡선 256비트(EC_prime256v1)
- 타원 프라임 곡선 384 비트(EC_secp384r1)
- 타원 프라임 곡선 521비트(EC_secp521r1)

Note

인증서는 FQDN 또는 IP 주소가 지정된 유효한 SSL/TLS X.509 버전 3 인증서여야 하며 발급자에 대한 정보가 포함되어야 합니다.

3. 자격 증명 공급자 선택에서 사용자 액세스를 관리하는 데 사용할 자격 증명 공급자를 선택합니다. 다음과 같은 옵션이 있습니다:

- AWS Directory Service for Microsoft Active Directory - 엔드포인트에 액세스할 Directory Service 디렉터리를 제공합니다. 그러면 Active Directory에 저장된 자격 증명을 사용하여 사용자를 인증할 수 있습니다. 자격 AWS Managed Microsoft AD 증명 공급자 작업에 대한 자세한 내용은 섹션을 참조하세요 [Microsoft Active Directory용 AWS Directory Service 사용](#).

Note

- 교차 계정 및 공유 디렉터리는에서 지원되지 않습니다 AWS Managed Microsoft AD.
- Directory Service를 자격 증명 공급자로 사용하여 서버를 설정하려면 몇 가지 Directory Service 권한을 추가해야 합니다. 자세한 내용은 [사용을 시작하기 전에 AWS Directory Service for Microsoft Active Directory](#)을 참조하세요.

- 맞춤 ID 공급자 - 다음 옵션 중 하나를 선택합니다.
 - AWS Lambda 를 사용하여 자격 증명 공급자 연결 - Lambda 함수가 지원하는 기존 자격 증명 공급자를 사용할 수 있습니다. 귀하는 Lambda 함수의 명칭을 제공합니다. 자세한 내용은 [AWS Lambda 를 사용하여 자격 증명 공급자 통합](#)를 참조하세요.

- Amazon API Gateway를 사용하는 자격 증명 공급자 연결 - Lambda 함수가 지원하는 API Gateway 방법을 생성하여 자격 증명 공급자로 사용할 수 있습니다. 사용자는 Amazon API Gateway URL과 호출 역할을 제공합니다. 자세한 내용은 [Amazon API Gateway를 ID 제공자 통합에 사용](#)을 참조하세요.

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. 다음을 선택합니다.
5. 엔드포인트 선택에서 다음을 수행합니다.

Note

Transfer Family용 FTPS 서버는 포트 21(제어 채널) 및 포트 범위 8192-8200(데이터 채널)을 통해 작동합니다.

- a. 엔드포인트 타입에서 서버의 엔드포인트를 호스팅할 VPC 호스팅 엔드포인트 타입을 선택합니다. VPC 호스팅 엔드포인트를 설정하는 자세한 설명은 [Virtual Private Cloud\(VPC\)에 서버 생성](#) 섹션을 참조하세요.

Note

공적 액세스 가능 엔드포인트는 지원되지 않습니다.

- b. (옵션) FIPS 지원의 경우 FIPS 지원 엔드포인트 확인란을 선택하여 엔드포인트가 미연방 정보 처리 표준(FIPS)을 준수하는지 확인합니다.

Note

FIPS 지원 엔드포인트는 북미 AWS 리전에서만 사용할 수 있습니다. 사용할 수 있는 리전은 AWS 일반 참조의 [AWS Transfer Family 엔드포인트 및 할당량](#)을 참조하세요. FIPS에 대한 자세한 설명은 [미연방 정보 처리 표준\(FIPS\) 140-2](#)를 참조하세요.

- c. 다음을 선택합니다.

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

▼

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

☐ FIPS Enabled endpoint

6. 도메인 선택 페이지에서 선택한 프로토콜을 통해 데이터를 저장하고 액세스하는 데 사용할 AWS 스토리지 서비스를 선택합니다.

- 선택한 프로토콜을 통해 파일을 객체로 저장하고 액세스하려면 Amazon S3를 선택하세요.
- 선택한 프로토콜을 통해 Amazon EFS 파일 시스템에 파일을 저장하고 액세스하려면 Amazon EFS를 선택합니다.

다음을 선택합니다.

7. 추가 상세 정보 구성에서 다음을 수행합니다.

- a. 로깅의 경우 기존 로그 그룹을 지정하거나 새 로그 그룹을 생성합니다(기본 옵션).

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/ [dropdown] [refresh] [Create 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.

로그 그룹 생성을 선택하면 CloudWatch 콘솔 (<https://console.aws.amazon.com/cloudwatch/>)에서 로그 그룹 생성 페이지가 열립니다. 자세한 내용은 [CloudWatch Logs의 로그 그룹 생성](#)을 참조하세요.

- b. (옵션) 관리형 워크플로의 경우 워크플로를 실행할 때 Transfer Family가 맡아야 하는 워크플로 ID(및 해당 역할)를 선택합니다. 업로드 완료 시 실행할 워크플로 하나와 부분 업로드 시 실행할 워크플로 하나를 선택할 수 있습니다. 관리형 워크플로를 사용하여 파일을 처리하는 방법에 대한 자세한 내용은 [AWS Transfer Family 관리형 워크플로](#)를 참조하세요.

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]

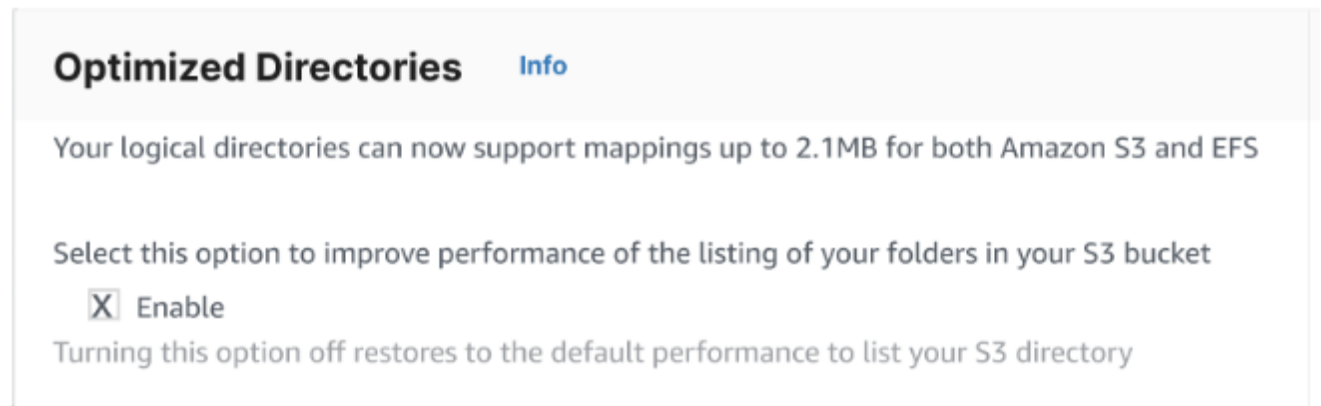
Managed workflows execution role Info
Select the role that AWS Transfer Family should assume when executing a workflow

[dropdown] [refresh]

- c. 암호화 알고리즘 옵션의 경우 서버에서 사용하도록 설정된 암호화 알고리즘이 포함된 보안 정책을 선택합니다. 최신 보안 정책은 기본값입니다. 자세한 내용은 [AWS Transfer Family 서버에 대한 보안 정책](#)을 참조하세요.

- d. 서버 호스트 키의 경우 비워 두세요.
- e. (옵션) 태그의 키 및 값에서 하나 이상의 태그를 키-값 쌍을 입력한 다음 태그 추가를 선택합니다.
- f. Amazon S3 디렉터리의 성능을 최적화할 수 있습니다. 예를 들어 홈 디렉터리로 이동하여 하위 디렉터리가 10,000개 있다고 가정해 보겠습니다. 즉, Amazon S3 버킷에는 10,000개의 폴더가 있습니다. 이 시나리오에서 `ls (list)` 명령을 실행하면 목록 작업이 6~8분 정도 걸립니다. 그러나 디렉터리를 최적화하는 경우이 작업은 몇 초 밖에 걸리지 않습니다.

콘솔을 사용하여 서버를 생성하면 최적화된 디렉터리가 기본적으로 활성화됩니다. API를 사용하여 서버를 생성하는 경우이 동작은 기본적으로 활성화되지 않습니다.



- g. 다음을 선택합니다.
- h. (선택 사항) 조직 정책 또는 이용 약관과 같은 사용자 지정 메시지를 최종 사용자에게 표시하도록 AWS Transfer Family 서버를 구성할 수 있습니다. 인증을 성공적으로 완료한 사용자에게 맞춤형 오늘의 메시지(MOTD)를 표시할 수도 있습니다.

디스플레이 배너의 경우 사전 인증 디스플레이 배너 텍스트 상자에 사용자가 인증하기 전에 표시할 텍스트 메시지를 입력하고, 인증 후 디스플레이 배너 텍스트 상자에는 인증에 성공한 후 사용자에게 표시할 텍스트를 입력합니다.

- i. (옵션) 다음 추가 옵션을 구성할 수 있습니다.
 - SetStat 옵션: 이 옵션을 사용하여 Amazon S3 버킷에 업로드하는 파일에 대해 클라이언트가 SETSTAT을 사용하려고 할 때 생성되는 오류를 무시합니다. 자세한 설명은 [ProtocolDetails](#) 주제의 SetStatOption 설명서를 참조하세요.
 - TLS 세션 재개: FTPS 세션에 대한 통제 및 데이터 연결 간에 협상된 암호 키를 재개하거나 공유하는 메커니즘을 제공합니다. 자세한 설명은 [ProtocolDetails](#) 주제의 TlsSessionResumptionMode 설명서를 참조하세요.

- 수동 IP: FTP 및 FTPS 프로토콜에 대한 수동 모드를 나타냅니다. 방화벽, 라우터 또는 로드 밸런서의 퍼블릭 IP 주소와 같은 단일 IPv4 주소를 입력합니다. 자세한 설명은 [ProtocolDetails](#) 주제의 PassiveIp 설명서를 참조하세요.

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. 검토 및 생성에서 옵션을 검토합니다.

- 모든 항목을 편집하려면 단계 옆에 있는 편집을 선택합니다.

Note

편집하기로 선택한 단계 이후의 각 단계를 검토해야 합니다.

- 변경 사항이 없는 경우 서버 생성을 선택하여 서버를 생성하세요. 서버 페이지로 이동하고, 새 서버가 나열되는 다음 화면이 표시됩니다.

새 서버 상태가 온라인으로 변경되기까지 몇 분 정도 걸릴 수 있습니다. 이때부터 서버는 사용자의 파일 작업을 수행할 수 있습니다.

다음 단계: 다음 단계에서는 [사용자 지정 자격 증명 공급자와 작업](#)를 계속 진행하여 사용자를 설정하세요.

FTP 지원 서버 생성

파일 전송 프로토콜(FTP)은 데이터 전송에 사용되는 네트워크 프로토콜입니다. FTP는 통제 및 데이터 전송에 별도의 채널을 사용합니다. 통제 채널은 종료되거나 비활성 제한 시간이 초과될 때까지 열려 있습니다. 데이터 채널은 전송 기간 동안 활성 상태입니다. FTP는 일반 텍스트를 사용하며 트래픽 암호화를 지원하지 않습니다.

Note

FTP를 활성화할 때는 VPC 호스팅 엔드포인트에 대한 내부 액세스 옵션을 선택해야 합니다. 서버가 퍼블릭 네트워크를 통과하는 데이터가 필요한 경우 SFTP 또는 FTPS와 같은 보안 프로토콜을 사용해야 합니다.

Note

Network Load Balancer에 대한 중요한 고려 사항은 섹션을 참조하세요 [NLBs 및 NATs 앞에 배치하지 마세요](#). [AWS Transfer Family](#).

FTP 지원 서버를 만들려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 열고 탐색 창에서 서버를 선택한 다음 서버 생성을 선택합니다.
2. 프로토콜 선택에서 FTP를 선택하고 다음을 선택합니다.
3. 자격 증명 공급자 선택에서 사용자 액세스를 관리하는 데 사용할 자격 증명 공급자를 선택합니다. 다음과 같은 옵션이 있습니다:
 - AWS Directory Service for Microsoft Active Directory - 엔드포인트에 액세스할 Directory Service 디렉터리를 제공합니다. 그러면 Active Directory에 저장된 자격 증명을 사용하여 사용자를 인증할 수 있습니다. 자격 AWS Managed Microsoft AD 증명 공급자 작업에 대한 자세한 내용은 섹션을 참조하세요 [Microsoft Active Directory용 AWS Directory Service 사용](#).

 Note

- 교차 계정 및 공유 디렉터리는에서 지원되지 않습니다 AWS Managed Microsoft AD.
- Directory Service를 자격 증명 공급자로 사용하여 서버를 설정하려면 몇 가지 Directory Service 권한을 추가해야 합니다. 자세한 내용은 [사용을 시작하기 전에 AWS Directory Service for Microsoft Active Directory](#)을 참조하세요.

- 맞춤 ID 공급자 – 다음 옵션 중 하나를 선택합니다.
- AWS Lambda 를 사용하여 자격 증명 공급자 연결 - Lambda 함수가 지원하는 기존 자격 증명 공급자를 사용할 수 있습니다. 귀하는 Lambda 함수의 명칭을 제공합니다. 자세한 내용은 [AWS Lambda 를 사용하여 자격 증명 공급자 통합](#)를 참조하세요.
- Amazon API Gateway를 사용하는 자격 증명 공급자 연결 - Lambda 함수가 지원하는 API Gateway 방법을 생성하여 자격 증명 공급자로 사용할 수 있습니다. 사용자는 Amazon API Gateway URL과 호출 역할을 제공합니다. 자세한 내용은 [Amazon API Gateway를 ID 제공자 통합에 사용](#)을 참조하세요.

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. 다음을 선택합니다.
5. 엔드포인트 선택에서 다음을 수행합니다.

[i](#) Note

Transfer Family용 FTP 및 FTPS 서버는 포트 21(제어 채널) 및 포트 범위 8192-8200(데이터 채널)을 통해 작동합니다.

- a. 엔드포인트 타입에서 서버의 엔드포인트를 호스팅할 VPC 호스팅을 선택합니다. VPC 호스팅 엔드포인트를 설정하는 자세한 설명은 [Virtual Private Cloud\(VPC\)에 서버 생성](#) 섹션을 참조하세요.

 Note

공적 액세스 가능 엔드포인트는 지원되지 않습니다.

- b. FIPS 지원의 경우 FIPS 지원 엔드포인트 확인란의 선택을 취소한 상태로 유지합니다.

 Note

FIPS 지원 엔드포인트는 FTP 서버에서 지원되지 않습니다.

- c. 다음을 선택합니다.

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

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

☐ FIPS Enabled endpoint

6. 도메인 선택 페이지에서 선택한 프로토콜을 통해 데이터를 저장하고 액세스하는 데 사용할 AWS 스토리지 서비스를 선택합니다.

- 선택한 프로토콜을 통해 파일을 객체로 저장하고 액세스하려면 Amazon S3를 선택하세요.
- 선택한 프로토콜을 통해 Amazon EFS 파일 시스템에 파일을 저장하고 액세스하려면 Amazon EFS를 선택합니다.

다음을 선택합니다.

7. 추가 상세 정보 구성에서 다음을 수행합니다.

- a. 로깅의 경우 기존 로그 그룹을 지정하거나 새 로그 그룹을 생성합니다(기본 옵션).

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/ [dropdown] [refresh] [Create 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.

로그 그룹 생성을 선택하면 CloudWatch 콘솔 (<https://console.aws.amazon.com/cloudwatch/>)에서 로그 그룹 생성 페이지가 열립니다. 자세한 내용은 [CloudWatch Logs의 로그 그룹 생성](#)을 참조하세요.

- b. (옵션) 관리형 워크플로의 경우 워크플로를 실행할 때 Transfer Family가 맡아야 하는 워크플로 ID(및 해당 역할)를 선택합니다. 업로드 완료 시 실행할 워크플로 하나와 부분 업로드 시 실행할 워크플로 하나를 선택할 수 있습니다. 관리형 워크플로를 사용하여 파일을 처리하는 방법에 대한 자세한 내용은 [AWS Transfer Family 관리형 워크플로](#)를 참조하세요.

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. 암호화 알고리즘 옵션의 경우 서버에서 사용하도록 설정된 암호화 알고리즘이 포함된 보안 정책을 선택합니다.

Note

Transfer Family는 FTP 서버에 최신 보안 정책을 할당합니다. 그러나 FTP 프로토콜은 암호화를 사용하지 않으므로 FTP 서버는 보안 정책 알고리즘을 사용하지 않습니다. 서버에서 FTPS 또는 SFTP 프로토콜도 사용하지 않는 한 보안 정책은 사용되지 않습니다.

- d. 서버 호스트 키의 경우 비워 두세요.
- e. (옵션) 태그의 키 및 값에서 하나 이상의 태그를 키-값 쌍을 입력한 다음 태그 추가를 선택합니다.
- f. Amazon S3 디렉터리의 성능을 최적화할 수 있습니다. 예를 들어 홈 디렉터리로 이동하여 하위 디렉터리가 10,000개 있다고 가정해 보겠습니다. 즉, Amazon S3 버킷에는 10,000개의 폴더가 있습니다. 이 시나리오에서 `ls` (list) 명령을 실행하면 목록 작업이 6~8분 정도 걸립니다. 그러나 디렉터리를 최적화하는 경우이 작업은 몇 초 밖에 걸리지 않습니다.

콘솔을 사용하여 서버를 생성하면 최적화된 디렉터리가 기본적으로 활성화됩니다. API를 사용하여 서버를 생성하는 경우이 동작은 기본적으로 활성화되지 않습니다.

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. 다음을 선택합니다.
- h. (선택 사항) 조직 정책 또는 이용 약관과 같은 사용자 지정 메시지를 최종 사용자에게 표시하도록 AWS Transfer Family 서버를 구성할 수 있습니다. 인증을 성공적으로 완료한 사용자에게 맞춤형 오늘의 메시지(MOTD)를 표시할 수도 있습니다.

디스플레이 배너의 경우 사전 인증 디스플레이 배너 텍스트 상자에 사용자가 인증하기 전에 표시할 텍스트 메시지를 입력하고, 인증 후 디스플레이 배너 텍스트 상자에는 인증에 성공한 후 사용자에게 표시할 텍스트를 입력합니다.

- i. (옵션) 다음 추가 옵션을 구성할 수 있습니다.

- SetStat 옵션: 이 옵션을 사용하여 Amazon S3 버킷에 업로드하는 파일에 대해 클라이언트가 SETSTAT를 사용하려고 할 때 생성되는 오류를 무시합니다. 자세한 설명은 [ProtocolDetails](#) 주제의 SetStatOption 설명서를 참조하세요.
- TLS 세션 재개: FTPS 세션에 대한 통제 및 데이터 연결 간에 협상된 암호 키를 재개하거나 공유하는 메커니즘을 제공합니다. 자세한 설명은 [ProtocolDetails](#) 주제의 TlsSessionResumptionMode 설명서를 참조하세요.
- 수동 IP: FTP 및 FTPS 프로토콜에 대한 수동 모드를 나타냅니다. 방화벽, 라우터 또는 로드 밸런서의 퍼블릭 IP 주소와 같은 단일 IPv4 주소를 입력합니다. 자세한 설명은 [ProtocolDetails](#) 주제의 PassiveIp 설명서를 참조하세요.

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. 검토 및 생성에서 옵션을 검토합니다.

- 모든 항목을 편집하려면 단계 옆에 있는 편집을 선택합니다.

Note

편집하기로 선택한 단계 이후의 각 단계를 검토해야 합니다.

- 변경 사항이 없는 경우 서버 생성을 선택하여 서버를 생성하세요. 서버 페이지로 이동하고, 새 서버가 나열되는 다음 화면이 표시됩니다.

새 서버 상태가 온라인으로 변경되기까지 몇 분 정도 걸릴 수 있습니다. 이때부터 서버는 사용자의 파일 작업을 수행할 수 있습니다.

다음 단계 - 그 다음 단계에서는 [사용자 지정 자격 증명 공급자와 작업](#)로 계속 진행하여 사용자를 설정하세요.

Virtual Private Cloud(VPC)에 서버 생성

가상 사설 클라우드(VPC) 내에 서버의 엔드포인트를 호스팅하여 퍼블릭 인터넷을 거치지 않고 Amazon S3 버킷 또는 Amazon EFS 파일 시스템과 데이터를 주고 받는 데 사용할 수 있습니다.

Note

2021년 5월 19일 이후에는 2021년 5월 19일 이전에 AWS 계정이 아직 생성하지 않은 경우 계정 EndpointType=VPC_ENDPOINT에서를 사용하여 서버를 생성할 수 없습니다. 2021년 2월 21일 또는 그 이전에 EndpointType=VPC_ENDPOINT AWS 계정에 있는 서버를 이미 생성한 경우 영향을 받지 않습니다. 이 날짜 이후에는 EndpointType=**VPC**를 사용하세요. 자세한 내용은 [the section called “VPC_ENDPOINT 사용 중단”](#) 단원을 참조하십시오.

Amazon Virtual Private Cloud(VPC)를 사용하여 AWS 리소스를 호스팅하는 경우 VPC와 서버 간에 프라이빗 연결을 설정할 수 있습니다. 그러면 퍼블릭 IP 주소 지정을 사용하거나 인터넷 게이트웨이 없이 이 서버를 사용하여 클라이언트를 통해 Amazon S3 버킷과 데이터를 주고 받을 수 있습니다.

Amazon VPC를 사용하면 사용자 지정 가상 네트워크에서 AWS 리소스를 시작할 수 있습니다. VPC를 사용하여 IP 주소 범위, 서브넷, 라우팅 테이블, 네트워크 게이트웨이 등의 네트워크 설정을 제어할 수 있습니다. VPC에 대한 자세한 내용은 Amazon VPC 사용 설명서의 [Amazon VPC란 무엇인가요?](#)를 참조하세요.

다음 섹션에는 VPC를 생성하고 서버에 연결하는 방법에 대한 지침이 나와 있습니다. 개요로 다음과 같이 이 작업을 수행합니다.

1. VPC 엔드포인트를 사용하여 서버를 설정합니다.
2. 그런 다음 VPC 엔드포인트를 통해 VPC 내부에 있는 클라이언트를 사용하여 서버에 연결합니다. 이렇게 하면 AWS Transfer Family를 사용하여 Amazon S3 버킷에 저장된 데이터를 클라이언트를 통

해 전송할 수 있습니다. 네트워크와 퍼블릭 인터넷 연결이 끊긴 경우에도 이 전송을 수행할 수 있습니다.

3. 또한 서버의 엔드포인트를 인터넷에 연결하도록 선택한 경우 탄력적 IP 주소를 엔드포인트와 연결할 수 있습니다. 이렇게 하면 VPC 외부의 클라이언트가 서버에 연결할 수 있습니다. VPC 보안 그룹을 사용하여 허용된 주소에서만 요청을 보내는 인증된 사용자에 대한 액세스를 제어할 수 있습니다.

Note

AWS Transfer Family 는 듀얼 스택 엔드포인트를 지원하므로 서버가 IPv4와 IPv6를 통해 통신할 수 있습니다. 듀얼 스택 지원을 활성화하려면 VPC 엔드포인트를 생성할 때 DNS 듀얼 스택 엔드포인트 활성화 옵션을 선택합니다. 이 기능을 사용하려면 먼저 IPv6를 지원하도록 VPC와 서브넷을 모두 구성해야 합니다. 듀얼 스택 지원은 두 프로토콜 중 하나를 사용하여 연결해야 하는 클라이언트가 있는 경우에 특히 유용합니다.

듀얼 스택(IPv4 및 IPv6) 서버 엔드포인트에 대한 자세한 내용은 섹션을 참조하세요 [Transfer Family 서버에 대한 IPv6 지원](#).

주제

- [VPC 내에서만 액세스할 수 있는 서버 엔드포인트 생성](#)
- [서버용 인터넷 경계 엔드포인트 생성](#)
- [서버의 엔드포인트 타입 변경](#)
- [VPC_ENDPOINT 사용 중단](#)
- [Transfer Family 서버에 대한 VPC 엔드포인트 액세스 제한](#)
- [추가 네트워킹 기능](#)
- [AWS Transfer Family 서버 엔드포인트 유형을 VPC_ENDPOINT에서 VPC로 업데이트](#)

VPC 내에서만 액세스할 수 있는 서버 엔드포인트 생성

다음 절차에서는 VPC 내 리소스에만 액세스할 수 있는 서버 엔드포인트를 생성합니다.

VPC 내부에 서버 엔드포인트를 생성하려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 탐색 창에서 서버를 선택한 다음 서버 생성을 선택합니다.

3. 프로토콜 선택에서 하나 이상의 프로토콜을 선택한 후 다음을 선택합니다. 프로토콜에 대한 자세한 내용은 [2단계: SFTP 지원 서버 생성](#)를 참조하세요.
4. 자격 증명 공급자 선택에서 사용자 자격 증명과 키를 저장할 서비스 관리형을 선택한 후 다음을 AWS Transfer Family 선택합니다.

이 절차에서는 서비스 관리형 옵션을 사용합니다. 사용자 지정을 선택하는 경우, Amazon API Gateway 엔드포인트와 AWS Identity and Access Management (IAM) 역할을 입력하여 엔드포인트에 액세스해야 합니다. 이렇게 하면 디렉터리 서비스를 통합해 사용자를 인증하고 승인할 수 있습니다. 사용자 지정 자격 증명 공급자와의 작업에 대한 자세한 내용은 [사용자 지정 자격 증명 공급자와 작업](#)를 참조하세요.

5. 엔드포인트 선택에서 다음을 수행합니다.
 - a. 엔드포인트 타입에서 서버의 엔드포인트를 호스팅할 VPC 호스팅 엔드포인트 타입을 선택합니다.
 - b. 액세스에서 내부를 선택하여 엔드포인트의 사설 IP 주소를 사용하는 클라이언트만 엔드포인트에 액세스할 수 있도록 합니다.

인터넷 연결 옵션에 대한 자세한 내용은 [서버용 인터넷 경계 엔드포인트 생성](#)를 참조하세요. 내부 액세스 전용으로 VPC에 생성된 서버는 사용자 지정 호스트 이름을 지원하지 않습니다.

- c. VPC의 경우 기존 VPC ID를 선택하거나 VPC 생성을 선택하여 새 VPC를 생성합니다.
- d. 가용 영역 섹션에서 가용 영역 및 관련 서브넷을 최대 3개까지 선택합니다.
- e. 보안 그룹 섹션에서 기존 보안 그룹 ID 또는 ID를 선택하거나 보안 그룹 생성을 선택하여 새 보안 그룹을 생성합니다. 보안 그룹에 관한 자세한 내용은 Amazon Virtual Private Cloud 사용 설명서의 [VPC의 보안 그룹](#)을 참조하세요. 보안 그룹을 생성하려면 Amazon Virtual Private Cloud 사용 설명서의 [보안 그룹 생성](#)을 참조하세요.

Note

VPC는 자동으로 기본 보안 그룹과 함께 제공됩니다. 서버를 시작할 때 다른 보안 그룹 또는 그룹을 지정하지 않을 경우 기본 보안 그룹이 서버에 연결됩니다.

- 보안 그룹에 대한 인바운드 규칙의 경우 포트 22, 2222, 22000 또는 모든 조합을 사용하도록 SSH 트래픽을 구성할 수 있습니다. 포트 22는 기본적으로 구성됩니다. 포트 2222 또는 포트 22000을 사용하려면 보안 그룹에 인바운드 규칙을 추가합니다. 유형에 사용자 지정 TCP를 선택한 다음 포트 범위에 **2222** 또는 **22000**를 입력하고 소스에 SSH 포트 22 규칙에 대해 보유한 것과 동일한 CIDR 범위를 입력합니다.

- 보안 그룹에 대한 인바운드 규칙의 경우 제어 채널 및 **8192-8200** 데이터 채널²¹에 포트 범위를 사용하도록 FTP 및 FTPS 트래픽을 구성합니다.

 Note

TCP "piggy-back" ACKs 필요한 클라이언트에 포트 2223을 사용하거나 TCP 3방향 핸드셰이크의 최종 ack에도 데이터를 포함할 수 있습니다.

일부 클라이언트 소프트웨어는 포트 2223과 호환되지 않을 수 있습니다. 예를 들어 클라이언트가 SFTP 식별 문자열을 전송하기 전에 서버가 필요한 클라이언트입니다.

VPC > Security Groups > sg-[redacted]-default > Edit inbound rules

Edit inbound rules [Info](#)

Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules [Info](#)

Security group rule ID	Type Info
sg-[redacted]	HTTP
sg-[redacted]	RDP
sg-[redacted]	HTTPS
sg-[redacted]	Custom TCP
sg-[redacted]	SSH

[Add rule](#)

- f. (옵션) FIPS 지원의 경우 FIPS 지원 엔드포인트 확인란을 선택하여 엔드포인트가 Federal Information Processing Standard(FIPS)를 준수하는지 확인합니다.

Note

FIPS 지원 엔드포인트는 북미 AWS 리전에서만 사용할 수 있습니다. 사용할 수 있는 리전은 AWS 일반 참조의 [AWS Transfer Family 엔드포인트 및 할당량](#)을 참조하세요. FIPS에 대한 자세한 설명은 [미연방 정보 처리 표준\(FIPS\) 140-2](#)를 참조하세요.

- g. 다음을 선택합니다.
6. 추가 상세 정보 구성에서 다음을 수행합니다.
 - a. CloudWatch 로깅의 경우 다음 중 하나를 선택하여 사용자 활동에 대한 Amazon CloudWatch 로깅을 활성화합니다.
 - 새 역할을 생성할 수 있는 적절한 권한이 있는 한 Transfer Family에서 자동으로 IAM 역할을 생성할 수 있도록 새 역할을 생성합니다. 생성된 IAM 역할은 `AWSTransferLoggingAccess(이)`라고 합니다.
 - 계정에서 기존 IAM 역할을 선택하려면 기존 역할을 선택합니다. 로깅 역할에서 역할을 선택합니다. 이 IAM 역할에는 서비스가 `transfer.amazonaws.com(으)`로 설정된 신뢰 정책이 포함되어야 합니다.

CloudWatch 로깅에 대한 자세한 내용은 [CloudWatch 로깅 역할 구성](#)를 참조하세요.

Note

- 로깅 역할을 지정하지 않으면 CloudWatch에서 최종 사용자 활동을 볼 수 없습니다.
- CloudWatch 로깅 역할을 설정하지 않으려면 기존 역할 선택을 선택하고 로깅 역할은 선택하지 마세요.

- b. 암호화 알고리즘 옵션의 경우 서버에서 사용하도록 설정된 암호화 알고리즘이 포함된 보안 정책을 선택합니다.

Note

다른 정책을 선택하지 않는 한 `TransferSecurityPolicy-2024-01` 보안 정책이 기본적으로 서버에 연결됩니다.

보안 정책에 대한 자세한 설명은 섹션을 참조하세요 [AWS Transfer Family 서버에 대한 보안 정책](#).

- c. (선택 사항: 이 섹션은 기존 SFTP 지원 서버에서 사용자를 마이그레이션하는 경우에만 해당됩니다.) 서버 호스트 키에 클라이언트가 SFTP를 통해 연결할 때 서버를 식별하는 데 사용할 RSA, ED25519 또는 ECDSA 프라이빗 키를 입력합니다.
 - d. (옵션) 태그의 키 및 값에서 하나 이상의 태그를 키-값 쌍을 입력한 다음 태그 추가를 선택합니다.
 - e. 다음을 선택합니다.
7. 검토 및 생성에서 옵션을 검토합니다. 경우에 따라 다음 작업을 수행합니다.
- 이들 중 하나를 편집하려면 단계 옆에 있는 편집을 선택합니다.

Note

편집하기로 선택한 단계 이후의 각 단계를 검토해야 합니다.

- 변경 사항이 없으면 서버 생성을 선택하여 서버를 생성합니다. 서버 페이지로 이동하고, 새 서버가 나열되는 다음 화면이 표시됩니다.

새 서버 상태가 온라인으로 변경되기까지 몇 분 정도 걸릴 수 있습니다. 이때 서버에서 파일 작업을 수행할 수 있지만 먼저 사용자를 만들어야 합니다. 사용자 생성에 대한 자세한 내용은 섹션을 참조하세요 [서버 엔드포인트에 대한 사용자 관리](#).

서버용 인터넷 경계 엔드포인트 생성

다음 절차에서는 서버 엔드포인트를 만듭니다. 이 엔드포인트는 VPC의 기본 보안 그룹에서 원본 IP 주소를 허용하는 클라이언트만 인터넷을 통해 액세스할 수 있습니다. 또한 탄력적 IP 주소를 사용하여 엔드포인트가 인터넷 경계가 되도록 하면 클라이언트는 탄력적 IP 주소를 사용하여 방화벽에서 엔드포인트에 대한 액세스를 허용할 수 있습니다.

Note

인터넷 경계 VPC 호스팅 엔드포인트에서는 SFTP와 FTPS만 사용할 수 있습니다.

인터넷 경계 엔드포인트를 생성하려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 탐색 창에서 서버를 선택한 다음 서버 생성을 선택합니다.
3. 프로토콜 선택에서 하나 이상의 프로토콜을 선택한 후 다음을 선택합니다. 프로토콜에 대한 자세한 내용은 [2단계: SFTP 지원 서버 생성](#)를 참조하세요.
4. 자격 증명 공급자 선택에서 사용자 자격 증명과 키를 저장할 서비스 관리형을 선택한 후 다음을 AWS Transfer Family 선택합니다.

이 절차에서는 서비스 관리형 옵션을 사용합니다. 사용자 지정을 선택하는 경우, Amazon API Gateway 엔드포인트와 AWS Identity and Access Management (IAM) 역할을 입력하여 엔드포인트에 액세스해야 합니다. 이렇게 하면 디렉터리 서비스를 통합해 사용자를 인증하고 승인할 수 있습니다. 사용자 지정 자격 증명 공급자와의 작업에 대한 자세한 내용은 [사용자 지정 자격 증명 공급자와 작업](#)를 참조하세요.

5. 엔드포인트 선택에서 다음을 수행합니다.
 - a. 엔드포인트 타입에서 서버의 엔드포인트를 호스팅할 VPC 호스팅 엔드포인트 타입을 선택합니다.
 - b. 액세스의 경우 인터넷 연결을 선택하여 클라이언트가 인터넷을 통해 엔드포인트에 액세스할 수 있게 합니다.

Note

인터넷 연결을 선택하면 각 서브넷 또는 서브넷에서 기존 탄력적 IP 주소를 선택할 수 있습니다. 또는 VPC 콘솔(<https://console.aws.amazon.com/vpc/>)으로 이동하여 하나 이상의 새 탄력적 IP 주소를 할당할 수 있습니다. 이러한 주소는 AWS 또는 사용자가 소유할 수 있습니다. 이미 사용 중인 탄력적 IP 주소는 엔드포인트와 연결할 수 없습니다.

- c. (옵션) 사용자 지정 호스트 이름의 경우 다음 중 하나를 선택합니다.

Note

의 고객은 탄력적 IP 주소를 통해 직접 연결하거나 상용 Route 53 내에서 EIP를 가리키는 호스트 이름 레코드를 생성 AWS GovCloud (US) 해야 합니다. GovCloud 엔드포인트용 Route 53 사용에 대한 자세한 내용은 AWS GovCloud (US) 사용 설명서의 [AWS GovCloud \(US\) 리소스로 Amazon Route 53 설정을 참조하세요](#).

- Amazon Route 53 DNS 별칭 – 사용하려는 호스트 이름이 Route 53에 등록된 경우. 이후 호스트 이름을 입력하면 됩니다.
- 기타 DNS – 사용할 호스트 이름이 다른 DNS 공급자를 통해 등록된 경우. 이후 호스트 이름을 입력하면 됩니다.
- 없음 – 서버의 엔드포인트를 사용하고 사용자 지정 호스트 이름은 사용하지 않을 경우. 서버 호스트 이름은 `server-id.server.transfer.region.amazonaws.com` 형식을 취합니다.

 Note

의 고객의 경우 없음을 AWS GovCloud (US)선택하면이 형식으로 호스트 이름이 생성되지 않습니다.

사용자 지정 호스트 이름 사용에 대한 자세한 내용은 [사용자 지정 호스트 이름으로 작업을 참조](#)하세요.

- VPC의 경우 기존 VPC ID를 선택하거나 VPC 생성을 선택하여 새 VPC를 생성합니다.
- 가용 영역 섹션에서 가용 영역 및 관련 서브넷을 최대 3개까지 선택합니다. IPv4 주소의 경우 각 서브넷의 탄력적 IP 주소를 선택합니다. 클라이언트가 방화벽에서 엔드포인트에 대한 액세스를 허용하는 데 사용할 수 있는 IP 주소입니다.

팁: 가용 영역에 퍼블릭 서브넷을 사용하거나 프라이빗 서브넷을 사용하려면 먼저 인터넷 게이트웨이를 설정해야 합니다.

- 보안 그룹 섹션에서 기존 보안 그룹 ID 또는 ID를 선택하거나 보안 그룹 생성을 선택하여 새 보안 그룹을 생성합니다. 보안 그룹에 관한 자세한 내용은 Amazon Virtual Private Cloud 사용 설명서의 [VPC의 보안 그룹](#)을 참조하세요. 보안 그룹을 생성하려면 Amazon Virtual Private Cloud 사용 설명서의 [보안 그룹 생성](#)을 참조하세요.

 Note

VPC는 자동으로 기본 보안 그룹과 함께 제공됩니다. 서버를 시작할 때 다른 보안 그룹 또는 그룹을 지정하지 않을 경우 기본 보안 그룹이 서버에 연결됩니다.

- 보안 그룹에 대한 인바운드 규칙의 경우 포트 22, 2222, 22000 또는 모든 조합을 사용하도록 SSH 트래픽을 구성할 수 있습니다. 포트 22는 기본적으로 구성됩니다. 포트 2222 또는 포트 22000을 사용하려면 보안 그룹에 인바운드 규칙을 추가합니다. 유형에 사용자 지정 TCP를 선택한 다음 포트 범위에 **2222** 또는 **22000**를 입력하고 소스에 SSH 포트 22 규칙에 대해 보유한 것과 동일한 CIDR 범위를 입력합니다.
- 보안 그룹에 대한 인바운드 규칙의 경우 제어 채널 및 **8192-8200** 데이터 채널에 포트 범위를 사용하도록 FTPS 트래픽을 구성합니다.

 Note

TCP "piggy-back" ACKs 필요한 클라이언트에 포트 2223을 사용하거나 TCP 3방향 핸드셰이크의 최종 ack에도 데이터를 포함할 수 있습니다.

일부 클라이언트 소프트웨어는 포트 2223과 호환되지 않을 수 있습니다. 예를 들어 클라이언트가 SFTP 식별 문자열을 전송하기 전에 서버가 필요한 클라이언트입니다.

VPC > Security Groups > sg-[redacted]-default > Edit inbound rules

Edit inbound rules [Info](#)

Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules [Info](#)

Security group rule ID	Type Info
sg-[redacted]	HTTP
sg-[redacted]	RDP
sg-[redacted]	HTTPS
sg-[redacted]	Custom TCP
sg-[redacted]	SSH

[Add rule](#)

- g. (옵션) FIPS 지원의 경우 FIPS 지원 엔드포인트 확인란을 선택하여 엔드포인트가 Federal Information Processing Standard(FIPS)를 준수하는지 확인합니다.

Note

FIPS 지원 엔드포인트는 북미 AWS 리전에서만 사용할 수 있습니다. 사용할 수 있는 리전은 AWS 일반 참조의 [AWS Transfer Family 엔드포인트 및 할당량](#)을 참조하세요. FIPS에 대한 자세한 설명은 [미연방 정보 처리 표준\(FIPS\) 140-2](#)를 참조하세요.

h. 다음을 선택합니다.

6. 추가 상세 정보 구성에서 다음을 수행합니다.

a. CloudWatch 로깅의 경우 다음 중 하나를 선택하여 사용자 활동에 대한 Amazon CloudWatch 로깅을 활성화합니다.

- 새 역할을 생성할 수 있는 적절한 권한이 있는 한 Transfer Family에서 자동으로 IAM 역할을 생성할 수 있도록 새 역할을 생성합니다. 생성된 IAM 역할은 `AWSTransferLoggingAccess(이)`라고 합니다.
- 계정에서 기존 IAM 역할을 선택하려면 기존 역할을 선택합니다. 로깅 역할에서 역할을 선택합니다. 이 IAM 역할에는 서비스가 `transfer.amazonaws.com(으)`로 설정된 신뢰 정책이 포함되어야 합니다.

CloudWatch 로깅에 대한 자세한 내용은 [CloudWatch 로깅 역할 구성](#)를 참조하세요.

Note

- 로깅 역할을 지정하지 않으면 CloudWatch에서 최종 사용자 활동을 볼 수 없습니다.
- CloudWatch 로깅 역할을 설정하지 않으려면 기존 역할 선택을 선택하고 로깅 역할은 선택하지 마세요.

b. 암호화 알고리즘 옵션의 경우 서버에서 사용하도록 설정된 암호화 알고리즘이 포함된 보안 정책을 선택합니다.

Note

다른 정책을 선택하지 않는 한 `TransferSecurityPolicy-2024-01` 보안 정책이 기본적으로 서버에 연결됩니다.

보안 정책에 대한 자세한 설명은 섹션을 참조하세요 [AWS Transfer Family 서버에 대한 보안 정책](#).

- c. (선택 사항: 이 섹션은 기존 SFTP 지원 서버에서 사용자를 마이그레이션하는 경우에만 해당됩니다.) 서버 호스트 키에 클라이언트가 SFTP를 통해 연결할 때 서버를 식별하는 데 사용할 RSA, ED25519 또는 ECDSA 프라이빗 키를 입력합니다.
- d. (옵션) 태그의 키 및 값에서 하나 이상의 태그를 키-값 쌍을 입력한 다음 태그 추가를 선택합니다.
- e. 다음을 선택합니다.
- f. (옵션) 관리형 워크플로의 경우 워크플로를 실행할 때 Transfer Family가 맡아야 하는 워크플로 ID(및 해당 역할)를 선택합니다. 업로드 완료 시 실행할 워크플로 하나와 부분 업로드 시 실행할 워크플로 하나를 선택할 수 있습니다. 관리형 워크플로를 사용하여 파일을 처리하는 방법에 대한 자세한 내용은 [AWS Transfer Family 관리형 워크플로](#)를 참조하세요.

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

▼

7. 검토 및 생성에서 옵션을 검토합니다. 경우에 따라 다음 작업을 수행합니다.

- 이들 중 하나를 편집하려면 단계 옆에 있는 편집을 선택합니다.

Note

편집하기로 선택한 단계 이후의 각 단계를 검토해야 합니다.

- 변경 사항이 없으면 서버 생성을 선택하여 서버를 생성합니다. 서버 페이지로 이동하고, 새 서버가 나열되는 다음 화면이 표시됩니다.

서버 ID를 선택하여 방금 생성한 서버의 세부 설정을 볼 수 있습니다. 퍼블릭 IPv4 주소 열이 채워지면 입력한 탄력적 IP 주소가 서버의 엔드포인트와 성공적으로 연결됩니다.

Note

VPC의 서버가 온라인 상태일 때는 [UpdateServer](#) API를 통해서만 서브넷을 수정할 수 있습니다. 서버 엔드포인트의 탄력적 IP 주소를 추가하거나 변경하려면 [서버를 중지](#)해야 합니다.

서버의 엔드포인트 타입 변경

인터넷을 통해 액세스할 수 있는 기존 서버가 있는 경우(즉, 퍼블릭 엔드포인트 타입이 있는 경우) 해당 엔드포인트를 VPC 엔드포인트로 변경할 수 있습니다.

Note

VPC_ENDPOINT로 표시된 VPC에 기존 서버가 있는 경우 새 VPC 엔드포인트 타입으로 수정하는 것이 좋습니다. 이 새 엔드포인트 타입을 사용하면 더 이상 NLB(Network Load Balancer)를 사용하여 탄력적 IP 주소를 서버의 엔드포인트와 연결할 필요가 없습니다. 또한 VPC 보안 그룹을 사용하여 서버 엔드포인트에 대한 액세스를 제한할 수 있습니다. 그러나 필요에 따라 VPC_ENDPOINT 엔드포인트 타입을 계속 사용할 수 있습니다.

다음 절차에서는 서버가 현재 퍼블릭 엔드포인트 타입 또는 이전 VPC_ENDPOINT 타입을 사용하는 것으로 가정합니다.

서버의 엔드포인트 타입을 변경하려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 탐색 창에서 서버를 선택합니다.
3. 엔드포인트 타입을 변경하려는 서버의 확인란을 선택합니다.

Important

엔드포인트를 변경하려면 먼저 서버를 중지해야 합니다.

4. 작업에서 중지를 선택합니다.
5. 표시되는 확인 대화 상자에서 중지를 선택하여 서버 중지를 확인합니다.

Note

다음 단계로 진행하기 전에 엔드포인트 세부 정보에서 서버 상태가 오프라인으로 변경될 때까지 기다리세요. 몇 분 정도 걸릴 수 있습니다. 상태 변경을 보려면 서버 페이지에서 새로 고침을 선택해야 할 수도 있습니다.

서버가 오프라인이 될 때까지는 어떤 편집도 할 수 없습니다.

6. 엔드포인트 세부 정보에서 편집을 선택합니다.
7. 엔드포인트 구성 편집에서 다음을 수행합니다.
 - a. 엔드포인트 타입 편집에서 VPC 호스팅을 선택합니다.
 - b. 액세스에서 다음 중 하나를 선택합니다.
 - 엔드포인트의 사설 IP 주소를 사용하는 클라이언트만 엔드포인트에 액세스할 수 있도록 하려면 내부로 설정합니다.
 - 클라이언트가 퍼블릭 인터넷을 통해 엔드포인트에 액세스할 수 있도록 하려면 인터넷 연결을 선택합니다.

Note

인터넷 연결을 선택하면 각 서브넷 또는 서브넷에서 기존 탄력적 IP 주소를 선택할 수 있습니다. 또는 VPC 콘솔(<https://console.aws.amazon.com/vpc/>)로 이동하여 하나 이상의 새 탄력적 IP 주소를 할당할 수 있습니다. 이러한 주소는 AWS 또는 사용자가 소유할 수 있습니다. 이미 사용 중인 탄력적 IP 주소는 엔드포인트와 연결할 수 없습니다.

- c. (인터넷 연결 액세스 전용 옵션) 사용자 지정 호스트 이름의 경우 다음 중 하나를 선택합니다.
 - Amazon Route 53 DNS 별칭 – 사용하려는 호스트 이름이 Route 53에 등록된 경우. 이후 호스트 이름을 입력하면 됩니다.
 - 기타 DNS – 사용할 호스트 이름이 다른 DNS 공급자를 통해 등록된 경우. 이후 호스트 이름을 입력하면 됩니다.
 - 없음 – 서버의 엔드포인트를 사용하고 사용자 지정 호스트 이름은 사용하지 않을 경우. 서버 호스트 이름은 `serverId.server.transfer.regionId.amazonaws.com` 형식을 취합니다.

사용자 지정 호스트 이름 사용에 대한 자세한 내용은 [사용자 지정 호스트 이름으로 작업을 참조](#)하세요.

- d. VPC의 경우 기존 VPC ID를 선택하거나 VPC 생성을 선택하여 새 VPC를 생성합니다.
- e. 가용 영역 섹션에서 가용 영역 및 관련 서브넷을 최대 3개까지 선택합니다. 인터넷 연결을 선택한 경우 각 서브넷의 탄력적 IP 주소도 선택합니다.

Note

최대 3개의 가용 영역을 원하지만 가용 영역이 충분하지 않은 경우 VPC 콘솔(<https://console.aws.amazon.com/vpc/>)에서 가용 영역을 생성합니다. 서브넷 또는 탄력적 IP 주소를 수정할 경우 서버를 업데이트하는 데 몇 분 정도 걸립니다. 서버 업데이트가 완료될 때까지는 변경 내용을 저장할 수 없습니다.

- f. 저장을 선택합니다.

- 8. 작업에서 시작을 선택하고 서버 상태가 온라인으로 변경될 때까지 기다립니다. 이 작업은 몇 분 정도 걸릴 수 있습니다.

Note

퍼블릭 엔드포인트 타입을 VPC 엔드포인트 타입으로 변경한 경우 서버의 엔드포인트 타입이 VPC로 변경되었음을 알 수 있습니다.

기본 보안 그룹이 엔드포인트에 연결되어 있습니다. 보안 그룹을 변경하거나 추가하려면 [보안 그룹 생성](#)을 참조하세요.

VPC_ENDPOINT 사용 중단

AWS Transfer Family 는 새 AWS 계정에 EndpointType=VPC_ENDPOINT 대해를 사용하여 서버를 생성하는 기능을 중단했습니다. 2021년 5월 19일부터 엔드포인트 유형이 인 AWS Transfer Family 서버를 소유하지 않은 AWS 계정VPC_ENDPOINT은를 사용하여 새 서버를 생성할 수 없습니다 EndpointType=VPC_ENDPOINT. VPC_ENDPOINT 엔드포인트 타입을 사용하는 서버를 이미 소유하고 있는 경우 가능한 한 빨리 EndpointType=VPC 사용을 시작하는 것이 좋습니다. 자세한 내용은 [AWS Transfer Family 서버 엔드포인트 유형을 VPC_ENDPOINT에서 VPC로 업데이트를 참조](#)하세요.

2020년 초에 새 VPC 엔드포인트 타입을 출시했습니다. 자세한 내용은 [SFTP에 대한AWS Transfer Family 가 VPC 보안 그룹 및 탄력적 IP 주소를 지원하는 경우](#)를 참조하세요. 이 새 엔드포인트는 기능이 더 풍부하고 비용 효율적이며 PrivateLink 요금이 부과되지 않습니다. 자세한 내용은 [AWS PrivateLink 요금](#)을 참조하세요.

이 엔드포인트 타입은 이전 엔드포인트 타입(VPC_ENDPOINT)과 기능적으로 동일합니다. 탄력적 IP 주소를 엔드포인트에 직접 연결하여 인터넷에 연결되도록 하고 소스 IP 필터링에 보안 그룹을 사용할 수 있습니다. 자세한 내용은 [IP 허용 목록을 사용하여 AWS Transfer Family for SFTP 서버 보호 블로그 게시물을 참조하세요](#).

공유 VPC 환경에서 이 엔드포인트를 호스팅할 수도 있습니다. 자세한 내용은 [AWS Transfer Family 에서 현재 공유 서비스 VPC 환경 지원](#)을 참조하세요.

SFTP 외에도 VPC EndpointType를 사용하여 FTPS 및 FTP를 활성화할 수 있습니다. 당사는 이러한 기능과 FTPS/FTP 지원을 EndpointType=VPC_ENDPOINT에 추가할 계획이 없습니다. 또한 이 엔드포인트 유형을 AWS Transfer Family 콘솔에서 옵션으로 제거했습니다.

Transfer Family 콘솔, API AWS CLI, SDKs 또는를 사용하여 서버의 엔드포인트 유형을 변경할 수 있습니다 CloudFormation. 서버의 엔드포인트 타입을 변경하려면 [AWS Transfer Family 서버 엔드포인트 유형을 VPC_ENDPOINT에서 VPC로 업데이트](#)를 참조하세요.

질문이 있는 경우 AWS Support 또는 AWS 계정 팀에 문의하세요.

Note

EndpointType=VPC_ENDPOINT에 이러한 기능과 FTPS 또는 FTP 지원을 추가할 계획이 없습니다. AWS Transfer Family 콘솔에서는 더 이상 옵션으로 제공하지 않습니다.

추가 질문이 있는 경우 AWS Support 또는 계정 팀을 통해 문의할 수 있습니다.

Transfer Family 서버에 대한 VPC 엔드포인트 액세스 제한

VPC 엔드포인트 유형으로 AWS Transfer Family 서버를 생성할 때 IAM 사용자와 보안 주체는 VPC 엔드포인트를 생성하고 삭제할 수 있는 권한이 필요합니다. 그러나 조직의 보안 정책은 이러한 권한을 제한할 수 있습니다. IAM 정책을 사용하여 다른 서비스에 대한 제한을 유지하면서 특히 Transfer Family에 대한 VPC 엔드포인트 생성 및 삭제를 허용할 수 있습니다.

⚠ Important

다음 IAM 정책은 사용자가 다른 서비스에 대해 이러한 작업을 거부하는 동안 Transfer Family 서버에 대해서만 VPC 엔드포인트를 생성하고 삭제할 수 있도록 허용합니다.

```
{
  "Effect": "Deny",
  "Action": [
    "ec2:CreateVpcEndpoint",
    "ec2:DeleteVpcEndpoints"
  ],
  "Resource": ["*"],
  "Condition": {
    "ForAnyValue:StringNotLike": {
      "ec2:VpceServiceName": [
        "com.amazonaws.INPUT-YOUR-REGION.transfer.server.*"
      ]
    },
    "StringNotLike": {
      "aws:PrincipalArn": [
        "arn:aws:iam::*:role/INPUT-YOUR-ROLE"
      ]
    }
  }
}
```

*INPUT-YOUR-REGION*을 AWS 리전(예: **us-east-1**)으로 바꾸고 *INPUT-YOUR-ROLE*을 이러한 권한을 부여하려는 IAM 역할로 바꿉니다.

추가 네트워킹 기능

AWS Transfer Family 는 VPC 구성을 사용할 때 보안과 유연성을 강화하는 몇 가지 고급 네트워킹 기능을 제공합니다.

- 공유 VPC 환경 지원 - 공유 VPC 환경에서 Transfer Family 서버 엔드포인트를 호스팅할 수 있습니다. 자세한 내용은 [공유 VPC에서 VPCs AWS Transfer Family](#).
- 인증 및 보안 - AWS 웹 애플리케이션 방화벽을 사용하여 Amazon API Gateway 엔드포인트를 보호할 수 있습니다. 자세한 내용은 [AWS 웹 애플리케이션 방화벽 및 Amazon API Gateway AWS Transfer Family 로 보안을 참조하세요](#).

AWS Transfer Family 서버 엔드포인트 유형을 VPC_ENDPOINT에서 VPC로 업데이트

AWS Management Console CloudFormation 또는 Transfer Family API를 사용하여 서버의 EndpointType에서 로 업데이트할 수 VPC_ENDPOINT 있습니다VPC. 이러한 각 방법을 사용하여 서버 엔드포인트 타입을 업데이트하는 자세한 절차와 예가 다음 섹션에 나와 있습니다. 여러 AWS 리전과 여러 AWS 계정에 서버가 있는 경우 다음 섹션에 제공된 예제 스크립트를 수정하여 업데이트해야 하는 VPC_ENDPOINT 유형을 사용하여 서버를 식별할 수 있습니다.

주제

- [VPC_ENDPOINT 엔드포인트 타입을 사용한 서버 식별](#)
- [를 사용하여 서버 엔드포인트 유형 업데이트 AWS Management Console](#)
- [를 사용하여 서버 엔드포인트 유형 업데이트 CloudFormation](#)
- [API를 사용하여 서버 엔드포인트 타입 업데이트](#)

VPC_ENDPOINT 엔드포인트 타입을 사용한 서버 식별

AWS Management Console을 사용하여 VPC_ENDPOINT를 사용하는 서버를 식별할 수 있습니다.

콘솔을 사용하여 **VPC_ENDPOINT** 엔드포인트 타입을 사용하는 서버를 식별하려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 탐색 창에서 서버를 선택하면 해당 리전의 사용자 계정에 있는 서버 목록이 표시됩니다.
3. 엔드포인트 타입별로 서버 목록을 정렬하면 VPC_ENDPOINT를 사용하는 모든 서버를 볼 수 있습니다.

여러 AWS 리전 및 계정VPC_ENDPOINT에서를 사용하여 서버를 식별하려면

여러 AWS 리전과 여러 AWS 계정에 서버가 있는 경우 다음 예제 스크립트를 수정하여 VPC_ENDPOINT 엔드포인트 유형을 사용하여 서버를 식별할 수 있습니다. 예제 스크립트는 Amazon EC2 [DescribeRegions](#) 및 Transfer Family [ListServers](#) API 작업을 사용합니다. AWS 계정이 많은 경우 자격 증명 공급자에 대한 세션 프로필을 사용하여 인증하면 읽기 전용 감사자 액세스 권한이 있는 IAM 역할을 사용하여 계정을 순회할 수 있습니다.

1. 다음은 간단한 예입니다.

```
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. 업데이트할 서버 목록을 확보한 후에는 다음 섹션에 설명된 방법 중 하나를 사용하여 EndpointType을 VPC(으)로 업데이트할 수 있습니다.

를 사용하여 서버 엔드포인트 유형 업데이트 AWS Management Console

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 탐색 창에서 서버를 선택합니다.
3. 엔드포인트 타입을 변경하려는 서버의 확인란을 선택합니다.

Important

엔드포인트를 변경하려면 먼저 서버를 중지해야 합니다.

4. 작업에서 중지를 선택합니다.
5. 표시되는 확인 대화 상자에서 중지를 선택하여 서버 중지를 확인합니다.

Note

다음 단계로 진행하기 전에 서버 상태가 오프라인으로 변경될 때까지 기다리세요. 몇 분 정도 걸릴 수 있습니다. 상태 변경을 보려면 서버 페이지에서 새로 고침을 선택해야 할 수도 있습니다.

6. 상태가 오프라인으로 변경되면 서버 세부 정보 페이지를 표시할 서버를 선택합니다.

7. 엔드포인트 세부 정보 섹션에서 편집을 선택합니다.
8. 엔드포인트 타입으로 VPC 호스팅을 선택합니다.
9. 저장을 선택합니다.
10. 작업에서 시작을 선택하고 서버 상태가 온라인으로 변경될 때까지 기다립니다. 이 작업은 몇 분 정도 걸릴 수 있습니다.

를 사용하여 서버 엔드포인트 유형 업데이트 CloudFormation

이 섹션에서는를 사용하여 서버의 CloudFormation 를 로 업데이트EndpointType하는 방법을 설명합니다VPC. 를 사용하여 배포한 Transfer Family 서버에 대해이 절차를 사용합니다 CloudFormation. 이 예에서 Transfer Family 서버를 배포하는 데 사용된 원본 CloudFormation 템플릿은 다음과 같습니다.

```

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

```

템플릿이 다음과 같이 변경되어 업데이트되었습니다.

- EndpointType이(가) VPC(으)로 변경되었습니다.
- AWS::EC2::VPCEndpoint 리소스가 제거됩니다.
- SecurityGroupId, SubnetIds, 및 VpcId(이)가 AWS::Transfer::Server 리소스의 EndpointDetails 섹션으로 이동되었습니다.
- EndpointDetails의 VpcEndpointId 속성이 삭제되었습니다.

업데이트된 템플릿입니다.

```

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

```


를 사용하여 배포된 Transfer Family 서버의 엔드포인트 유형을 업데이트하려면 CloudFormation

1. 다음 단계를 사용하여 업데이트하려는 서버를 중지합니다.
 - a. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
 - b. 탐색 창에서 서버를 선택합니다.
 - c. 엔드포인트 타입을 변경하려는 서버의 확인란을 선택합니다.

 Important

엔드포인트를 변경하려면 먼저 서버를 중지해야 합니다.

- d. 작업에서 중지를 선택합니다.
 - e. 표시되는 확인 대화 상자에서 중지를 선택하여 서버 중지를 확인합니다.

 Note

다음 단계로 진행하기 전에 서버 상태가 오프라인으로 변경될 때까지 기다리세요. 몇 분 정도 걸릴 수 있습니다. 상태 변경을 보려면 서버 페이지에서 새로 고침을 선택해야 할 수도 있습니다.

2. CloudFormation 스택 업데이트

- a. <https://console.aws.amazon.com/cloudformation> CloudFormation 콘솔을 엽니다.
 - b. Transfer Family 서버를 생성하는 데 사용할 스택을 선택합니다.
 - c. 업데이트를 선택합니다.
 - d. 현재 템플릿 교체를 선택합니다.
 - e. 새 템플릿을 업로드합니다. CloudFormation 변경 세트를 사용하면 템플릿 변경 사항을 구현하기 전에 템플릿 변경이 실행 중인 리소스에 미치는 영향에 대해 이해하는 데 도움이 됩니다. 이 예에서는 전송 서버 리소스가 수정되고 VPCendPoint 리소스가 제거됩니다. VPC 엔드포인트 타입 서버는 사용자를 대신하여 VPC 엔드포인트를 생성하여 원본 VPCEndpoint 리소스를 대체합니다.

새 템플릿을 업로드한 후 변경 세트는 다음과 유사합니다.

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. 스택을 업데이트합니다.
3. 스택 업데이트가 완료되면 Transfer Family 관리 콘솔 <https://console.aws.amazon.com/transfer/>로 이동합니다.
4. 서버를 시작합니다. 업데이트한 서버를 선택한 CloudFormation 다음 작업 메뉴에서 시작을 선택합니다.

API를 사용하여 서버 엔드포인트 타입 업데이트

[describe-server](#) AWS CLI 명령 또는 [UpdateServer](#) API 명령을 사용할 수 있습니다. 다음 예 스크립트는 Transfer Family 서버를 중지하고, 엔드포인트 타입을 업데이트하고, VPC_ENDPOINT를 제거하고, 서버를 시작합니다.

```
import boto3
import time

profile = input("Enter the name of the AWS account you'll be working in: ")
region_name = input("Enter the AWS Region you're working in: ")
server_id = input("Enter the AWS Transfer Server Id: ")

session = boto3.Session(profile_name=profile)

ec2 = session.client("ec2", region_name=region_name)
transfer = session.client("transfer", region_name=region_name)

group_ids=[]

transfer_description = transfer.describe_server(ServerId=server_id)
```

```

if transfer_description['Server']['EndpointType']=='VPC_ENDPOINT':
    transfer_vpc_endpoint = transfer_description['Server']['EndpointDetails']
['VpcEndpointId']
    transfer_vpc_endpoint_descriptions =
ec2.describe_vpc_endpoints(VpcEndpointIds=[transfer_vpc_endpoint])
    for transfer_vpc_endpoint_description in
transfer_vpc_endpoint_descriptions['VpcEndpoints']:
        subnet_ids=transfer_vpc_endpoint_description['SubnetIds']
        group_id_list=transfer_vpc_endpoint_description['Groups']
        vpc_id=transfer_vpc_endpoint_description['VpcId']
        for group_id in group_id_list:
            group_ids.append(group_id['GroupId'])
    if transfer_description['Server']['State']=='ONLINE':
        transfer_stop = transfer.stop_server(ServerId=server_id)
        print(transfer_stop)
        time.sleep(300) #safe
        transfer_update =
transfer.update_server(ServerId=server_id,EndpointType='VPC',EndpointDetails={'SecurityGroupId
        print(transfer_update)
        time.sleep(10)
        transfer_start = transfer.start_server(ServerId=server_id)
        print(transfer_start)
        delete_vpc_endpoint =
ec2.delete_vpc_endpoints(VpcEndpointIds=[transfer_vpc_endpoint])

```

사용자 지정 호스트 이름으로 작업

서버 호스트 이름은 사용자가 서버에 연결할 때 클라이언트에 입력하는 호스트 이름입니다. 작업할 때 서버 호스트 이름에 등록된 사용자 지정 도메인을 사용할 수 있습니다 AWS Transfer Family. 예를 들어 `mysftpserver.mysubdomain.domain.com` 같은 사용자 지정 호스트 이름을 사용할 수 있습니다.

등록한 사용자 지정 도메인에서 서버 엔드포인트로 트래픽을 리디렉션하려면, Amazon Route 53 또는 아무 DNS(Domain Name System) 공급자를 이용하면 됩니다. Route 53은 AWS Transfer Family (이)가 기본적으로 지원하는 DNS 서비스입니다.

주제

- [Amazon Route 53을 DNS 공급자로 사용](#)
- [다른 DNS 공급자 사용](#)
- [콘솔에서 생성하지 않은 서버의 사용자 지정 호스트 이름](#)

콘솔에서, 사용자 지정 호스트 이름을 설정하려면 다음 방법 중 하나를 선택합니다.

- Amazon Route 53 DNS 별칭 – 사용하려는 호스트 이름이 Route 53에 등록된 경우. 이후 호스트 이름을 입력하면 됩니다.
- 기타 DNS – 사용할 호스트 이름이 다른 DNS 공급자를 통해 등록된 경우. 이후 호스트 이름을 입력하면 됩니다.
- 없음 – 서버의 엔드포인트를 사용하고 사용자 지정 호스트 이름은 사용하지 않을 경우.

이 옵션은 새 서버를 만들거나 기존 서버의 구성을 수정할 때 설정합니다. 새 서버 생성에 대한 자세한 정보는 [2단계: SFTP 지원 서버 생성](#) 섹션을 참조하세요. 기존 서버 구성 편집에 대한 자세한 내용은 [서버 세부 정보 편집](#) 섹션을 참조하세요.

서버 호스트 이름에 자체 도메인을 사용하는 방법과가 Route 53을 AWS Transfer Family 사용하는 방법에 대한 자세한 내용은 다음 섹션을 참조하세요.

Amazon Route 53을 DNS 공급자로 사용

서버를 만들 때 DNS 공급자로 Amazon Route 53을 사용할 수 있습니다. Route 53을 도메인과 함께 사용하려면 먼저 도메인을 등록합니다. 자세한 내용은 Amazon Route 53 개발자 안내서의 [도메인 등록 방식](#)을 참조하세요.

Route 53을 사용하여 서버에 DNS 라우팅을 제공하는 경우는 입력한 사용자 지정 호스트 이름을 AWS Transfer Family 사용하여 호스팅 영역을 추출합니다. 가 호스팅 영역을 AWS Transfer Family 추출하면 다음과 같은 세 가지 상황이 발생할 수 있습니다.

1. Route 53을 처음 사용하고 호스팅 영역이 없는 경우 새 호스팅 영역과 CNAME 레코드를 AWS Transfer Family 추가합니다. 이 CNAME 레코드의 값은 서버의 엔드포인트 호스트 이름입니다. CNAME은 대체 도메인 이름입니다.
2. Route 53에 호스팅 영역이 있지만 CNAME 레코드가 없는 경우, AWS Transfer Family 에서 호스팅 영역에 CNAME 레코드를 추가합니다.
3. 서비스가 호스팅 영역에 존재하는 CNAME 레코드를 감지하면, CNAME 레코드가 이미 존재함을 알려주는 오류 메시지가 표시됩니다. 이 경우, CNAME 레코드 값을 서버의 호스트 이름으로 변경합니다.

Route 53의 호스팅 영역에 대한 자세한 내용은 Amazon Route 53 개발자 안내서의 [호스팅 영역](#)을 참조하세요.

다른 DNS 공급자 사용

서버를 생성할 때 Amazon Route 53 대신 DNS 공급자를 사용할 수도 있습니다. 대체 DNS 공급자를 사용하는 경우 도메인에서 나오는 트래픽이 서버 엔드포인트로 가는지 확인합니다.

이렇게 하려면 도메인을 서버의 엔드포인트 호스트 이름으로 설정합니다.

- IPv4 엔드포인트의 경우 호스트 이름은 콘솔에서 다음과 같습니다.

`serverid.server.transfer.region.amazonaws.com`

- 듀얼 스택 엔드포인트의 경우 콘솔에서 호스트 이름은 다음과 같습니다.

`serverid.transfer-server.region.on.aws`

Note

서버에 VPC 엔드포인트가 있는 경우 호스트 이름의 형식은 위에서 설명한 형식과 다릅니다. VPC 엔드포인트를 찾으려면 서버의 세부 정보 페이지에서 VPC를 선택한 다음 VPC 대시보드에서 VPC 엔드포인트 ID를 선택합니다. 엔드포인트는 나열된 DNS 이름 중 첫 번째 DNS 이름입니다.

콘솔에서 생성하지 않은 서버의 사용자 지정 호스트 이름

를 사용하거나 CLI를 AWS Cloud Development Kit (AWS CDK) CloudFormation 통해 서버를 생성할 때 해당 서버에 사용자 지정 호스트 이름을 지정하려면 태그를 추가해야 합니다. 콘솔을 사용하여 Transfer Family 서버를 생성하는 경우 태그가 자동으로 수행됩니다.

Note

또한 도메인에서 서버 엔드포인트로 트래픽을 리디렉션하려면 DNS 레코드를 생성해야 합니다. 자세한 내용은 Amazon Route 53 개발자 안내서의 [레코드 작업을](#) 참조하세요.

사용자 지정 호스트 이름에 사용할 키:

- 콘솔에 사용자 지정 호스트 이름을 표시하려면 `transfer:customHostname`를 추가합니다.
- Route 53을 DNS 공급자로 사용하고 있는 경우 `transfer:route53HostedZoneId`를 추가합니다. 이 태그는 사용자 지정 호스트 이름을 Route 53 호스팅 영역 ID에 연결합니다.

사용자 지정 호스트 이름을 추가하려면 다음 CLI 명령을 실행합니다.

```
aws transfer tag-resource --arn arn:aws:transfer:region:AWS #:server/server-ID --tags
Key=transfer:customHostname,Value="custom-host-name"
```

예제:

```
aws transfer tag-resource --arn arn:aws:transfer:us-east-1:111122223333:server/
s-1234567890abcdef0 --tags Key=transfer:customHostname,Value="abc.example.com"
```

Route 53을 사용하는 경우 다음 명령을 실행하여 사용자 지정 호스트 이름을 Route 53 호스팅 영역 ID에 연결합니다.

```
aws transfer tag-resource --arn server-ARN:server/server-ID --tags
Key=transfer:route53HostedZoneId,Value=HOSTED-ZONE-ID
```

예제:

```
aws transfer tag-resource --arn arn:aws:transfer:us-east-1:111122223333:server/
s-1234567890abcdef0 --tags Key=transfer:route53HostedZoneId,Value=ABCDE1111222233334444
```

이전 명령의 샘플 값을 가정하고 다음 명령을 실행하여 태그를 봅니다.

```
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

Amazon Route 53에서 퍼블릭, 호스팅 영역 및 해당 ID를 사용할 수 있습니다.

에 로그인 AWS Management Console 하고 <https://console.aws.amazon.com/route53/> Route 53 콘솔을 엽니다.

FTP 및 FTPS Network Load Balancer 고려 사항

AWS Transfer Family 서버 앞에서 Network Load Balancer를 사용하지 않는 것이 좋지만 FTP 또는 FTPS 구현에 클라이언트의 통신 경로에 NLB 또는 NAT가 필요한 경우 다음 권장 사항을 따르세요.

- NLB의 경우 포트 8192-8200 대신 상태 확인에 포트 21을 사용합니다.
- AWS Transfer Family 서버의 경우를 설정하여 TLS 세션 재개를 활성화합니다
다TlsSessionResumptionMode = ENFORCED.

Note

이는 향상된 보안을 제공하므로 권장되는 모드입니다.

- 클라이언트가 후속 연결에 TLS 세션 재개를 사용해야 합니다.
- 일관된 암호화 파라미터를 보장하여 더 강력한 보안 보장을 제공합니다.
- 잠재적 다운그레이드 공격을 방지하는 데 도움이 됩니다.
- 성능을 최적화하면서 보안 표준 준수를 유지합니다.

- 가능하면 NLB를 사용하지 않고 마이그레이션하여 성능 및 연결 제한을 최대한 활용 AWS Transfer Family 하세요.

NLB 대안에 대한 추가 지침은 AWS Support를 통해 AWS Transfer Family 제품 관리 팀에 문의하세요. 보안 태세 개선에 대한 자세한 내용은 블로그 게시물 [AWS Transfer Family 서버 보안을 개선하기 위한 6가지 팁을 참조하세요.](#)

NLBs 나와 있습니다 [NLBs 및 NATs 앞에 배치하지 마세요. AWS Transfer Family.](#)

클라이언트를 사용하여 서버 엔드포인트를 통한 파일 전송

클라이언트에서 전송 작업을 지정하여 AWS Transfer Family 서비스를 통해 파일을 전송합니다.는 다음 클라이언트를 AWS Transfer Family 지원합니다.

- SFTP 프로토콜 버전 3을 지원합니다.

- OpenSSH (macOS 및 Linux)

 Note

이 클라이언트는 Secure Shell (SSH) File Transfer Protocol(SSH)를 지원하는 서버에서만 작동합니다.

- WinSCP (Microsoft Windows 전용)
- Cyberduck (Windows, macOS, 및 Linux)
- FileZilla (Windows, macOS, 및 Linux)

모든 클라이언트에는 다음과 같은 제한 사항이 적용됩니다:

- SCP 프로토콜은 안전하지 않은 것으로 간주되므로 지원되지 않습니다. 예 설명된 대로 OpenSSH scp 명령을 사용할 수 있습니다 [scp 명령 사용](#).
- 연결당 최대 동시 멀티플렉스 SFTP 세션 수는 10개입니다.
- 유휴 연결의 경우 제한 시간 값은 모든 프로토콜(SFTP/FTP/FTPS)에 대해 1800초(30분)입니다. 이 기간 이후에 활동이 없으면 클라이언트 연결이 끊어질 수 있습니다. 응답하지 않는 연결의 경우:
 - 클라이언트가 완전히 응답하지 않는 경우 SFTP의 제한 시간은 300초(5분)입니다.
 - FTPS 및 FTP에는 기본 라이브러리에서 처리하는 약 10분의 응답하지 않는 제한 시간이 있습니다.
- Amazon S3와 Amazon EFS는 (NFSv4 프로토콜로 인해) 파일 명칭이 UTF-8 인코딩이어야 합니다. 다른 인코딩을 사용하면 예상치 못한 결과가 발생할 수 있습니다. Amazon S3의 경우, [객체 키 명칭 지정 가이드라인](#)을 참조하세요.
- SSL을 통한 파일 전송 프로토콜 (FTPS) 의 경우, 명시적 모드만 지원됩니다. 묵시적 모드는 지원되지 않습니다.
- FTP (파일 전송 프로토콜) 및 FTPS의 경우, 패시브 모드만 지원됩니다.
- FTP 및 FTPS의 경우, 스트림 모드만 지원됩니다.
- FTP 및 FTPS의 경우, 이미지/바이너리 모드만 지원됩니다.
- FTP 및 FTPS의 경우 데이터 연결을 위한 TLS - PROT C(보호되지 않음) TLS가 기본값이지만 AWS Transfer Family FTPS 프로토콜에서는 PROT C가 지원되지 않습니다. 따라서 FTPS의 경우, 데이터 작업이 승인되려면 PROT P를 발급해야 합니다.
- Amazon S3를 서버 스토리지로 사용하고 있고 클라이언트에 단일 전송에 복수 연결을 사용하는 옵션이 포함되어 있는 경우, 해당 옵션을 비활성화해야 합니다. 그렇지 않으면 대용량 파일 업로드가

예상치 못한 방식으로 실패할 수 있습니다. Amazon EFS를 스토리지 백엔드로 사용하는 경우, EFS는 단일 전송에 대해 복수 연결을 지원합니다.

FTP 및 FTPS에 사용할 수 있는 명령 목록은 다음과 같습니다:

가용 명령					
ABOR	FEAT	MLST	PASS	RETR	STOR
AUTH	LANG	MKD	PASV	RMD	STOU
CDUP	LIST	MODE	PBSZ	RNFR	STRU
CWD	MDTM	NLST	PROT	RNTO	SYST
DELE	MFMT	NOOP	PWD	SIZE	타입
EPSV	MLSD	OPTS	QUIT	STAT	USER

Note

APPE는 지원되지 않습니다.

SFTP의 경우, Amazon Elastic File System(Amazon EFS)을 사용하는 서버의 논리적 홈 디렉터리를 사용하는 사용자에게는 현재 다음 작업이 지원되지 않습니다.

지원되지 않는 SFTP 명령			
SSH_FXP_R EADLINK	SSH_FXP_SYMLINK	요청된 파일이 심볼릭 링크인 경우, SSH_FXP_STAT	요청된 경로에 심볼릭 링크 구성 요소가 포함된 경우, SSH_FXP_R EALPATH

퍼블릭-프라이빗 키 쌍을 생성합니다.

파일을 전송하려면 먼저 퍼블릭-프라이빗 키 쌍을 사용할 수 있어야 합니다. 이전에 키 쌍을 생성하지 않은 경우, [서비스 관리 사용자를 위한 SSH 키 생성](#)를 참조하세요.

주제

- [가용 SFTP/FTPS/FTP 명령](#)
- [Amazon VPC 엔드포인트 찾기](#)
- [setstat 오류 방지](#)
- [OpenSSH 사용](#)
- [WinSCP 사용](#)
- [Cyberduck 사용하기](#)
- [FileZilla 사용](#)
- [Perl 클라이언트 사용](#)
- [LFTP 사용](#)
- [업로드 후 처리](#)
- [SFTP 메시지](#)

가용 SFTP/FTPS/FTP 명령

다음 표에서는 SFTP AWS Transfer Family, FTPS 및 FTP 프로토콜에 사용할 수 있는 명령을 설명합니다.

Note

이 표에는 버킷과 객체만 지원하는 Amazon S3의 파일 및 디렉터리가 나와 있습니다. 계층 구조는 없습니다. 그러나 객체 키 명칭에 접두사를 사용하여 계층 구조를 암시하고 폴더와 유사한 방식으로 데이터를 구성할 수 있습니다. 이 동작은 Amazon Simple Storage Service 사용자가이드에서 [객체 메타데이터 작업](#)에 설명되어 있습니다.

SFTP/FTPS/FTP 명령

명령	Amazon S3	Amazon EFS
cd	지원됨	지원됨

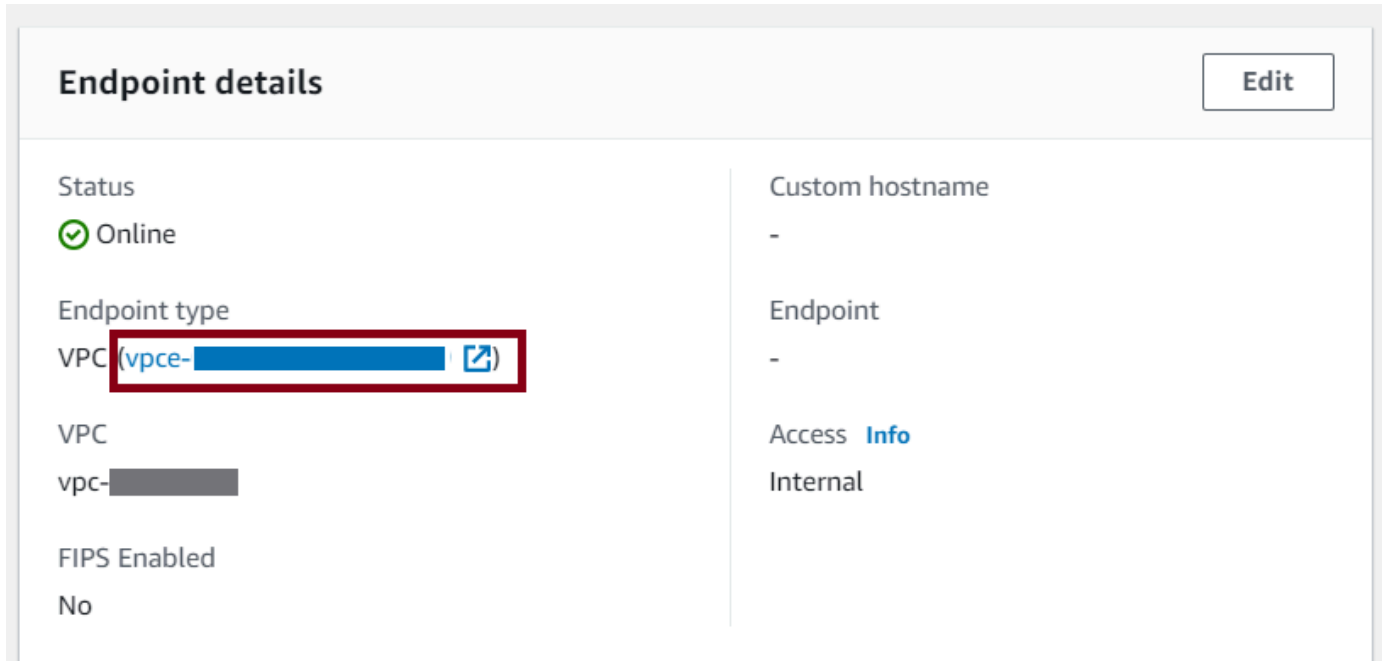
명령	Amazon S3	Amazon EFS
chgrp	지원되지 않음	지원(root 또는 owner만 해당)
chmod	지원되지 않음	지원(root만 해당)
chmtime	지원되지 않음	지원됨
chown	지원되지 않음	지원(root만 해당)
get	지원됨	지원(심볼 링크 해결 포함)
ln -s	지원되지 않음	지원
ls/dir	지원됨	지원됨
mkdir	지원됨	지원됨
put	지원됨	지원됨
pwd	지원됨	지원됨
rename	파일에만 지원됩니다.  Note 기존 파일을 덮어쓰는 이름 변경은 지원되지 않습니다.	지원됨  Note 기존 파일 또는 디렉터리를 덮어쓰는 명칭 변경은 지원되지 않습니다.
rm	지원됨	지원됨
rmdir	지원(빈 디렉터리만 해당)	지원됨
version	지원됨	지원됨

Amazon VPC 엔드포인트 찾기

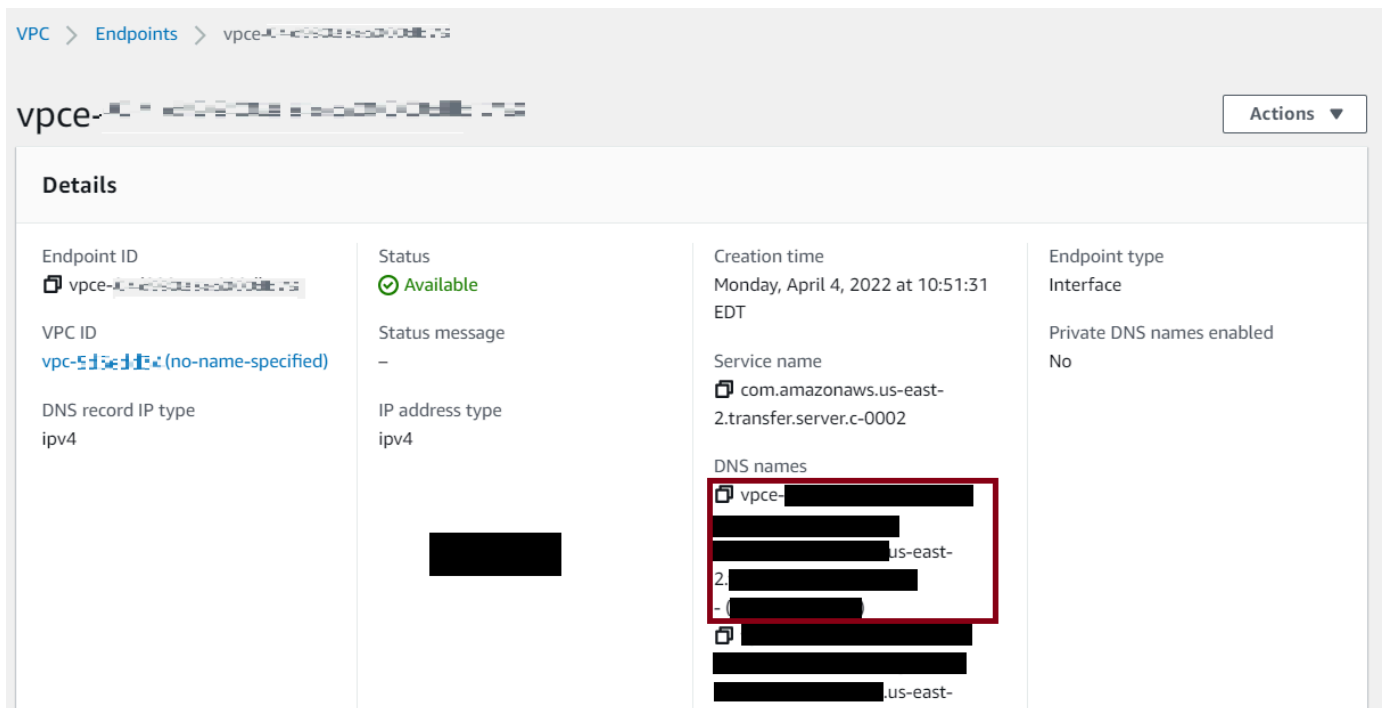
Transfer Family 서버의 엔드포인트 타입이 VPC인 경우, 파일 전송에 사용할 엔드포인트를 식별하는 것은 간단하지 않습니다. 이 경우, 다음 절차에 따라 Amazon VPC 엔드포인트를 찾습니다.

Amazon VPC 엔드포인트를 찾으려면

1. 서버의 세부 정보 페이지로 이동합니다.
2. 엔드포인트 세부 정보 창에서 VPC를 선택합니다.



3. Amazon VPC 대시보드에서 VPC 엔드포인트 ID를 선택합니다.
4. DNS 명칭 목록에서 서버 엔드포인트가 첫 번째로 나열됩니다.



setstat 오류 방지

일부 SFTP 파일 전송 클라이언트는 파일을 업로드할 때 SETSTAT 등의 명령을 사용하여 타임스탬프 및 권한 등 원격 파일의 속성을 변경하려고 시도할 수 있습니다. 그러나 이러한 명령은 Amazon S3와 같은 객체 스토리지 시스템과 호환되지 않습니다. 이러한 비호환성으로 인해 이러한 클라이언트에서 파일을 업로드하면 파일이 성공적으로 업로드된 경우에도 오류가 발생할 수 있습니다.

- CreateServer 또는 UpdateServer API를 호출할 때 클라이언트가 S3 버킷에 업로드하는 파일에 SETSTAT를 사용하려고 할 때 생성되는 오류를 무시하는 ProtocolDetails 옵션 SetStatOption를 사용합니다.
- Transfer Family 서버가 SETSTAT 명령을 무시하고 SFTP 클라이언트를 변경할 필요 없이 파일을 업로드하려면 값을 ENABLE_NO_OP로 설정합니다.
- SetStatOption ENABLE_NO_OP 설정은 오류를 무시하지만, CloudWatch 로그에 로그 항목을 생성하므로 클라이언트가 SETSTAT 호출을 하는 시점을 확인할 수 있습니다.

이 옵션에 대한 API 세부 정보는 [ProtocolDetails](#)를 참조하세요.

OpenSSH 사용

이 섹션에는 OpenSSH를 사용하여 명령줄에서 파일을 전송하는 지침이 포함되어 있습니다.

Note

이 클라이언트는 SFTP 지원 서버에서만 작동합니다.

주제

- [OpenSSH 사용](#)
- [scp 명령 사용](#)

OpenSSH 사용

OpenSSH 명령줄 유틸리티를 AWS Transfer Family 사용하여를 통해 파일을 전송하려면

1. Linux, macOS 또는 Windows에서 명령 터미널을 엽니다.
2. 프롬프트에서 다음 명령을 입력합니다.

```
sftp -i transfer-key sftp_user@service_endpoint
```

앞의 명령에서 *sftp_user*는 사용자 이름이고 *transfer-key*는 SSH 프라이빗 키입니다. 여기서 *service_endpoint*는 선택한 서버의 AWS Transfer Family 콘솔에 표시된 서버의 엔드포인트입니다.

Note

이 명령은 기본 `ssh_config` 파일에 있는 설정을 사용합니다. 이전에이 파일을 편집하지 않은 경우 SFTP는 포트 22를 사용합니다. 다음과 같이 명령에 **-P** 플래그를 추가하여 다른 포트(예: 2222)를 지정할 수 있습니다.

```
sftp -P 2222 -i transfer-key sftp_user@service_endpoint
```

또는 포트 2222 또는 포트 22000을 항상 사용하려는 경우 `ssh_config` 파일의 기본 포트를 업데이트할 수 있습니다.

sftp 프롬프트가 나타날 것입니다.

3. (옵션) 사용자의 홈 디렉터리를 보려면 sftp 프롬프트에 다음 명령을 입력합니다.

```
pwd
```

4. 파일 시스템에서 Transfer Family 서버로 파일을 업로드하려면 `put` 명령을 사용합니다. 예를 들어 업로드하려면 `hello.txt` (파일이 파일 시스템의 현재 디렉터리에 있다고 간주) `sftp` 프롬프트에서 다음 명령을 실행합니다.

```
put hello.txt
```

파일 전송이 진행 중이거나 완료되었음을 나타내는, 다음과 비슷한 메시지가 표시됩니다.

```
Uploading hello.txt to /amzn-s3-demo-bucket/home/sftp_user/hello.txt
```

```
hello.txt 100% 127 0.1KB/s 00:00
```

Note

서버가 생성된 후, 환경에서 DNS 서비스에 대한 서버 엔드포인트 호스트 명칭을 해석하려면 몇 분 정도 걸릴 수 있습니다.

scp 명령 사용

Transfer Family는 SCP 프로토콜을 지원하지 않습니다. 그러나이 기능이 필요한 경우 OpenSSH `scp` 명령을 사용할 수 있습니다.

SFTP를 통한 SCP 사용에 대한 권장 사항은 OpenSSH 버전 9.0 이상을 사용하는 것입니다. OpenSSH 버전 9 이상에서 `scp` 명령은 기본적으로 레거시 SCP 프로토콜 대신 파일 전송에 SFTP 프로토콜을 사용합니다.

Important

Transfer Family 서버가 S3 최적화 디렉터리 액세스를 사용하도록 구성되어 있는지 확인합니다.

WinSCP 사용

아래의 지침을 이용해 WinSCP를 이용하는 명령줄로 파일을 전송하세요.

Note

WinSCP 5.19를 사용하는 경우, AWS 보안 인증을 사용하여 Amazon S3에 직접 연결하고 파일을 업로드/다운로드할 수 있습니다. 자세한 내용은 [Amazon S3 서비스에 연결](#)을 참조하세요.

WinSCP AWS Transfer Family 를 사용하여를 통해 파일을 전송하려면

1. WinSCP 클라이언트를 엽니다.
2. 로그인 대화 상자의 파일 프로토콜에서 프로토콜을 선택합니다. SFTP 또는 FTP.

FTP를 선택한 경우, 암호화에서 다음 중 하나를 선택합니다.

- FTP에는 암호화가 적용되지 않습니다
- FTPS를 위한 TLS/SSL 명시적 암호화

3. 호스트 명칭에 서버 엔드포인트를 입력합니다. 서버 엔드포인트는 서버 세부 정보 페이지에 있습니다. 자세한 내용은 [SFTP, FTPS 및 FTP 서버 세부 정보 보기](#)를 참조하세요.

서버가 VPC 엔드포인트를 사용하는 경우, [Amazon VPC 엔드포인트 찾기](#)를 참조하세요.

4. 포트 번호에 다음을 입력합니다:

- SFTP에 대한 **22**
- FTP/FTPS에 대한 **21**

5. 사용자 이름에는 특정 ID 공급자용으로 생성한 사용자 이름을 입력합니다.

팁: 사용자 이름은 자격 증명 공급자에 대해 생성하거나 구성한 사용자 중 하나여야 합니다.는 다음 자격 증명 공급자를 AWS Transfer Family 제공합니다.

- [서비스 관리형과 작업](#)
- [Microsoft Active Directory용 AWS Directory Service 사용](#)
- [사용자 지정 자격 증명 공급자와 작업](#)

6. 고급을 선택하여 고급 사이트 설정 대화 상자를 엽니다. SSH 섹션에서 인증을 선택합니다.
7. 프라이빗 키 파일의 경우, 파일 시스템에서 SSH 프라이빗 키 파일을 찾아 선택합니다.

WinSCP가 SSH 프라이빗 키를 PPK 형식으로 변환하는 기능을 제공한다면, OK(확인)을 선택합니다.

8. OK(확인)을 선택해 Login(로그인) 대화 상자로 돌아간 다음 Save(저장)을 선택합니다.

9. 다음과 같은 Save session as site(세션을 사이트로 저장) 대화 상자에서 OK(확인)을 선택해 연결 설정을 완료합니다.
10. 로그인 대화 상자에서 도구를 선택한 다음 기본 설정을 선택합니다.
11. 환경 설정 대화 상자의 전송에서 Endurance를 선택합니다.

임시 파일 명칭으로 전송 재개/전송 활성화 옵션에 대해 비활성화를 선택합니다.

Important

이 옵션을 활성화한 상태로 두면 업로드 비용이 증가하여 업로드 성능이 크게 저하됩니다. 또한 대용량 파일 업로드가 실패할 수도 있습니다.

12. 전송의 경우, 배경을 선택하고 단일 전송에 복수 연결 사용 확인란의 선택을 취소합니다.

팁: 이 옵션을 선택한 상태로 두면 대용량 파일 업로드가 예측할 수 없는 방식으로 실패할 수 있습니다. 예를 들어 Amazon S3 요금이 발생하는 분리된 멀티파트 업로드를 생성할 수 있습니다. 조종한 데이터 손상도 발생할 수 있습니다.

13. 파일 전송을 실행합니다.

대상 창에서 소스 창까지 파일을 끌어다 놓아 복사할 수도 있습니다. 도구 모음 아이콘을 이용해 WinSCP의 파일을 업로드, 다운로드, 삭제, 편집하거나 속성을 수정할 수 있습니다.

Note

Amazon EFS를 스토리지로 사용하는 경우에는 이 참고 사항이 적용되지 않습니다. 타임스탬프를 포함하여 원격 파일의 속성을 변경하려는 명령은 Amazon S3와 같은 객체 스토리지 시스템과 병립되지 않습니다. 따라서 Amazon S3를 스토리지로 사용하는 경우, 파일 전송을 수행하기 전에 WinSCP 타임스탬프 설정을 비활성화해야 합니다(또는 [setstat 오류 방지](#)에 설명된 대로 SetStatOption 사용). 이렇게 하려면, WinSCP Transfer 설정 대화 상자에서 권한 설정 업로드 옵션과 타임스탬프 유지 일반 옵션을 비활성화해야 합니다.

Cyberduck 사용하기

아래의 지침을 이용해 Cyberduck을 이용하는 명령줄로 파일을 전송하세요.

Cyberduck을 AWS Transfer Family 사용하여 통해 파일을 전송하려면

1. [Cyberduck](#) 클라이언트를 엽니다.
2. 연결 열기를 선택합니다.
3. 연결 열기 대화 상자에서 프로토콜을 선택합니다. SFTP (SSH 파일 전송 프로토콜), FTP-SSL (명시적 AUTH TLS), 또는 FTP (파일 전송 프로토콜).
4. 서버에 서버 엔드포인트를 입력합니다. 서버 엔드포인트는 서버 세부 정보 페이지에 있습니다. 자세한 내용은 [SFTP, FTPS 및 FTP 서버 세부 정보 보기](#)를 참조하세요.

서버가 VPC 엔드포인트를 사용하는 경우, [Amazon VPC 엔드포인트 찾기](#)를 참조하세요.

5. 포트 번호에 다음을 입력합니다:
 - SFTP에 대한 **22**
 - FTP/FTPS에 대한 **21**
6. 사용자 이름에 [서버 엔드포인트에 대한 사용자 관리](#)에서 생성한 사용자 이름을 입력합니다.
7. SFTP를 선택한 경우, SSH 프라이빗 키에 대해 SSH 프라이빗 키를 선택하거나 입력합니다.
8. 연결을 선택합니다.
9. 파일 전송을 실행합니다.

파일 위치에 따라, 다음 중 하나를 수행하세요.

- 로컬 디렉터리(소스)에서 전송할 파일을 선택하고, Amazon S3 디렉터리(대상)로 끌어다 놓습니다.
- Amazon S3 디렉터리(소스)에서 전송할 파일을 선택하고, 로컬 디렉터리(대상)로 끌어다 놓습니다.

FileZilla 사용

다음 지침을 이용해 FileZilla를 사용하여 파일을 전송하세요.

파일 전송을 위한 FileZilla 설정 방법

1. FileZilla 클라이언트를 엽니다.
2. 파일을 선택한 다음 사이트 관리자를 선택합니다.
3. 사이트 관리자 대화 상자에서 새 사이트를 선택합니다.
4. 일반 탭에서 프로토콜을 위해 프로토콜을 선택합니다. SFTP 또는 FTP.

FTP를 선택한 경우, 암호화에서 다음 중 하나를 선택합니다.

- 일반 FTP(비보안)만 사용 – FTP의 경우
- 가능한 경우, TLS를 통한 명시적 FTP를 사용 – FTPS의 경우

5. 호스트 명칭에는 사용 중인 프로토콜을 입력한 다음 서버 엔드포인트를 입력합니다. 서버 엔드포인트는 서버 세부 정보 페이지에 있습니다. 자세한 내용은 [SFTP, FTPS 및 FTP 서버 세부 정보 보기](#) 단원을 참조하십시오.

- SFTP를 사용하는 경우, 다음을 입력합니다: `sftp://hostname`
- FTPS를 사용하는 경우, 다음을 입력합니다: `ftps://hostname`

##을 실제 서버 엔드포인트로 바꿔야 합니다.

서버가 VPC 엔드포인트를 사용하는 경우, [Amazon VPC 엔드포인트 찾기](#)를 참조하세요.

6. 포트 번호에 다음을 입력합니다:

- SFTP에 대한 **22**
- FTP/FTPS에 대한 **21**

7. SFTP를 선택한 경우, 로그인 타입으로 키 파일을 선택합니다.

키 파일의 경우, SSH 프라이빗 키를 선택하거나 입력합니다.

8. 사용자 이름에 [서버 엔드포인트에 대한 사용자 관리](#)에서 생성한 사용자 이름을 입력합니다.

9. 연결을 선택합니다.

10. 파일 전송을 실행합니다.

Note

진행 중인 파일 전송을 중단하면 Amazon S3 버킷에 부분 객체를 AWS Transfer Family 쓸 수 있습니다. 업로드를 중단했다면, Amazon S3 버킷의 파일 크기가 소스 개체의 파일 크기와 동일한지 확인한 후 다음 단계로 계속합니다.

Perl 클라이언트 사용

NET: :SFTP: :Foreign 펄 클라이언트를 사용하는 경우, `queue_size`를 1으(로) 설정해야 합니다. 예제:

```
my $sftp = Net::SFTP::Foreign->new('user@s-12345.server.transfer.us-east-2.amazonaws.com', queue_size => 1);
```

Note

이 해결 방법은 [1.92.02](#) 이전의 Net::SFTP::Foreign을 위해 필요합니다.

LFTP 사용

LFTP는 사용자가 대부분의 Linux 시스템에서 명령줄 인터페이스를 통해 파일 전송을 수행할 수 있는 무료 FTP 클라이언트입니다.

대용량 파일 다운로드의 경우 LFTP에 순서가 맞지 않는 패킷과 관련된 알려진 문제가 있어 파일 전송이 실패합니다.

업로드 후 처리

Amazon S3 객체 메타데이터 및 이벤트 알림을 비롯한 업로드 후 처리 정보를 볼 수 있습니다.

주제

- [Amazon S3 객체 메타데이터](#)
- [Amazon S3 이벤트 알림](#)

Amazon S3 객체 메타데이터

객체 메타데이터의 일부로, 'x-amz-meta-user-agent'의 값은 AWSTransfer(0)고 x-amz-meta-user-agent-id의 값은 username@server-id(0)라는 키가 표시됩니다. username은(는) 파일을 업로드한 Transfer Family 사용자이며, server-id은(는) 업로드에 사용되는 서버입니다. Lambda 함수 내의 S3 객체에서 [HeadObject](#) 작업을 사용하여 이 정보에 액세스할 수 있습니다.

Key	Value
Content-Type	text/plain
x-amz-meta-user-agent	AWSTransfer
x-amz-meta-user-agent-id	sftp_user@s-

Amazon S3 이벤트 알림

Transfer Family를 사용하여 S3 버킷에 객체를 업로드하면 RoleSessionName이(가) [S3 이벤트 알림 구조](#)의 요청자 필드에 [AWS:Role Unique Identifier]/username.sessionid@server-id와(과) 같이 포함됩니다. 예를 들어, 다음은 S3 버킷으로 복사된 파일에 대한 S3 액세스 로그의 샘플 요청자 필드 콘텐츠입니다.

```
arn:aws:sts::AWS-Account-ID:assumed-role/IamRoleName/
username.sessionid@server-id
```

위의 요청자 필드에는 IamRoleName(이)라고 불리는 IAM 역할이 표시됩니다. S3 이벤트 알림 구성에 대한 자세한 설명은 Amazon Simple Storage Service 개발자 가이드의 [Configuring Amazon S3 이벤트 알림](#)을 참조하세요. AWS Identity and Access Management (IAM) 역할 고유 식별자에 대한 자세한 내용은 AWS Identity and Access Management 사용 설명서의 [고유 식별자](#)를 참조하세요.

SFTP 메시지

이 섹션에서는 Transfer Family 서버를 사용할 때 SFTP 파일 전송 중 또는 전송 후에 수신할 수 있는 클라이언트 측 메시지를 설명합니다. SFTP 이벤트에 대한 자세한 내용은 SFTP 클라이언트 로그를 확인하세요. 이 정보를 사용하여 오류를 해결하거나 문제를 식별하는 데 도움이 되도록 네트워크 팀에 해당 정보를 전달할 수 있습니다.

SFTP 클라이언트 측 메시지

활동	설명
AUTH_FAILURE	사용자가 인증에 실패했습니다. 이는 사용자 지정 자격 증명 공급자 또는 서비스 관리형 사용자의 모든 종류의 실패일 수 있습니다. 이벤트의 세부 정보는 실패의 근본 원인을 명확히 하는 데 도움이 됩니다.
CLOSE	열려 있는 파일 또는 디렉터리가 성공적으로 닫혔음을 나타냅니다.
연결됨/연결 해제됨	정상적인 연결 성공 및 연결 해제를 나타냅니다.
CREATE_SYMLINK	심볼 링크가 생성되었습니다(성공 또는 실패).
DELETE	파일이 삭제되었습니다(성공 또는 실패).
오류	예상치 못한 일반적인 오류입니다. 관련 설명에는 사용자 또는 네트워크 관리자가 특정 문제를 식별하는 데 도움이 될 수 있는 정보가 포함되어 있습니다.
EXIT_REASON	예기치 않은 오류로 인해 SFTP 세션이 종료될 때 발생합니다. 이벤트와 연결된 메시지는 원인을 설명합니다.
MKDIR	디렉터리가 생성되었습니다(성공 또는 실패).
OPEN	파일이 읽기 또는 쓰기를 위해 열렸습니다(성공 또는 실패).
PARTIAL_CLOSE	파일이 아직 열려 있지만 CLOSE 메시지가 수신되지 않은 상태에서 클라이언트가 서버에서 연결 해제되었습니다. Transfer Family는 파일의 수신된 부분(실제로 전체 파일일 수 있음)을 저장하고 PARTIAL_CLOSE 이벤트를 내보내 고객에게 문제를 알립니다. 워크플로 통합은 파일을

활동	설명
	적절하게 처리하는 onPartialClose 이벤트도 수신합니다.
RENAME	파일 이름이 변경되었습니다(성공 또는 실패).
RMDIR	디렉터리가 삭제되었습니다(성공 또는 실패).
SETSTAT	파일의 속성이 변경됩니다(성공 또는 실패).
 Note 스토리지에 Amazon S3를 사용하는 경우 Transfer Family는 SETSTAT를 지원하지 않습니다. 이 setstat 오류 방지 섹션에서는 설정을 해제하여 SetStat 오류를 방지하는 방법에 대한 세부 정보를 제공합니다. 이렇게 하면 fail unsupported error 대신 success but do nothing 메시지가 수신됩니다.	
TLS_RESUME_FAILURE	서버는 TLS 세션 재개를 적용하도록 구성되며 클라이언트는 이를 지원하지 않습니다.

서버 엔드포인트에 대한 사용자 관리

다음 섹션에서는 AWS Transfer Family AWS Directory Service for Microsoft Active Directory 또는 사용자 지정 자격 증명 공급자를 사용하여 사용자를 추가하는 방법에 대한 정보를 찾을 수 있습니다.

또한 각 사용자 속성의 일부로, 사용자의 SSH(보안 셸) 퍼블릭 키를 저장합니다. 키 기반 인증에는 이렇게 해야 합니다. 프라이빗 키는 사용자의 컴퓨터에 로컬로 저장됩니다. 사용자가 클라이언트를 사용하여 서버에 인증 요청을 보내면, 서버에서는 먼저 사용자를 사용자가 연관된 SSH 프라이빗 키에 대한 액세스 권한을 가지고 있는지 확인합니다. 그러면 서버가 사용자를 성공적으로 인증합니다.

Note

여러 SSH 키가 있는 사용자의 자동 배포 및 관리는 섹션을 참조하세요 [Transfer Family Terraform 모듈](#).

또한 사용자의 홈 디렉터리 또는 랜딩 디렉터리를 지정하고, AWS Identity and Access Management (IAM) 역할을 사용자에게 할당합니다. 원한다면 세션 정책을 제공해 사용자 액세스를 Amazon S3 버킷의 홈 디렉터리로 제한할 수도 있습니다.

Important

AWS Transfer Family 는 SFTP 서버에 대한 인증에서 1~2자 길이의 사용자 이름을 차단합니다. 또한 root 사용자 이름도 차단합니다.

그 이유는 암호 스캐너의 악의적인 로그인 시도가 대량으로 발생하기 때문입니다.

아마존 EFS와 아마존 S3

각 스토리지 옵션의 특징:

- 액세스를 제한하려면: Amazon S3는 세션 정책을 지원하고, Amazon EFS는 POSIX 사용자, 그룹 및 보조 그룹 ID를 지원합니다.
- 둘 다 퍼블릭/프라이빗 키 지원
- 둘 다 홈 디렉터리 지원
- 둘 다 논리적 디렉터리 지원

Note

Amazon S3의 경우 논리적 디렉터리에 대한 대부분의 지원은 API/CLI를 통해 이루어집니다. 콘솔의 제약 확인란을 사용하여 사용자를 홈 디렉터리에 잠글 수 있지만 가상 디렉터리 구조를 지정할 수는 없습니다.

논리적 디렉터리

사용자의 논리적 디렉터리 값을 지정하는 경우 사용하는 파라미터는 사용자 타입에 따라 달라집니다.

- 서비스 관리 사용자의 경우 HomeDirectoryMappings에서 논리적 디렉터리 값을 제공하세요.
- 사용자 지정 자격 증명 공급자 사용자의 경우에 논리적 디렉터리 값을 제공합니다.
다HomeDirectoryDetails.

AWS Transfer Family 는 LOGICAL HomeDirectoryType을 사용할 때 HomeDirectory 값 지정을 지원합니다. 이는 응답에 HomeDirectoryDetails가 제공되는 서비스 관리형 사용자, Active Directory 액세스 및 사용자 지정 자격 증명 공급자 구현에 적용됩니다.

Important

LOGICAL HomeDirectoryType을 사용하여 HomeDirectory를 지정할 때 값은 논리적 디렉터리 매핑 중 하나에 매핑되어야 합니다. 이 서비스는 사용자 생성 및 업데이트 중에 이를 검증하여 작동하지 않는 구성을 방지합니다.

기본 동작

기본적으로 지정하지 않은 상태로 두면 LOGICAL 모드에 대해 HomeDirectory가 "/"로 설정됩니다. 이 동작은 변경되지 않으며 기존 사용자 정의와 호환됩니다.

- HomeDirectory를 대상이 아닌 항목에 매핑해야 합니다. 자세한 내용은 [논리적 디렉터리 사용 규칙](#) 섹션을 참조하세요.
- 가상 디렉터리의 구조에 대한 자세한 내용은 단원을 참조하십시오 [가상 디렉터리 구조](#).

사용자 지정 자격 증명 공급자 고려 사항

사용자 지정 자격 증명 공급자를 사용하는 경우 이제 LOGICAL HomeDirectoryType을 사용하는 동안 응답에 HomeDirectory를 지정할 수 있습니다. TestIdentityProvider API 호출은 사용자 지정 IDP가 LOGICAL 모드에서 HomeDirectory를 지정할 때 올바른 결과를 생성합니다.

HomeDirectory 및 LOGICAL HomeDirectoryType을 사용한 사용자 지정 IDP 응답의 예:

```
{
  "Role": "arn:aws:iam::123456789012:role/transfer-user-role",
  "HomeDirectoryType": "LOGICAL",
  "HomeDirectory": "/marketing",
  "HomeDirectoryDetails": "[{\"Entry\":\"/\",\"Target\":\"/bucket/home\"},{\"Entry\":\"/marketing\",\"Target\":\"/marketing-bucket/campaigns\"}]\""
```

}

Active Directory 그룹 할당량

AWS Transfer Family 의 기본 제한은 서버당 Active Directory 그룹 100개입니다. 사용 사례에 100개 이상의 그룹이 필요한 경우 용 사용자 지정 자격 증명 공급자를 [사용한 Active Directory 인증 간소화에 설명된 대로 사용자 지정 자격 증명 공급자 AWS Transfer Family](#) 솔루션을 사용하는 것이 좋습니다.

이 제한은 다음 자격 증명 공급자를 사용하는 서버에 적용됩니다.

- AWS Microsoft Active Directory용 디렉터리 서비스
- AWS Entra ID 도메인 서비스를 위한 디렉터리 서비스

서비스 한도 증가를 요청해야 하는 경우의 [AWS 서비스 할당량을 참조하세요](#) AWS 일반 참조. 사용 사례에 100개 이상의 그룹이 필요한 경우 용 사용자 지정 자격 증명 공급자를 [사용한 Active Directory 인증 간소화에 설명된 대로 사용자 지정 자격 증명 공급자 AWS Transfer Family](#) 솔루션을 사용하는 것이 좋습니다.

Active Directory 그룹 제한과 관련된 문제 해결 정보는 섹션을 참조하세요 [Active Directory 그룹 제한 초과](#).

주제

- [서비스 관리형과 작업](#)
- [사용자 지정 자격 증명 공급자와 작업](#)
- [Microsoft Active Directory용 AWS Directory Service 사용](#)
- [Entra ID 도메인 서비스에 AWS 디렉터리 서비스 사용](#)

서비스 관리형과 작업

서버의 도메인 설정에 따라 Amazon S3 또는 Amazon EFS 서비스 관리 사용자를 서버에 추가할 수 있습니다. 자세한 내용은 [SFTP, FTPS 또는 FTP 서버 엔드포인트 구성](#) 단원을 참조하십시오.

서비스 관리형 자격 증명 타입을 사용한다면, 사용자를 파일 전송 프로토콜 지원 서버에 추가해야 합니다. 이 작업을 할 때 각 사용자 이름은 서버 내에서 고유해야 합니다.

서비스 관리 사용자를 프로그래밍 방식으로 추가하려면 [CreateUser](#) API의 [예](#)를 참조하세요.

Note

서비스 관리형 사용자의 경우 논리적 디렉터리 항목은 2,000개로 제한됩니다. 논리적 디렉터리 사용에 대한 자세한 내용은 섹션을 참조하세요 [논리적 디렉터리를 사용하여 Transfer Family 디렉터리 구조를 단순화합니다..](#)

주제

- [Amazon S3 서비스 관리 사용자 추가](#)
- [Amazon EFS 서비스 관리 사용자 추가](#)
- [서비스 관리 사용자 관리](#)

Amazon S3 서비스 관리 사용자 추가**Note**

교차 계정 Amazon S3 버킷을 구성하려면 지식 센터 문서: [다른 AWS 계정에 있는 Amazon Simple Storage Service 버킷을 사용하도록 AWS Transfer Family 서버를 구성하려면 어떻게 해야 합니까?](#)에 설명된 단계를 따르세요.

Amazon S3 서비스 관리 사용자를 서버에 추가하려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 연 다음 탐색 창에서 서버를 선택합니다.
2. 서버 페이지에서 사용자를 추가할 서버의 확인란을 선택합니다.
3. 사용자 추가를 선택합니다.
4. 사용자 구성 섹션의 사용자 이름에 사용자 이름을 입력합니다. 이 사용자 이름은 3~100자여야 합니다. 사용자 이름에 a~z, A~Z, 0~9, 밑줄 '_', 하이픈 '-', 마침표 '.' 및 at 기호 '@' 문자를 사용할 수 있습니다. 사용자 이름은 하이픈 '-', 마침표 '.' 또는 at 기호 '@'로 시작할 수 없습니다.
5. 액세스에는 Amazon S3 버킷에 대한 액세스를 제공하는, 이전에 생성한 IAM 역할을 선택합니다.

이 IAM 역할은 [IAM 역할 및 정책 생성](#)의 절차를 사용하여 생성했습니다. 해당 IAM 역할은 Amazon S3 버킷에 대한 액세스를 제공하는 IAM 정책을 포함합니다. 또한 다른 IAM 정책에 정의된 AWS Transfer Family 서비스와의 신뢰 관계도 포함됩니다. 사용자에게 대한 세분화된 액세스 제

어가 필요한 경우 [AWS Transfer Family 및 Amazon S3로 데이터 액세스 제어 향상](#) 블로그 게시물을 참조하세요.

6. (옵션) 정책에서 다음 중 하나를 선택합니다.

- 없음
- 기존 정책
- IAM에서 정책 선택: 기존 세션 정책을 선택할 수 있습니다. 정책의 세부 정보가 포함된 JSON 객체를 보려면 보기를 선택합니다.
- 홈 폴더 기반 정책 자동 생성: 세션 정책을 자동으로 생성합니다. 정책의 세부 정보가 포함된 JSON 객체를 보려면 보기를 선택합니다.

 Note

홈 폴더를 기반으로 정책 자동 생성을 선택하는 경우 이 사용자에게 대해 제약을 선택하지 마세요.

세션 정책에 대한 자세한 내용은 [IAM 역할 및 정책 생성](#), [Amazon S3 버킷을 위한 세션 정책 생성](#) 또는 단원을 참조하십시오 [동적 권한 관리 접근 방식](#).

7. 홈 디렉터리에서 전송할 데이터를 저장할 Amazon S3 버킷을 선택합니다 AWS Transfer Family. 사용자가 클라이언트를 이용해 로그인하면 도착하게 되는 home 디렉터리의 경로를 입력합니다.

이 파라미터를 입력하지 않으면 Amazon 버킷의 root 디렉터리를 사용합니다. 이 경우 IAM 역할이 이 root 디렉터리에 대한 액세스를 제공하는지 확인하세요.

 Note

사용자의 사용자 이름을 포함하는 디렉터리 경로 선택을 권장합니다. 세션 정책을 효과적으로 사용할 수 있습니다. 세션 정책은 사용자의 home 디렉터리에 대한 Amazon S3 버킷에서의 사용자 액세스를 제한합니다.

8. (옵션) 제약의 경우, 사용자가 해당 폴더 외부에 액세스할 수 없고 Amazon S3 버킷 또는 폴더 이름을 볼 수 없도록 확인란을 선택합니다.

Note

사용자에게 홈 디렉터리를 할당하고 사용자를 해당 홈 디렉터리로 제한하면 지정된 폴더에 대한 사용자 액세스를 차단하기에 충분해야 합니다. 추가 통제를 적용해야 하는 경우 세션 정책을 사용하세요.

이 사용자에게 제약 선택하면 홈 폴더가 제한된 사용자에게 대해 정의된 값이 아니기 때문에 홈 폴더 기반 정책 자동 생성을 선택할 수 없습니다.

9. SSH 퍼블릭 키에는 SSH 키 쌍의 퍼블릭 SSH 키 부분을 입력합니다.

서비스에서 사용자의 키를 확인해야 새 사용자를 추가할 수 있습니다.

Note

SSH 키 쌍을 만드는 방법에 대한 지침은 [서비스 관리 사용자를 위한 SSH 키 생성](#)을 참조하세요.

10. (옵션) 키와 값에 하나 이상의 태그를 키-값 쌍으로 입력하고 태그 추가를 선택합니다.
11. 추가를 선택해 새 사용자를 원하는 서버에 추가합니다.

새 사용자는 서버 세부 정보 페이지의 사용자 섹션에 나타납니다.

다음 단계 - 그 다음 단계에서는 [클라이언트를 사용하여 서버 엔드포인트를 통한 파일 전송](#)로 계속 진행하세요.

Amazon EFS 서비스 관리 사용자 추가

Amazon EFS는 Portable Operating System Interface(POSIX) 파일 권한 모델을 사용하여 파일 소유권을 나타냅니다.

- Amazon EFS 파일 소유권에 대한 자세한 설명은 [Amazon EFS 파일 소유권](#)을 참조하세요.
- EFS 사용자를 위한 디렉터리 설정에 대한 자세한 설명은 [Transfer Family를 위한 Amazon EFS 사용자 설정](#) 섹션을 참조하세요.

Amazon EFS 서비스 관리 사용자를 서버에 추가하려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 연 다음 탐색 창에서 서버를 선택합니다.
2. 서버 페이지에서 사용자를 추가하려는 Amazon EFS 서버를 선택합니다.
3. 사용자 추가를 선택하여 사용자 추가 페이지를 표시합니다.
4. 사용자 구성 섹션에서 다음 설정을 구성합니다.
 - a. 이 사용자 이름은 최소 3글자에서 최대 100자여야 합니다. 사용자 이름에는 a~z, A~Z, 0~9, 밑줄(_)와 하이픈(-), 마침표(.) 및 골뱅이 기호(@) 등의 문자를 사용할 수 있습니다. 사용자 이름은 하이픈 '-', 마침표 '.' 및 "@" 기호로 시작할 수 없습니다.
 - b. 사용자 ID 및 그룹 ID의 경우, 다음을 참고하세요:
 - 처음 생성하는 사용자의 경우, 그룹 ID와 사용자 ID 모두에 대한 0의 값을 입력하는 것이 좋습니다. 이렇게 하면 Amazon EFS에 대한 사용자 관리자 권한이 부여됩니다.
 - 추가 사용자의 경우, 사용자의 POSIX 사용자 ID 및 그룹 ID를 입력합니다. 이러한 ID는 사용자가 수행하는 모든 Amazon Elastic File System 작업에 사용됩니다.
 - 사용자 ID 및 그룹 ID의 경우 앞에 0을 사용하지 마세요. 예컨대, **12345**는 허용되지만 **012345**는 허용되지 않습니다.
 - c. (옵션) 보조 그룹 ID의 경우 각 사용자에게 대해 하나 이상의 추가 POSIX 그룹 ID를 쉼표로 구분하여 입력합니다.
 - d. 액세스에서 다음과 같은 IAM 역할을 선택합니다.
 - 액세스하려는 Amazon EFS 리소스(파일 시스템)에 대한 액세스 권한만 사용자에게 부여합니다.
 - 사용자가 수행할 수 있는 파일 시스템 작업과 수행할 수 없는 파일 시스템 작업을 정의합니다.

마운트 액세스 및 읽기/쓰기 권한이 있는 Amazon EFS 파일 시스템 선택에는 IAM 역할을 사용하는 것이 좋습니다. 예를 들어, 다음 두 AWS 관리형 정책의 조합은 매우 허용적이지만 사용자에게 필요한 권한을 부여합니다.

- AmazonElasticFileSystemClientFullAccess
- AWSTransferConsoleFullAccess

자세한 내용은 블로그 게시물 [Amazon Elastic File System을 위한AWS Transfer Family 지원](#)을 참조하세요.

e. 홈 디렉터리의 경우, 다음을 수행합니다.

- AWS Transfer Family를 사용하여 전송할 데이터를 저장하는 데 사용할 Amazon EFS 파일 시스템을 선택합니다.
- 홈 디렉터리를 제약으로 설정할지 여부를 결정하세요. 홈 디렉터리를 제약으로 설정하면 다음과 같은 결과가 발생합니다:
 - Amazon EFS 사용자는 해당 폴더 외부의 파일이나 디렉터리에 액세스할 수 없습니다.
 - 아마존 EFS 사용자는 아마존 EFS 파일 시스템 이름(fs-xxxxxxx)을 볼 수 없습니다.

 Note

제약 옵션을 선택하면 Amazon EFS 사용자의 경우 심볼릭 링크가 확인되지 않습니다.

- (옵션) 사용자가 클라이언트를 사용하여 로그인할 때 들어갈 홈 디렉터리의 경로를 입력합니다.

홈 디렉터리를 지정하지 않는 경우 Amazon EFS 파일 시스템의 루트 디렉터리가 사용됩니다. 이 경우 IAM 역할이 이 루트 디렉터리에 대한 액세스를 제공하는지 확인하세요.

5. SSH 퍼블릭 키에는 SSH 키 쌍의 퍼블릭 SSH 키 부분을 입력합니다.

서비스에서 사용자의 키를 확인해야 새 사용자를 추가할 수 있습니다.

 Note

SSH 키 쌍을 만드는 방법에 대한 지침은 [서비스 관리 사용자를 위한 SSH 키 생성](#)을 참조하세요.

6. (옵션) 사용자의 태그를 입력하세요. 키 및 값에 하나 이상의 태그를 키-값 쌍으로 입력하고 태그 추가를 선택합니다.
7. 추가를 선택해 새 사용자를 원하는 서버에 추가합니다.

새 사용자는 서버 세부 정보 페이지의 사용자 섹션에 나타납니다.

Transfer Family 서버에 처음 SFTP를 연결할 때 발생할 수 있는 문제는 다음과 같습니다:

- sftp 명령을 실행해도 프롬프트가 나타나지 않으면 다음 메시지가 표시될 수 있습니다.

```
Couldn't canonicalize: Permission denied
```

```
Need cwd
```

이 경우 사용자 역할에 대한 정책 권한을 늘려야 합니다. 와 같은 AWS 관리형 정책을 추가할 수 있습니다 AmazonElasticFileSystemClientFullAccess.

- 사용자의 홈 디렉터리를 보라는 sftp 프롬프트에 pwd를 입력하면 다음 메시지가 표시될 수 있습니다. 여기서 **USER-HOME-DIRECTORY**는 SFTP 사용자의 홈 디렉터리입니다.

```
remote readdir("/USER-HOME-DIRECTORY"): No such file or directory
```

이 경우 상위 디렉터리(cd ..)로 이동하여 사용자의 홈 디렉터리(mkdir **username**)를 만들 수 있어야 합니다.

다음 단계 - 그 다음 단계에서는 [클라이언트를 사용하여 서버 엔드포인트를 통한 파일 전송](#)로 계속 진행하세요.

서비스 관리 사용자 관리

이 섹션에서는 사용자 목록을 보는 방법, 사용자 세부 정보를 편집하는 방법, SSH 퍼블릭 키를 추가하는 방법에 대한 정보를 찾을 수 있습니다.

- [사용자 목록 보기](#)
- [사용자 세부 정보 보기 또는 편집](#)
- [사용자 삭제](#)
- [SSH 퍼블릭 키 추가](#)
- [SSH 퍼블릭 키 삭제](#)

사용자 목록을 찾으려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 탐색 창에서 서버를 선택하여 서버 페이지를 표시합니다.
3. 서버 ID 옆에서 식별자를 선택하여 서버 세부 정보 페이지를 표시합니다.

4. 사용자에서 사용자 목록을 확인합니다.

사용자 세부 정보를 보거나 편집하려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 탐색 창에서 서버를 선택하여 서버 페이지를 표시합니다.
3. 서버 ID 옆에서 식별자를 선택하여 서버 세부 정보 페이지를 표시합니다.
4. 사용자에서 사용자 이름을 선택하면 사용자 세부 정보 페이지가 표시됩니다.

편집을 선택하여 이 페이지에서 사용자의 속성을 변경할 수 있습니다.

5. 사용자 세부 정보 페이지에서 사용자 구성 옆의 편집을 선택합니다.

Edit configuration

User configuration

Access [Info](#)
User's IAM role for Amazon S3 access

Admin ▼

Policy [Info](#)
Scope down policy to apply to the user

☒ None
☐ Existing policy
☐ Select a policy from IAM

[View](#)

Home directory
User's login directory

Choose an S3 bucket ▼

Enter optional folder

☐ Restricted [Info](#)

Cancel Save

6. 구성 편집 페이지의 Access에서 Amazon S3 버킷에 대한 액세스를 제공하는 이전에 생성한 IAM 역할을 선택합니다.

이 IAM 역할은 [IAM 역할 및 정책 생성](#)의 절차를 사용하여 생성했습니다. 해당 IAM 역할은 Amazon S3 버킷에 대한 액세스를 제공하는 IAM 정책을 포함합니다. 또한 다른 IAM 정책에 정의된 AWS Transfer Family 서비스와의 신뢰 관계도 포함됩니다.

7. (옵션) 정책에서 다음 중 하나를 선택합니다.

- 없음
- 기존 정책
- IAM에서 정책을 선택하여 기존 정책을 선택합니다. 정책의 세부 정보가 포함된 JSON 객체를 보려면 보기를 선택합니다.

세션 정책에 대한 자세한 설명은 [IAM 역할 및 정책 생성](#) 섹션을 참조하세요. 세션 정책 생성에 대한 자세한 설명은 [Amazon S3 버킷을 위한 세션 정책 생성](#) 섹션을 참조하세요.

8. 홈 디렉터리에서 전송할 데이터를 저장할 Amazon S3 버킷을 선택합니다 AWS Transfer Family. 사용자가 클라이언트를 이용해 로그인하면 도착하게 되는 home 디렉터리의 경로를 입력합니다.

이 파라미터를 입력하지 않으면 Amazon S3 버킷의 root 디렉터리가 사용됩니다. 이 경우 IAM 역할이 이 root 디렉터리에 대한 액세스를 제공하는지 확인하세요.

Note

사용자의 사용자 이름을 포함하는 디렉터리 경로 선택을 권장합니다. 세션 정책을 효과적으로 사용할 수 있습니다. 세션 정책은 사용자의 home 디렉터리에 대한 Amazon S3 버킷에서의 사용자 액세스를 제한합니다.

9. (옵션) 제약의 경우, 사용자가 해당 폴더 외부에 액세스할 수 없고 Amazon S3 버킷 또는 폴더 이름을 볼 수 없도록 확인란을 선택합니다.

Note

사용자에게 홈 디렉터리를 할당하고 해당 홈 디렉터리로 사용자를 제한하는 경우 이 정도면 지정된 폴더에 대한 사용자 액세스를 잠그기에 충분합니다. 추가 통제를 적용해야 하는 경우 세션 정책을 사용하세요.

10. 저장을 선택하여 변경 사항을 저장합니다.

사용자 삭제

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 탐색 창에서 서버를 선택하여 서버 페이지를 표시합니다.
3. 서버 ID 옆에서 식별자를 선택하여 서버 세부 정보 페이지를 표시합니다.
4. 사용자에서 사용자 이름을 선택하면 사용자 세부 정보 페이지가 표시됩니다.

5. 사용자 세부 정보 페이지에서 사용자 이름 오른쪽에 있는 삭제를 선택합니다.
6. 표시되는 확인 대화 상자에서 단어 **delete**을 입력한 다음 삭제를 선택하여 사용자 삭제를 확인합니다.

사용자 목록에서 사용자가 삭제됩니다.

사용자의 SSH 퍼블릭 키를 추가하려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 탐색 창에서 서버를 선택합니다.
3. 서버 ID 옆에서 식별자를 선택하여 서버 세부 정보 페이지를 표시합니다.
4. 사용자에서 사용자 이름을 선택하면 사용자 세부 정보 페이지가 표시됩니다.
5. Add SSH public key(SSH 퍼블릭 키 추가)를 선택해 새 SSH 퍼블릭 키를 사용자에게 추가합니다.

Note

SSH 키는 SSH(Secure Shell) File Transfer Protocol(SFTP)을 지원하는 서버에서만 사용됩니다. SSH 키 쌍을 생성하는 자세한 방법은 [서비스 관리 사용자를 위한 SSH 키 생성](#) 섹션을 참조하세요.

6. SSH public key(SSH 퍼블릭 키)에는 SSH 키 쌍의 SSH 퍼블릭 키 부분을 입력합니다.

서비스에서 사용자의 키를 확인해야 새 사용자를 추가할 수 있습니다. SSH 키의 형식은 `ssh-rsa string`입니다. SSH 키 쌍을 생성하려면 [서비스 관리 사용자를 위한 SSH 키 생성](#) 섹션을 참조하세요.

7. 키 추가를 선택합니다.

사용자의 SSH 퍼블릭 키를 삭제하려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 탐색 창에서 서버를 선택합니다.
3. 서버 ID 옆에서 식별자를 선택하여 서버 세부 정보 페이지를 표시합니다.
4. 사용자에서 사용자 이름을 선택하면 사용자 세부 정보 페이지가 표시됩니다.
5. 퍼블릭 키를 삭제하려면 해당 SSH 키 확인란을 선택하고 삭제를 선택합니다.

사용자 지정 자격 증명 공급자와 작업

AWS Transfer Family 는 사용자 지정 자격 증명 공급자가 안전한 파일 전송을 위해 사용자를 인증하고 권한을 부여할 수 있는 몇 가지 옵션을 제공합니다. 다음은 주요 접근 방식입니다.

- [사용자 지정 자격 증명 공급자 솔루션](#)-이 주제에서는 GitHub에서 호스팅되는 툴킷을 사용하여 Transfer Family 사용자 지정 자격 증명 공급자 솔루션을 설명합니다.

Note

대부분의 사용 사례에서 권장되는 옵션입니다. 특히 100개 이상의 Active Directory 그룹을 지원해야 하는 경우 사용자 지정 ID 제공업체 솔루션은 그룹 제한 없이 확장 가능한 대안을 제공합니다. 이 솔루션은 블로그 게시물 [Simplify Active Directory authentication with a custom identity provider for](#)에 설명되어 있습니다 [AWS Transfer Family](#).

- [Amazon API Gateway를 ID 제공자 통합에 사용](#)-이 주제에서는 AWS Lambda 함수를 사용하여 Amazon API Gateway 메서드를 지원하는 방법을 설명합니다.

단일 Amazon API Gateway 메서드를 사용하여 RESTful 인터페이스를 제공할 수 있습니다. Transfer Family는 이 메서드를 호출하여 자격 증명 공급자에 연결합니다. 자격 증명 공급자는 사용자가 Amazon S3 또는 Amazon EFS에 액세스할 수 있도록 인증하고 권한을 부여합니다. ID 제공업체를 통합하기 위해 RESTful API가 필요하거나 사용하여 지리 차단 또는 속도 제한 요청에 해당 기능을 AWS WAF 활용하려는 경우이 옵션을 사용합니다. 자세한 내용은 [Amazon API Gateway를 ID 제공자 통합에 사용](#)을 참조하세요.

- [동적 권한 관리 접근 방식](#)-이 주제에서는 세션 정책을 사용하여 사용자 권한을 동적으로 관리하는 접근 방식을 설명합니다.

사용자를 인증하려면, AWS Transfer Family를 사용하여 기존 ID 제공자를 사용할 수 있습니다. Amazon S3 또는 Amazon Elastic File System (Amazon EFS)에 액세스할 수 있도록 사용자를 인증하고 권한을 부여하는 AWS Lambda 함수를 사용하여 자격 증명 공급자를 통합합니다. 자세한 내용은 [AWS Lambda 를 사용하여 자격 증명 공급자 통합](#)을 참조하세요. 또한 AWS Transfer Family Management Console에서 전송된 파일 수 및 바이트 수와 같은 지표에 대한 CloudWatch 그래프에 액세스할 수 있으므로 중앙 집중식 대시보드를 사용하여 파일 전송을 모니터링할 수 있는 단일 창을 제공합니다.

- Transfer Family는 파일 전송 솔루션 구축을 안내하는 블로그 게시물과 워크숍을 제공합니다. 이 솔루션은 관리형 SFTP/FTPS 엔드포인트 AWS Transfer Family 에를 활용하고 사용자 관리를 위해 Amazon Cognito 및 DynamoDB를 활용합니다.

블로그 게시물은 [AWS Transfer Family 및 Amazon S3에서 자격 증명 공급자로 Amazon Cognito 사용에서 확인할 수 있습니다Amazon S3](#). 여기에서 워크숍에 대한 세부 정보를 볼 수 [있습니다](#).

Note

사용자 지정 자격 증명 공급자의 경우 사용자 이름은 최소 3자에서 최대 100자여야 합니다. 사용자 이름에 a~z, A~Z, 0~9, 밑줄 '_', 하이픈 '-', 마침표 '.' 및 at 기호 '@' 문자를 사용할 수 있습니다. 사용자 이름은 하이픈 '-', 마침표 '.' 또는 기호 '@'로 시작할 수 없습니다.

사용자 지정 자격 증명 공급자를 구현할 때는 다음 모범 사례를 고려하세요.

- Transfer Family 서버와 동일한 AWS 계정 및 리전에 솔루션을 배포합니다.
- IAM 역할 및 정책을 구성할 때 최소 권한 원칙을 구현합니다.
- 보안 강화를 위해 IP 허용 목록 및 표준화된 로깅과 같은 기능을 사용합니다.
- 배포하기 전에 비프로덕션 환경에서 사용자 지정 자격 증명 공급자를 철저히 테스트합니다.

주제

- [사용자 지정 자격 증명 공급자 솔루션](#)
- [AWS Lambda 를 사용하여 자격 증명 공급자 통합](#)
- [Amazon API Gateway를 ID 제공자 통합에 사용](#)
- [여러 인증 방법 사용](#)
- [사용자 지정 자격 증명 공급자에 대한 IPv6 지원](#)

사용자 지정 자격 증명 공급자 솔루션

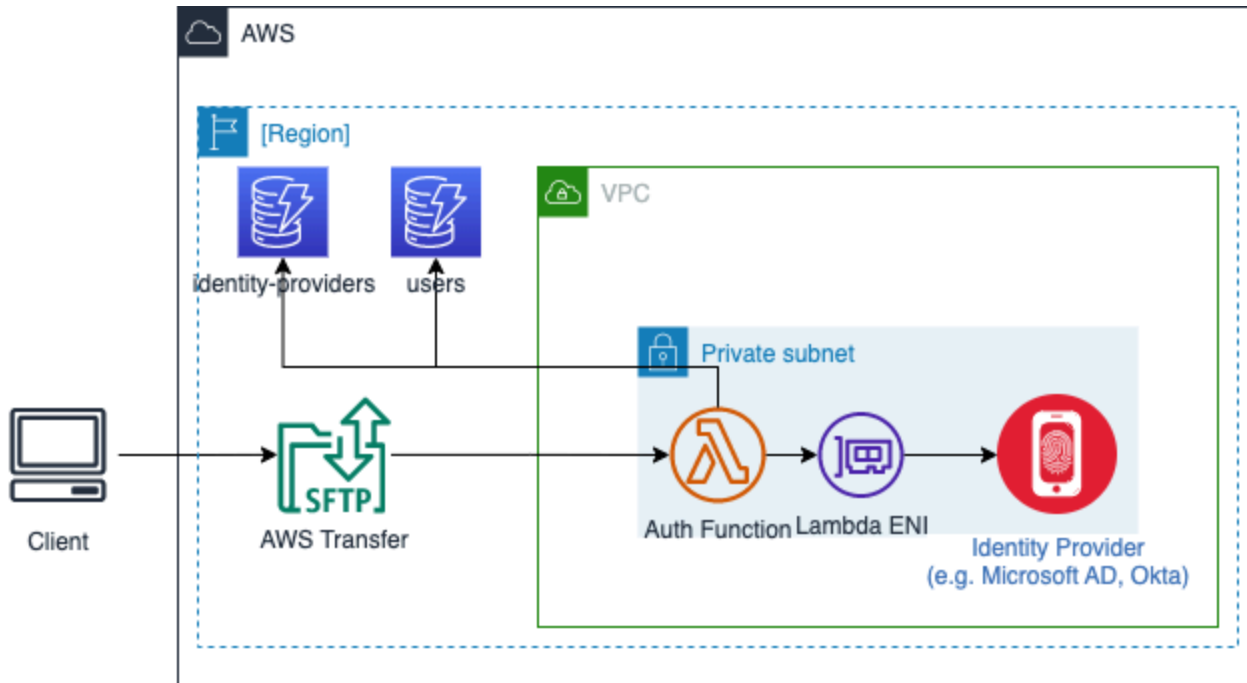
AWS Transfer Family 사용자 지정 ID 제공업체 솔루션은 기업이 서비스를 구현할 때 가지고 있는 많은 일반적인 인증 및 권한 부여 사용 사례를 해결하는 모듈식 사용자 지정 ID 제공업체 솔루션입니다. 이 솔루션은 세분화된 사용자별 세션 구성으로 사용자 지정 자격 증명 공급자를 구현하기 위한 재사용 가능한 기반을 제공하고 인증 및 권한 부여 로직을 분리하여 다양한 사용 사례에 유연하고 easy-to-maintain 기반을 제공합니다.

AWS Transfer Family 사용자 지정 ID 제공업체 솔루션을 사용하면 일반적인 엔터프라이즈 인증 및 권한 부여 사용 사례를 해결할 수 있습니다. 이 모듈식 솔루션은 다음을 제공합니다.

- 사용자 지정 자격 증명 공급자를 구현하기 위한 재사용 가능한 기반
- 세분화된 사용자별 세션 구성
- 별도의 인증 및 권한 부여 로직

사용자 지정 자격 증명 도구 키트의 구현 세부 정보

이 솔루션은 다양한 사용 사례를 위한 유연하고 유지 관리 가능한 기반을 제공합니다. 시작하려면 <https://github.com/aws-samples/toolkit-for-aws-transfer-family> 도구 키트를 검토한 다음 [시작하기](#) 섹션의 배포 지침을 따르세요.



Note

이전에 사용자 지정 자격 증명 공급자 템플릿과 예제를 사용한 적이 있는 경우 이 솔루션을 대신 채택하는 것이 좋습니다. 앞으로 공급자별 모듈은 이 솔루션을 표준화합니다. 이 솔루션에는 지속적인 유지 관리 및 기능 개선 사항이 적용됩니다.

이 솔루션에는 로깅을 포함한 세부 정보와 HomeDirectoryDetails 파라미터와 AWS Transfer Family가 필요한 추가 세션 메타데이터를 저장할 위치를 설명하는 사용자 지정 공급자를 구현하기 위한 표준 패턴이 포함되어 있습니다. 이 솔루션은 사용자별 세션 구성을 사용하여 사용자 지정 자격 증명 공급자를 구현하기 위한 재사용 가능한 기반을 제공하고, 인증을 완료하고 세션 설정을 설정하기

위해 Transfer Family에 반환되는 구성을 빌드하는 재사용 가능한 로직에서 자격 증명 공급자 인증 로직을 분리합니다.

이 솔루션의 코드 및 지원 리소스는 <https://github.com/aws-samples/toolkit-for-aws-transfer-family> 확인할 수 있습니다.

도구 키트에는 다음 기능이 포함되어 있습니다.

- 필요한 리소스를 프로비저닝하는 [AWS Serverless Application Model](#) 템플릿입니다. 선택적으로 블로그 게시물 웹 애플리케이션 방화벽 및 Amazon API Gateway로 보안에 설명된 AWS WAF대로 통합하도록 Amazon API Gateway를 배포하고 구성합니다. [AWS Transfer FamilyAWS Amazon API Gateway](#)
- , HomeDirectoryDetails Role 및와 같은 사용자 세션 설정을 포함하여 자격 증명 공급자에 대한 구성 메타데이터를 저장하는 [Amazon DynamoDB](#) 스키마입니다Policy.
- 향후 솔루션에 새 자격 증명 공급자를 모듈로 추가할 수 있는 모듈식 접근 방식입니다.
- 속성 검색: 선택적으로 AD, LDAP, Okta 등 지원되는 자격 증명 공급자로부터 IAM 역할 및 POSIX 프로파일(UID 및 GID) 속성을 검색합니다.
- 솔루션의 동일한 배포를 사용하여 단일 Transfer Family 서버와 여러 Transfer Family 서버에 연결된 여러 자격 증명 공급자를 지원합니다.
- 사용자 또는 자격 증명 공급자별로 선택적으로 구성할 수 있는 IP 허용 목록과 같은 기본 제공 IP 허용 목록 검사입니다.
- 문제 해결에 도움이 되는 구성 가능한 로그 수준 및 추적 지원이 포함된 세부 로깅.

사용자 지정 ID 제공업체 솔루션을 배포하기 전에 다음 AWS 리소스가 있어야 합니다.

- NAT 게이트웨이 또는 DynamoDB 게이트웨이 엔드포인트를 통해 인터넷에 연결되는 프라이빗 서브넷이 있는 Amazon Virtual Private Cloud(VPC)입니다.
- 다음 작업을 수행할 수 있는 적절한 IAM 권한:
 - custom-idp.yaml CloudFormation 템플릿을 배포합니다.
 - AWS CodePipeline 프로젝트 생성
 - AWS CodeBuild 프로젝트 생성
 - IAM 역할 및 정책 생성

⚠ Important

대상 Transfer Family 서버 AWS 리전 가 포함된 동일한 AWS 계정 및에 솔루션을 배포해야 합니다.

지원되는 ID 제공업체

다음 목록에는 사용자 지정 자격 증명 공급자 솔루션에 지원되는 자격 증명 공급자에 대한 세부 정보가 포함되어 있습니다.

제공업체	암호 흐름	퍼블릭 키 흐름	멀티 팩터	속성 검색	세부 정보
Active Directory 및 LDAP	예	예	아니요	예	사용자 확인은 퍼블릭 키 인증 흐름의 일부로 수행할 수 있습니다.
Argon2(로컬 해시)	예	아니요	아니요	아니요	Argon2 해시는 '로컬' 암호 기반 인증 사용 사례에 대한 사용자 레코드에 저장됩니다.
Amazon Cognito	예	아니요	예*	아니요	시간 기반 일회용 암호(TOTP) 기반 다중 인증만 해당됩니다. *SMS 기반 MFA는 지원

제공업체	암호 흐름	퍼블릭 키 흐름	멀티 팩터	속성 검색	세부 정보
					되지 않습니다.
Entra ID(이전 Azure AD)	예	아니요	아니요	아니요	
Okta	예	예	예*	예	TOTP 기반 MFA만 해당.
퍼블릭 키	아니요	예	아니요	아니요	퍼블릭 키는 DynamoDB의 사용자 레코드에 저장됩니다.
Secrets Manager	예	예	아니요	아니요	

AWS Lambda 를 사용하여 자격 증명 공급자 통합

이 주제에서는 사용자 지정 자격 증명 공급자에 연결하는 AWS Lambda 함수를 생성하는 방법을 설명합니다. Okta, Secrets Manager, OneLogin과 같은 사용자 지정 ID 공급자를 사용하거나 권한 부여 및 인증 로직이 포함된 사용자 지정 데이터 저장소를 사용할 수 있습니다.

대부분의 사용 사례에서 사용자 지정 자격 증명 공급자를 구성하는 권장 방법으론 사용하는 것입니다 [사용자 지정 자격 증명 공급자 솔루션](#).

Note

Lambda를 ID 공급자로 사용하는 Transfer Family 서버를 생성하기 전에 함수를 생성해야 합니다. Lambda 함수의 예는 [예 Lambda 함수](#)를 참조하세요. 또는 [Lambda 함수 템플릿](#) 중 하나를 사용하는 CloudFormation 스택을 배포할 수 있습니다. 또한 Lambda 함수가 Transfer Family를 신뢰하는 리소스 기반 정책을 사용하는지 확인하세요. 정책 예제는 [Lambda 리소스 기반 정책](#)을 참조하세요.

1. [AWS Transfer Family 콘솔](#)을 엽니다.
2. 서버 생성을 선택하여 서버 생성 페이지를 엽니다. ID 제공자 선택을 위해 다음 스크린샷과 같이 사용자 지정 ID 공급자를 선택합니다.

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

Note Either a valid password or valid private key will be required during user authentication

Cancel Previous **Next**

Note

인증 방법은 SFTP를 Transfer Family 서버의 프로토콜 중 하나로 활성화한 경우에만 사용할 수 있습니다.

3. 기본값인 자격 증명 공급자를 연결하는 AWS Lambda 데 사용이 선택되어 있는지 확인합니다.

4. AWS Lambda 함수에서 Lambda 함수의 이름을 선택합니다.
5. 나머지 상자를 채운 다음 서버 만들기를 선택합니다. 서버를 만들기 위한 나머지 단계에 대한 자세한 내용은 [SFTP, FTPS 또는 FTP 서버 엔드포인트 구성](#)을 참조하세요.

Lambda 리소스 기반 정책

Transfer Family 서버 및 Lambda ARN을 참조하는 정책이 있어야 합니다. 예를 들어 ID 공급자에 연결되는 Lambda 함수와 함께 다음 정책을 사용할 수 있습니다. 정책은 JSON을 문자열로 이스케이프합니다.

```
"Policy":
"{\"Version\":\"2012-10-17\",
  \"Id\":\"default\",
  \"Statement\":[
    {\"Sid\":\"AllowTransferInvocation\",
      \"Effect\":\"Allow\",
      \"Principal\":{\"Service\":\"transfer.amazonaws.com\"},
      \"Action\":\"lambda:InvokeFunction\",
      \"Resource\":\"arn:aws:lambda:region:123456789012:function:my-lambda-auth-function\",
      \"Condition\":{\"ArnLike\":{\"AWS:SourceArn\":
        \"arn:aws:transfer:region:123456789012:server/server-id\"}}}}
  ]}"
```

Note

상기 예 정책에서 각 ### ## ## ###를 자신의 정보로 바꿉니다.

이벤트 메시지 구조

사용자 지정 IDP에 대해 권한 부여자 Lambda 함수로 전송되는 SFTP 서버의 이벤트 메시지 구조는 다음과 같습니다.

```
{
  "username": "value",
  "password": "value",
  "protocol": "SFTP",
```

```

    "serverId": "s-abcd123456",
    "sourceIp": "192.168.0.100"
  }

```

서버로 전송되는 로그인 자격 증명의 값인 username과 password의 위치는 어디입니까?

예를 들어, 다음 명령을 입력해서 연결합니다.

```
sftp bobusa@server_hostname
```

새 암호를 두 번 입력하라는 메시지가 나타납니다.

```

Enter password:
mysecretpassword

```

Lambda 함수 내에서 전달된 이벤트를 인쇄하여 Lambda 함수에서 이를 확인할 수 있습니다. 그것은 다음 텍스트 블록과 비슷하게 보여야 합니다.

```

{
  "username": "bobusa",
  "password": "mysecretpassword",
  "protocol": "SFTP",
  "serverId": "s-abcd123456",
  "sourceIp": "192.168.0.100"
}

```

이벤트 구조는 FTP와 FTPS와 비슷한데, 유일한 차이점은 SFTP라기 보다 오히려 아닌 해당 값이 protocol파라미터에 사용된다는 것입니다.

인증을 위한 Lambda 함수

다양한 인증 전략을 구현하려면 Lambda 함수를 편집하세요. 애플리케이션의 요구 사항을 충족하는 데 도움이 되도록 CloudFormation 스택을 배포할 수 있습니다. 자세한 내용은 [개발자 안내서](#)나 [AWS Lambda Node.js로 Lambda 함수 구축](#)을 참조하세요.

주제

- [유효 Lambda 값](#)
- [예 Lambda 함수](#)

- [구성 테스트](#)
- [Lambda 함수 템플릿](#)

유효 Lambda 값

다음 표에는 사용자 지정 ID 공급자에 사용되는 Lambda 함수에 대해 Transfer Family가 허용하는 값에 대한 세부 정보가 설명되어 있습니다.

값	설명	필수
Role	Amazon S3 버킷 또는 Amazon EFS 파일 시스템에 대한 사용자 액세스를 제어하는 IAM 역할의 Amazon 리소스 이름 (ARN)을 지정합니다. 이 역할에 연결된 정책은 Amazon S3 버킷 또는 Amazon EFS 파일 시스템에 대한 파일 송수신 시 사용자에게 제공할 액세스의 수준을 결정합니다. 또한 IAM 역할에는 사용자의 전송 요청을 처리할 때 서버가 해당 리소스에 액세스할 수 있도록 허용하는 신뢰 관계가 포함되어야 합니다. 신뢰 관계 설정에 대한 자세한 내용은 신뢰 관계를 구축하기 위해를 참조하세요.	필수
PosixProfile	Amazon EFS 파일 시스템에 대한 사용자의 액세스를 제어하는 사용자 ID(Uid), 그룹 ID(Gid) 및 보조 그룹 ID(SecondaryGids)를 포함한 전체 POSIX 자격 증명입니다. 파일 시스템의 파일 및 디렉	Amazon EFS 백업 스토리지에 필요

값	설명	필수
	터리에 설정된 POSIX 권한에 따라 Amazon EFS 파일 시스템에서 파일을 송수신할 때 사용자에게 제공되는 액세스 수준이 결정됩니다.	
PublicKeys	이 사용자에게 유효한 SSH 퍼블릭 키 값 목록. 목록이 비어 있으면 유효한 로그인인 아님을 의미합니다. 암호 인증 중에는 반환할 수 없습니다.	선택 사항
Policy	여러 사용자에게 대해 동일한 IAM 역할을 사용할 수 있도록 한 사용자에게 대한 세션 정책입니다. 이 정책은 Amazon S3 버킷의 부분에 대한 사용자 액세스의 범위를 축소합니다. 사용자 지정 자격 증명 공급자와 함께 세션 정책을 사용하는 방법에 대한 자세한 내용은 이 주제의 세션 정책 예제를 참조하세요.	선택 사항

값	설명	필수
HomeDirectoryType	사용자가 서버에 로그인하는 경우 홈 디렉터리가 될 랜딩 디렉터리(폴더) 타입입니다. - PATH로 설정하면, 사용자는 파일 전송 프로토콜 클라이언트에서와 같이 절대 Amazon S3 버킷 또는 Amazon EFS 경로를 볼 수 있습니다. - LOGICAL로 설정하면 HomeDirectoryDetails 파라미터에서 Amazon S3 또는 Amazon EFS 경로를 사용자에게 시각적으로 표시할 방법에 대한 매핑을 제공해야 합니다.	선택 사항
HomeDirectoryDetails	사용자에게 표시할 Amazon S3 또는 Amazon EFS 경로 및 키와 이러한 경로 및 키를 사용자에게 시각적으로 표시할 방법을 지정하는 논리적 디렉터리 매핑입니다. Entry 및 Target 쌍을 지정해야 합니다. 여기서 Entry는 경로가 표시되는 방식을 보여주고 Target는 실제 Amazon S3 경로입니다.	HomeDirectoryType 이 LOGICAL의 값을 가진 경우 필요

값	설명	필수
HomeDirectory	클라이언트를 사용하여 서버에 로그인하는 경우 랜딩 디렉터리(폴더)입니다. 형식은 스토리지 백엔드에 따라 다릅니다. - Amazon S3의 경우: / bucket-name/user-home-directory 예시: /amzn-s3-demo-bucket/users/john - Amazon EFS의 경우: / fs-12345/user-home-directory 예시: /fs-faa1a123/users/john  Important 버킷 이름 또는 Amazon EFS 파일 시스템 ID가 경로에 포함되어야 합니다. 이 정보를 생략하면 파일 전송 중에 "파일을 찾을 수 없음" 오류가 발생합니다.	선택 사항

Note

HomeDirectoryDetails는 JSON 맵의 문자열 표현입니다. 이는 실제 JSON 맵 객체인 PosixProfile과 문자열로 구성된 JSON 배열인 PublicKeys과 대조적입니다. 언어별 세부 정보는 코드 예를 참조하세요.

⚠ HomeDirectory 형식 요구 사항

HomeDirectory 파라미터를 사용할 때는 전체 경로 형식을 포함해야 합니다.

- Amazon S3 스토리지의 경우: 항상 형식으로 버킷 이름 포함 /bucket-name/path
- Amazon EFS 스토리지의 경우: 파일 시스템 ID를 항상 형식으로 포함 /fs-12345/path

"파일을 찾을 수 없음" 오류의 일반적인 원인은 HomeDirectory 경로에서 버킷 이름 또는 EFS 파일 시스템 ID를 생략하는 것입니다. 스토리지 식별자 / 없이 HomeDirectory로 설정하면 인증은 성공하지만 파일 작업은 실패합니다.

예 Lambda 함수

이 섹션에서는 NodeJS와 Python 모두에서 사용할 수 있는 Lambda 함수 몇 가지를 소개합니다.

Note

이 예에서는 사용자, 역할, POSIX 프로필, 암호 및 홈 디렉토리 세부 정보가 모두 예시이므로 실제 값으로 바뀌어야 합니다.

Logical home directory, NodeJS

다음 NodeJS 예 함수는 [논리적 홈 디렉터리](#)가 있는 사용자에게 대한 세부 정보를 제공합니다.

```
// 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

다음 NodeJS 예 함수는 경로 기반의 홈 디렉터리가 있는 사용자에게 대한 세부 정보를 제공합니다.

```

// 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 format depends on your storage backend:
        // For S3: '/bucket-name/user-home-directory' (e.g., '/my-transfer-bucket/
users/john')
        // For EFS: '/fs-12345/user-home-directory' (e.g., '/fs-faa1a123/users/john')
        HomeDirectory: '/my-transfer-bucket/users/example-user' // S3 example -
replace with your bucket name
        // HomeDirectory: '/fs-faa1a123/users/example-user' // EFS example - uncomment
for EFS
    };

    // 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

다음 Python 예 함수는 [논리적 홈 디렉터리](#)가 있는 사용자에 대한 세부 정보를 제공합니다.

```
# 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

다음 Python 예 함수는 경로 기반 홈 디렉터리가 있는 사용자에 대한 세부 정보를 제공합니다.

```
# 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 format depends on your storage backend:
            # For S3: '/bucket-name/user-home-directory' (e.g., '/my-transfer-bucket/
            # users/john')
            # For EFS: '/fs-12345/user-home-directory' (e.g., '/fs-faa1a123/users/john')
            'HomeDirectory': '/my-transfer-bucket/users/example-user', # S3 example -
            # replace with your bucket name
            # 'HomeDirectory': '/fs-faa1a123/users/example-user', # EFS example -
            # uncomment for EFS
            '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

```

구성 테스트

사용자 지정 ID 공급자를 만든 후에는 구성을 테스트해야 합니다.

Console

AWS Transfer Family 콘솔을 사용하여 구성을 테스트하려면

1. [AWS Transfer Family 콘솔](#)을 엽니다.
2. 서버 페이지에서 새 서버를 선택하고 작업을 선택한 다음 테스트를 선택합니다.
3. CloudFormation 스택을 배포할 때 설정한 사용자 이름 및 암호에 텍스트를 입력합니다. 기본 옵션을 유지한 경우 사용자 이름은 myuser 이고 암호는 MySuperSecretPassword입니다.
4. 서버 프로토콜을 선택하고 CloudFormation 스택을 배포할 때 설정한 경우 소스 IP의 IP 주소를 입력합니다.

CLI

AWS CLI를 사용하여 구성을 테스트하려면

1. [test-identity-provider](#) 명령을 실행합니다. 다음 단계에서 설명하는 대로 *user input placeholder* 각각을 사용자 고유의 정보로 바꾸세요.

```

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. 서버 ID를 입력합니다.
3. CloudFormation 스택을 배포할 때 설정한 사용자 이름과 암호를 입력합니다. 기본 옵션을 유지한 경우 사용자 이름은 myuser 이고 암호는 MySuperSecretPassword입니다.
4. CloudFormation 스택을 배포할 때 설정한 경우 서버 프로토콜과 소스 IP 주소를 입력합니다.

사용자 인증에 성공하면 테스트 결과 StatusCode: 200 HTTP 응답, 빈 문자열 Message: "" (그렇지 않으면 실패 이유가 포함됨) 및 Response 필드가 반환됩니다.

Note

아래 응답 예에서 Response 필드는 '문자열화' (프로그램 내에서 사용할 수 있는 플랫폼 JSON 문자열로 변환) 된 JSON 객체이며, 사용자의 역할 및 권한에 대한 세부 정보를 포함합니다.

```
{
  "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\": \"\"
}
```

Lambda 함수 템플릿

인증에 Lambda 함수를 사용하는 CloudFormation 스택을 배포할 수 있습니다. 로그인 자격 증명을 사용하여 사용자를 인증하고 권한을 부여하는 여러 템플릿을 제공합니다. 이러한 템플릿 또는 AWS Lambda 코드를 수정하여 사용자 액세스를 추가로 사용자 지정할 수 있습니다.

Note

템플릿에 FIPS 지원 보안 정책을 지정 CloudFormation 하역을 통해 FIPS 지원 AWS Transfer Family 서버를 생성할 수 있습니다. 사용 가능한 보안 정책은 [AWS Transfer Family 서버에 대한 보안 정책](#)에 설명되어 있습니다.

인증에 사용할 CloudFormation 스택을 생성하려면

1. <https://console.aws.amazon.com/cloudformation> CloudFormation 콘솔을 엽니다.
2. AWS CloudFormation 사용 설명서의 CloudFormation 스택 템플릿 [선택에서 기존 템플릿의 스택](#) 배포 지침을 따릅니다.
3. 다음 템플릿 중 하나를 사용하여 Transfer Family의 인증에 사용할 Lambda 함수를 생성합니다.

- [클래식 \(Amazon Cognito\) 스택 템플릿](#)

에서 사용자 지정 자격 증명 공급자로 사용할 생성하기 AWS Lambda 위한 기본 템플릿입니다 AWS Transfer Family. 암호 기반 인증을 위해 Amazon Cognito에 대해 인증하며, 퍼블릭 키 기반 인증을 사용하는 경우 Amazon S3 버킷에서 퍼블릭 키가 반환됩니다. 배포 후에는 Lambda 함수 코드를 수정하여 다른 작업을 수행할 수 있습니다.

- [AWS Secrets Manager 스택 템플릿](#)

가 AWS Transfer Family 서버 AWS Lambda 와 함께 사용하여 Secrets Manager를 자격 증명 공급자로 통합하는 기본 템플릿입니다. 형식 AWS Secrets Manager 의에 있는 항목에 대해 인증합니다 `aws/transfer/server-id/username`. 또한 암호에는 Transfer Family에 반환된 모든 사용자 속성에 대한 카-값 쌍이 들어 있어야 합니다. 배포 후에는 Lambda 함수 코드를 수정하여 다른 작업을 수행할 수 있습니다.

- [Okta 스택 템플릿](#):가 AWS Transfer Family 서버 AWS Lambda 와 함께 사용하여 Okta를 사용자 지정 자격 증명 공급자로 통합하는 기본 템플릿입니다.
- [Okta-mfa 스택 템플릿](#):가 AWS Transfer Family 서버와 AWS Lambda 함께 사용하여 Okta를 다중 인증과 통합하는 기본 템플릿으로, 사용자 지정 자격 증명 공급자입니다.
- [Azure Active Directory 템플릿](#):이 스택에 대한 세부 정보는 블로그 게시물 [Authenticating to AWS Transfer Family with Azure Active Directory and AWS Lambda](#)에 설명되어 있습니다.

스택을 배포한 후에는 CloudFormation 콘솔의 출력 탭에서 스택에 대한 세부 정보를 볼 수 있습니다.

사용자 지정 ID 공급자를 Transfer Family 워크플로에 통합하는 가장 쉬운 방법은 이러한 스택 중 하나를 배포하는 것입니다.

Amazon API Gateway를 ID 제공자 통합에 사용

이 주제에서는 AWS Lambda 함수를 사용하여 API Gateway 메서드를 지원하는 방법을 설명합니다. ID 제공업체를 통합하기 위해 RESTful API가 필요하거나 사용하여 지리 차단 또는 속도 제한 요청에 대한 기능을 AWS WAF 활용하려는 경우 이 옵션을 사용합니다.

대부분의 사용 사례에서 사용자 지정 자격 증명 공급자를 구성하는 권장 방법론을 사용하는 것입니다 [사용자 지정 자격 증명 공급자 솔루션](#).

API Gateway를 사용하여 ID 공급자를 통합하는 경우의 제한

- 이 구성은 사용자 지정 도메인을 지원하지 않습니다.
- 이 구성은 프라이빗 API Gateway URL을 지원하지 않습니다.

둘 중 하나가 필요한 경우, API Gateway 없이 Lambda를 ID 공급자로서 사용할 수 있습니다. 자세한 내용은 [AWS Lambda를 사용하여 자격 증명 공급자 통합](#)을 참조하세요.

API Gateway 메서드를 사용하여 인증

Transfer Family의 ID 공급자로 사용할 API Gateway 메서드를 생성할 수 있습니다. 이 접근 방식은 API를 생성하고 제공할 수 있는 매우 안전한 방법을 제공합니다. API Gateway를 사용하면 모든 수신 API 작업이 더 안전하게 전송되도록 HTTPS 엔드포인트를 생성할 수 있습니다. API Gateway 서비스에 대한 자세한 내용은 [API Gateway 개발자 안내서](#)를 참조하세요.

API Gateway는 내부적으로 AWS를 사용하는 AWS Identity and Access Management (IAM)을 기반으로 동일한 인증을 AWS_IAM 제공하는 라는 권한 부여 방법을 제공합니다. 로 AWS_IAM 인증을 활성화 하면 API를 호출할 수 있는 명시적 권한을 가진 호출자만 해당 API의 API Gateway 메서드에 연결할 수 있습니다.

API Gateway 메서드를 Transfer Family의 사용자 지정 ID 공급자로 사용하려면 API Gateway 메서드에 IAM을 활성화해야 합니다. 이 프로세스의 일환으로 Transfer Family가 게이트웨이를 사용할 수 있는 권한을 IAM 역할에 제공합니다.

Note

보안을 강화하기 위해 웹 애플리케이션 방화벽을 구성할 수 있습니다. AWS WAF 은(는) Amazon API Gateway에 전달되는 HTTP 및 HTTPS 요청을 모니터링할 수 있게 해주는 웹 애플리케이션 방화벽입니다. 자세한 내용은 [웹 애플리케이션 방화벽 추가](#)을 참조하세요.

API Gateway 캐싱을 활성화하지 않음

Transfer Family의 사용자 지정 자격 증명 공급자로 API Gateway 메서드를 사용하는 경우 API Gateway 메서드에 대한 캐싱을 활성화하지 마십시오. 캐싱은 다음과 같은 이유로 인증 요청에 부적절하고 유효하지 않습니다.

- 각 인증 요청은 고유하며 캐시된 응답이 아닌 라이브 응답이 필요합니다.
- Transfer Family는 중복되거나 반복적인 요청을 API Gateway로 보내지 않으므로 캐싱은 이 점을 제공하지 않습니다.

- 캐싱을 활성화하면 API Gateway가 일치하지 않는 데이터로 응답하여 인증 요청에 유효하지 않은 응답이 발생합니다.

Transfer Family를 통한 사용자 지정 인증에 API Gateway 방법을 사용하려면

1. CloudFormation 스택을 생성합니다. 방법:

Note

스택 템플릿이 BASE64-encoded 암호를 사용하도록 업데이트되었습니다. 자세한 내용은 섹션을 참조하세요 [CloudFormation 템플릿 개선 사항](#).

- <https://console.aws.amazon.com/cloudformation> CloudFormation 콘솔을 엽니다.
- AWS CloudFormation 사용 설명서의 CloudFormation 스택 템플릿 [선택에서 기존 템플릿의 스택](#) 배포 지침을 따릅니다.
- 다음 기본 템플릿 중 하나를 사용하여 Transfer Family에서 사용자 지정 ID 공급자로 사용할 AWS Lambda지원 API Gateway 메서드를 만들 수 있습니다.

- [기본 스택 템플릿](#)

기본적으로 API Gateway 메서드는 하드 코딩된 SSH(Secure Shell) 키 또는 암호를 사용하여 단일 서버의 단일 사용자를 인증하는 사용자 지정 자격 증명 공급자로 사용됩니다. 배포 후에는 Lambda 함수 코드를 수정하여 다른 작업을 수행할 수 있습니다.

- [AWS Secrets Manager 스택 템플릿](#)

기본적으로 API Gateway 메서드는 Secrets Manager에 있는 다음 형식의 `aws/transfer/server-id/username` 항목에 대해 인증합니다. 또한 암호에는 Transfer Family에 반환된 모든 사용자 속성에 대한 키-값 쌍이 들어 있어야 합니다. 배포 후에는 Lambda 함수 코드를 수정하여 다른 작업을 수행할 수 있습니다. 자세한 내용은 블로그 게시물 [AWS Transfer Family 사용 시 암호 인증 활성화를 참조하세요 AWS Secrets Manager](#).

- [Okta 스택 템플릿](#)

API Gateway 메서드는 Transfer Family의 사용자 지정 ID 공급자로서 Okta와 통합됩니다. 자세한 내용은 블로그 게시물 [Okta를 AWS Transfer Family를 사용한 ID 공급자로서 사용](#)을 참조하세요.

사용자 지정 ID 공급자를 Transfer Family 워크플로에 통합하는 가장 쉬운 방법은 이러한 스택 중 하나를 배포하는 것입니다. 각 스택은 Lambda 함수를 사용하여 API Gateway에 기반한 API 메서드를 지원합니다. 그런 다음 Transfer Family에서 API 메서드를 사용자 지정 ID 공급자로 사용할 수 있습니다. 기본적으로 Lambda 함수는 암호로 myuser 호출된 단일 사용자를 인증합니다. MySuperSecretPassword 배포 후에는 보안 인증 정보를 편집하거나 Lambda 함수 코드를 업데이트하여 다른 작업을 수행할 수 있습니다.

Important

기본 사용자 및 암호 보안 인증 정보를 편집하는 것이 좋습니다.

스택을 배포한 후에는 CloudFormation 콘솔의 출력 탭에서 스택에 대한 세부 정보를 볼 수 있습니다. 이러한 세부 정보에는 스택의 Amazon 리소스 이름(ARN), 스택이 생성한 IAM 역할의 ARN, 새 게이트웨이의 URL이 포함됩니다.

Note

사용자 지정 ID 공급자 옵션을 사용하여 사용자를 위한 암호 기반 인증을 활성화하고 API Gateway에서 제공하는 요청 및 응답 로깅을 활성화하는 경우, API Gateway는 사용자의 암호를 Amazon CloudWatch Logs에 기록합니다. 프로덕션 환경에서는 이 로깅을 사용하지 않는 것이 좋습니다. 자세한 내용은 API 게이트웨이 개발자 안내서의 [Set up CloudWatch를 API 게이트웨이에 로그인 하도록 설정](#)을 참조하세요.

2. 서버의 API Gateway 메서드 구성을 확인합니다. 방법:

- a. <https://console.aws.amazon.com/apigateway/>에서 Amazon API Gateway 콘솔을 엽니다.
- b. 템플릿이 생성한 사용자 지정 자격 증명 공급자 기본 템플릿 전송 API를 선택합니다. CloudFormation 게이트웨이를 보려면 리전을 선택해야 할 수 있습니다.
- c. 리소스 창에서 GET을 선택합니다. 다음 스크린샷은 올바른 방법 설정을 보여줍니다.

Method request settings Edit

Authorization: AWS_IAM
 Request validator: None
 API key required: False
 SDK operation name: Generated based on method and path

Request paths (0) < 1 >

Name	Caching
No request paths defined	

URL query string parameters (2) < 1 >

Name	Required	Caching
protocol	False	Inactive
sourceip	False	Inactive

HTTP request headers (1) < 1 >

Name	Required	Caching
PasswordBase64	False	Inactive

Request body (0) < 1 >

Content type	Name
No request body defined.	

이 지점에서 API Gateway 를 배포할 준비가 완료됩니다.

3. 작업 및 API 배포를 선택합니다. 배포 단계에서 prod를 선택한 다음 배포를 선택합니다.

API Gateway 메서드가 성공적으로 배포되면 다음 스크린샷과 같이 스테이지 > 스테이지 세부 정보에서 성능을 확인합니다.

Note

화면 상단에 표시되는 URL 호출 주소를 복사합니다. 다음 단계에 필요할 수 있습니다.

API Gateway > APIs > Transfer Custom Identity Provider basic template API > Stages

Stages

Stage actions ▼ Create stage

prod

Stage details info

Stage name: prod

API cache: ☐ Inactive

Rate: 10000

Burst: 5000

Web ACL: -

Client certificate: -

Invoke URL: <https://execute-api.us-east-1.amazonaws.com/prod>

Active deployment: t8aqrm on December 12, 2023, 10:49 (UTC-05:00)

Logs and tracing info

CloudWatch logs: Error and info logs

Detailed metrics: ☐ Inactive

X-Ray tracing: ☐ Inactive

Custom access logging: ☐ Inactive

Stage variables (0/0)

Find resources

No variables

No variables associated with the stage.

Manage variables

4. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
5. 스택을 생성할 때 Transfer Family가 생성되었어야 합니다. 그렇지 않은 경우 다음 단계를 사용하여 서버를 구성합니다.
 - a. 서버 생성을 선택하여 서버 생성 페이지를 엽니다. ID 공급자 선택에서 사용자 지정을 선택한 다음 Amazon API Gateway를 선택하여 다음 스크린샷과 같이 ID 공급자에 연결합니다.

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

https://abode1234.execute-api.us-east-2.amazonaws.com/prod

Role
IAM role for the service to invoke your Amazon API Gateway URL

Stack-name-TransferIdentityProviderRole-ABC123DEF456GHI

- b. Amazon API Gateway URL 제공 텍스트 상자에 이 절차의 3단계에서 생성한 API Gateway 엔드포인트의 호출 URL 주소를 붙여넣습니다.
- c. 역할에서 CloudFormation 템플릿에 의해 생성된 IAM 역할을 선택합니다. 이 역할을 통해 Transfer Family는 API 게이트웨이 메서드를 간접 호출할 수 있습니다.

호출 역할에는 1단계에서 생성한 CloudFormation 스택에 대해 선택한 스택 이름이 포함됩니다. 형식은 다음과 같습니다. **CloudFormation-stack-name-TransferIdentityProviderRole-ABC123DEF456GHI**

- d. 나머지 상자를 채운 다음 서버 만들기를 선택합니다. 서버를 만들기 위한 나머지 단계에 대한 자세한 내용은 [SFTP, FTPS 또는 FTP 서버 엔드포인트 구성](#)을 참조하세요.

API Gateway 메서드 구현

Transfer Family를 위한 사용자 지정 ID 공급자를 생성하려면 API Gateway 메서드에서 리소스 경로가 `/servers/serverId/users/username/config` 인 단일 메서드를 구현해야 합니다.

serverId와 **username** 값은 RESTful 리소스 경로에서 얻습니다. 또한 다음 이미지에서 보여지듯이 메서드 요청의 같이 URL 쿼리 문자열 파라미터로서 **sourceIp** 및 **protocol**를 추가합니다.

The screenshot shows the AWS API Gateway console for a resource named `/servers/{serverId}/users/{username}/config` with the GET method. The resource is associated with the ARN `arn:aws:execute-api-east-1:1:*/GET/servers/{serverId}/users/{username}/config` and has a Resource ID of `aw4ihv`.

The **Method request settings** section shows:

- Authorization: `AWS_IAM`
- Request validator: `None`
- API key required: `False`
- SDK operation name: `Generated based on method and path`

The **Request paths (0)** section shows no request paths defined.

The **URL query string parameters (2)** section shows two parameters:

Name	Required	Caching
protocol	False	Inactive
sourceIp	False	Inactive

Note

이 사용자 이름은 최소 3글자에서 최대 100자여야 합니다. 사용자 이름에 `a~z`, `A~Z`, `0~9`, 밑줄 `_`, 하이픈 `-`, 마침표 `.` 및 `at` 기호 `@` 문자를 사용할 수 있습니다. 사용자 이름은 하이픈 `-`, 마침표 `.` 또는 기호 `@`로 시작할 수 없습니다.

Transfer Family가 사용자를 대신해 암호 인증을 시도하면, 서비스는 `Password:` 헤더 필드를 공급합니다. `Password:` 헤더가 없는 경우, Transfer Family는 퍼블릭 키 인증을 시도하여 사용자를 인증합니다.

ID 공급자를 사용하여 최종 사용자를 인증하고 권한을 부여하는 경우 자격 증명을 검증하는 것 외에도 최종 사용자가 사용하는 클라이언트의 IP 주소를 기반으로 액세스 요청을 허용하거나 거부할 수 있습니다. 이 기능을 사용하면 S3 버킷 또는 Amazon EFS 파일 시스템에 저장된 데이터가 지원되는 프로토

콜을 통해 신뢰할 수 있는 것으로 지정한 IP 주소에서만 액세스할 수 있습니다. 이 기능을 활성화하려면 `sourceIp`를 쿼리 문자열에 포함해야 합니다.

서버에 여러 프로토콜을 사용하도록 설정한 상태에서 여러 프로토콜에서 동일한 사용자 이름을 사용하여 액세스를 제공하려는 경우 ID 공급자에 각 프로토콜별 자격 증명이 설정되어 있으면 그렇게 할 수 있습니다. 이 기능을 활성화하려면 RESTful 리소스 경로에 `protocol` 값을 포함해야 합니다.

API Gateway 메서드는 항상 HTTP 상태 코드를 반환해야 200 합니다. 모든 다른 HTTP 상태 코드는 API 액세스하는데 오류를 나타냅니다.

Amazon S3 응답 예

예 응답 본문은 Amazon S3용 다음 형식의 JSON 문서입니다.

```
{
  "Role": "IAM role with configured S3 permissions",
  "PublicKeys": [
    "ssh-rsa public-key1",
    "ssh-rsa public-key2"
  ],
  "Policy": "STS Assume role session policy",
  "HomeDirectory": "/amzn-s3-demo-bucket/path/to/home/directory"
}
```

Note

정책은 JSON을 문자열로 이스케이프합니다. 예제:

```
"Policy":
"{
  \"Version\": \"2012-10-17\",
  \"Statement\":
  [
    {\"Condition\":
      {\"StringLike\":
        {\"s3:prefix\":
          [\"user/*\", \"user/\"]}},
      \"Resource\": \"arn:aws:s3:::amzn-s3-demo-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\"}]
}"

```

다음 예 응답은 사용자의 홈 디렉터리 타입이 논리적임을 보여줍니다.

```

{
  "Role": "arn:aws:iam::123456789012:role/transfer-access-role-s3",
  "HomeDirectoryType": "LOGICAL",
  "HomeDirectoryDetails": "[{\"Entry\": \"\\\", \"Target\": \"\\\"/amzn-s3-demo-bucket1\\\"}]\"",
  "PublicKeys": [""],
}

```

Amazon EFS 예 응답

예 응답 본문은 Amazon EFS용 다음 형식의 JSON 문서입니다.

```

{
  "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"
}

```

Role 필드는 인증에 성공했음을 나타냅니다. 암호 인증을 할 때(Password: 헤더를 제공할 때), SSH 퍼블릭 키를 제공할 필요가 없습니다. 사용자를 인증할 수 없는 경우(예: 암호가 잘못된 경우) 메서드는 Role 설정되지 않은 응답을 반환해야 합니다. 이러한 응답의 예로는 빈 JSON 객체를 들 수 있습니다.

다음 예 응답은 사용자가 논리적인 홈 디렉터리 타입을 가졌다는 것을 보여줍니다.

```
{
  "Role": "arn:aws:iam::123456789012:role/transfer-access-role-efs",
  "HomeDirectoryType": "LOGICAL",
  "HomeDirectoryDetails": "[{\"Entry\": \"\\\\\", \"Target\": \"\\\\/faa1a123\"}]",
  "PublicKeys": [""],
  "PosixProfile": { "Uid": 65534, "Gid": 65534 }
}
```

Lambda 함수에 JSON 형식으로 사용자 정책을 포함할 수 있습니다. Transfer Family의 사용자 정책 구성에 대한 자세한 내용은 단원을 참조하세요 [액세스 통제 관리](#).

기본 Lambda 함수

다양한 인증 전략을 구현하려면, 사용자의 게이트웨이 사용하는 Lambda 함수를 편집하세요. 애플리케이션의 요구 사항을 충족하는 데 도움이 되려면 Node.js에서 다음 예 Lambda 함수를 사용할 수 있습니다. 자세한 내용은 [개발자 안내서](#)나 [AWS Lambda Node.js로 Lambda 함수 구축](#)을 참조하세요.

다음 예 Lambda 함수는 사용자 이름, 암호 (암호 인증을 수행하는 경우), 서버 ID, 프로토콜 및 클라이언트 IP 주소를 가져옵니다. 이러한 입력을 조합하여 ID 공급자를 조회하고 로그인을 허용할지 여부를 결정할 수 있습니다.

 Note

서버에 여러 프로토콜을 사용하도록 설정한 상태에서 여러 프로토콜에서 동일한 사용자 이름을 사용하여 액세스를 제공하려는 경우 ID 공급자에 각 프로토콜별 보안 인증이 설정되어 있으면 그렇게 할 수 있습니다.

FTP (파일 전송 프로토콜)의 경우, Secure Shell(SSH) SFTP (파일 전송 프로토콜) 및 SSL 을 통한 파일 전송 프로토콜 (FTPS)의 분리된 보아 인증을 유지하는 것이 좋습니다. SFTP 및 FTPS와 달리 FTP는 자격 증명을 일반 텍스트로 전송하므로 FTP에 대해 별도의 보안 인증을 유지하는 것이 좋습니다. FTP 자격 증명을 SFTP 또는 FTPS에서 분리하면 FTP 자격 증명 이 공유되거나 노출되더라도 SFTP 또는 FTPS를 사용하는 워크로드의 보안을 유지할 수 있습 니다.

이 예 함수는 퍼블릭 키 인증을 수행하는 경우 퍼블릭 키와 함께 역할 및 논리적 홈 디렉터리 세부 정보를 반환합니다.

서비스 관리 사용자를 생성할 때는 해당 사용자의 홈 디렉터리를 논리적 또는 물리적 디렉터리로 설정합니다. 마찬가지로 원하는 사용자의 물리적 또는 논리적 디렉터리 구조를 전달하려면 Lambda 함수 결과가 필요합니다. 설정하는 파라미터는 [HomeDirectoryType](#) 필드 값에 따라 달라집니다.

- HomeDirectoryType을 PATH로 설정 — HomeDirectory 필드는 사용자에게 표시되는 절대 Amazon S3 버킷 접두사 또는 Amazon EFS 절대 경로여야 합니다.
- HomeDirectoryType을 LOGICAL로 설정 — HomeDirectory 필드를 설정하지 마세요. 대신 서비스 관리 사용자를 위한 [HomeDirectoryDetails](#) 매개 변수에 설명된 값과 유사하게 원하는 항목/대상 매핑을 제공하는 HomeDirectoryDetails 필드를 설정합니다.

예 함수는 [예 Lambda 함수](#)에 나열되어 있습니다.

와 함께 사용할 Lambda 함수 AWS Secrets Manager

를 자격 증명 공급자 AWS Secrets Manager 로 사용하려면 샘플 CloudFormation 템플릿에서 Lambda 함수로 작업할 수 있습니다. Lambda 함수는 사용자 자격 증명으로 Secrets Manager 서비스를 쿼리하고, 성공하면 지정된 시크릿을 반환합니다. Secrets Manager 사용에 대한 자세한 내용은 [AWS Secrets Manager 사용 설명서](#)를 참조하세요.

이 Lambda 함수를 사용하는 샘플 CloudFormation 템플릿을 다운로드하려면 [여기서 제공하는 Amazon S3 버킷 AWS Transfer Family](#)으로 이동합니다.

CloudFormation 템플릿 개선 사항

API Gateway 인터페이스가 게시된 CloudFormation 템플릿을 개선했습니다. 템플릿은 이제 API Gateway에서 BASE64-encoded 암호를 사용합니다. 기존 배포는 이러한 개선 사항 없이 계속 작동하지만 기본 US-ASCII 문자 집합을 벗어나는 문자가 있는 암호는 허용하지 않습니다.

이 기능을 활성화하는 템플릿의 변경 사항은 다음과 같습니다.

- GetUserConfigRequest AWS::ApiGateway::Method 리소스에 RequestTemplates 코드가 있어야 합니다(기울임꼴 줄은 업데이트된 줄임).

```
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')",
  }

```

- PasswordBase64 헤더를 사용하려면 GetUserConfig 리소스RequestParameters의가 변경되어야 합니다(기울임꼴 줄은 업데이트된 줄임).

```

RequestParameters:
  method.request.header.PasswordBase64: false
  method.request.querystring.protocol: false
  method.request.querystring.sourceIp: false

```

스택의 템플릿이 최신인지 확인하려면

1. <https://console.aws.amazon.com/cloudformation> CloudFormation 콘솔을 엽니다.
2. 스택 목록에서 스택을 선택합니다.
3. 세부 정보 패널에서 템플릿 탭을 선택합니다.
4. 다음을 찾습니다.

- 를 검색RequestTemplates하고 다음 줄이 있는지 확인합니다.

```

"password":
  "$util.escapeJavaScript($util.base64Decode($input.params('PasswordBase64'))).replaceAll(
    \",\", \"'\", \"'\")",

```

- 를 검색RequestParameters하고 다음 줄이 있는지 확인합니다.

```

method.request.header.PasswordBase64: false

```

업데이트된 줄이 표시되지 않으면 스택을 편집합니다. CloudFormation 스택을 업데이트하는 방법에 대한 자세한 내용은 AWS CloudFormation, 사용 설명서의 [스택 템플릿 수정](#)을 참조하세요.

여러 인증 방법 사용

Transfer Family 서버는 여러 인증 방법을 사용할 때 AND 로직을 제어합니다. Transfer Family는 이를 사용자 지정 자격 증명 공급자에 대한 두 개의 개별 요청으로 취급하지만 그 효과가 결합됩니다.

인증을 완료하려면 두 요청 모두 올바른 응답으로 성공적으로 반환되어야 합니다. Transfer Family는 두 개의 응답을 완료해야 합니다. 즉, Amazon EFS를 스토리지로 사용하는 경우 필요한 모든 요소(역

할, 홈 디렉터리, 정책 및 POSIX 프로파일)가 포함됩니다. 또한 Transfer Family는 암호 응답에 퍼블릭 키를 포함해서는 안 됩니다.

퍼블릭 키 요청에는 자격 증명 공급자와 별도의 응답이 있어야 합니다. 암호 또는 키 또는 암호 및 키를 사용할 때는이 동작이 변경되지 않습니다.

SSH/SFTP 프로토콜은 먼저 퍼블릭 키 인증을 사용하여 소프트웨어 클라이언트에 챌린지한 다음 암호 인증을 요청합니다. 이 작업을 수행하려면 사용자가 인증을 완료하기 전에 두 작업이 모두 성공해야 합니다.

사용자 지정 자격 증명 공급자 옵션의 경우 인증 방법에 대해 다음 옵션 중 하나를 지정할 수 있습니다.

- 암호 또는 키 - 사용자는 암호 또는 키로 인증할 수 있습니다. 이것이 기본값입니다.
- 암호만 - 연결하려면 사용자가 암호를 제공해야 합니다.
- 키만 - 연결하려면 사용자가 프라이빗 키를 제공해야 합니다.
- 암호 및 키 - 사용자가 연결하려면 프라이빗 키와 암호를 모두 제공해야 합니다. 서버가 먼저 키를 확인한 다음 키가 유효하면 암호를 입력하라는 메시지가 표시됩니다. 제공된 프라이빗 키가 저장된 퍼블릭 키와 일치하지 않는 경우 인증이 실패합니다.

사용자 지정 자격 증명 공급자에 대한 IPv6 지원

AWS Transfer Family 사용자 지정 자격 증명 공급자는 IPv6 연결을 완벽하게 지원합니다. 사용자 지정 자격 증명 공급자를 구현할 때 Lambda 함수는 추가 구성 없이 IPv4 및 IPv6 클라이언트 모두에서 인증 요청을 수신하고 처리할 수 있습니다. Lambda 함수는 요청의 `sourceIp` 필드에서 클라이언트의 IP 주소를 수신하며, IPv4 주소(예: 203.0.113.42) 또는 IPv6 주소(예:)일 수 있습니다. 2001:db8:85a3:8d3:1319:8a2e:370:7348. 사용자 지정 자격 증명 공급자 구현은 두 주소 형식을 모두 적절하게 처리해야 합니다.

Important

사용자 지정 자격 증명 공급자가 IP 기반 검증 또는 로깅을 수행하는 경우 구현이 IPv6 주소 형식을 올바르게 처리하는지 확인합니다. IPv6 주소는 IPv4 주소보다 길며 다른 표기법 형식을 사용합니다.

Note

사용자 지정 자격 증명 공급자에서 IPv6 주소를 처리할 때는 간단한 문자열 비교 대신 적절한 IPv6 주소 구문 분석 함수를 사용해야 합니다. IPv6 주소는 다양한 정식 형식(예: fd00:b600::ec2 또는 fd00:b600:0:0:0:0:0:ec2)으로 표현할 수 있습니다. 구현 언어에서 적절한 IPv6 주소 라이브러리 또는 함수를 사용하여 IPv6 주소를 올바르게 검증하고 비교합니다.

Example 사용자 지정 자격 증명 공급자에서 IPv4 및 IPv6 주소 모두 처리

```
def lambda_handler(event, context):
    # Extract the source IP address from the request
    source_ip = event.get('sourceIp', '')

    # Log the client IP address (works for both IPv4 and IPv6)
    print(f"Authentication request from: {source_ip}")

    # Example of IP-based validation that works with both IPv4 and IPv6
    if is_ip_allowed(source_ip):
        # Continue with authentication
        # ...
    else:
        # Reject the authentication request
        return {
            "Role": "",
            "HomeDirectory": "",
            "Status": "DENIED"
        }
```

사용자 지정 자격 증명 공급자 구현에 대한 자세한 내용은 섹션을 참조하세요 [AWS Lambda 를 사용하여 자격 증명 공급자 통합](#).

Microsoft Active Directory용 AWS Directory Service 사용

AWS Transfer Family 를 사용하여 파일을 전송할 때 최종 사용자를 인증할 수 있습니다 AWS Directory Service for Microsoft Active Directory. 최종 사용자의 자격 증명을 변경하거나 사용자 지정 권한 부여자를 사용하지 않고도 Active Directory 인증을 사용하는 파일 전송 워크플로를 원활하게 마 이그레이션할 수 있습니다.

를 사용하면 Amazon Simple Storage Service(Amazon S3 EFS)에 저장된 데이터에 대해 SFTP, FTPS 및 FTP를 통해 Directory Service 사용자 및 그룹에 액세스 권한을 안전하게 제공할 AWS Managed Microsoft AD가 있습니다. Amazon Elastic File System Active Directory를 사용하여 사용자의 자격 증명을 저장하면 이제 이러한 사용자에게 대한 파일 전송을 더 쉽게 활성화할 수 있습니다.

Active Directory 커넥터를 사용하여 온프레미스 환경 또는 AWS 클라우드 AWS Managed Microsoft AD 의에서 Active Directory 그룹에 대한 액세스를 제공할 수 있습니다. AWS 클라우드 또는 온프레미스 네트워크에서 Microsoft Windows 환경에 이미 구성된 사용자에게가 자격 증명에 사용하는 AWS Transfer Family 서버에 AWS Managed Microsoft AD 대한 액세스 권한을 부여할 수 있습니다. AWS 스토리지 블로그에는 Transfer Family: [Simplify Active Directory authentication with a custom identity provider for](#)에서 Active Directory를 사용하기 위한 솔루션을 자세히 설명하는 게시물이 포함되어 있습니다 [AWS Transfer Family](#).

Note

- AWS Transfer Family 는 Simple AD를 지원하지 않습니다.
- Transfer Family는 리전 간 Active Directory 구성을 지원하지 않습니다. Transfer Family 서버와 동일한 리전에 있는 Active Directory 통합만 지원합니다.
- Transfer Family는 AWS Managed Microsoft AD 또는 AD 커넥터를 사용하여 기존 RADIUS 기반 MFA 인프라에 대한 다중 인증(MFA)을 활성화하는 것을 지원하지 않습니다.
- AWS Transfer Family 는 Managed Active Directory의 복제된 리전을 지원하지 않습니다.

를 사용하려면 다음 단계를 수행해야 AWS Managed Microsoft AD합니다.

1. Directory Service 콘솔을 사용하여 하나 이상의 AWS Managed Microsoft AD 디렉터리를 생성합니다.
2. Transfer Family 콘솔을 사용하여 자격 증명 공급자 AWS Managed Microsoft AD 로 사용하는 서버를 생성합니다.
3. Active AWS Directory 커넥터를 사용하여 디렉터리를 설정합니다.
4. 하나 이상의 Directory Service 그룹에서 액세스를 추가합니다.
5. 필수 사항은 아니지만 사용자 액세스를 테스트하고 확인하는 것이 좋습니다.

주제

- [사용을 시작하기 전에 AWS Directory Service for Microsoft Active Directory](#)

- [액티브 디렉터리 영역 사용](#)
- [자격 증명 공급자 AWS Managed Microsoft AD 로 선택](#)
- [온프레미스 Microsoft Active Directory에 연결](#)
- [그룹에 액세스 권한 부여](#)
- [테스트 사용자](#)
- [그룹의 서버 액세스 삭제](#)
- [SSH\(보안 셸\)를 사용하여 서버에 연결](#)
- [포리스트 및 트러스트를 사용하여 자체 관리형 Active Directory AWS Transfer Family 에 연결](#)

사용을 시작하기 전에 AWS Directory Service for Microsoft Active Directory

Note

AWS Transfer Family 의 기본 제한은 서버당 Active Directory 그룹 100개입니다. 사용 사례에 100개 이상의 그룹이 필요한 경우 용 사용자 지정 자격 증명 공급자를 [사용한 Active Directory 인증 간소화에 설명된 대로 사용자 지정 자격 증명 공급자 AWS Transfer Family](#) 솔루션을 사용하는 것이 좋습니다.

AD 그룹의 고유 식별자를 입력합니다.

를 사용하려면 먼저 Microsoft AD 디렉터리의 각 그룹에 고유한 식별자를 제공해야 AWS Managed Microsoft AD합니다. 각 그룹의 SID(보안 식별자)를 사용하여 이 작업을 수행할 수 있습니다. 연결한 그룹의 사용자는 AWS Transfer Family를 사용하여 활성화된 프로토콜을 통해 Amazon S3 또는 Amazon EFS 리소스에 액세스할 수 있습니다.

다음 Windows PowerShell 명령을 사용하여 그룹의 SID를 검색하고 *YourGroupName*을 그룹 이름으로 바꿉니다.

```
Get-ADGroup -Filter {samAccountName -like "YourGroupName*"} -Properties * | Select SamAccountName, ObjectSid
```

Note

를 자격 증명 공급자 AWS Directory Service 로 사용하고 있고 및 값이 다른 경우 userPrincipalNameSamAccountName는의 값을 AWS Transfer Family 수락합니

다SamAccountName. Transfer Family는 userPrincipalName에 지정된 값을 수락하지 않습니다.

역할에 Directory Service 권한 추가

를 자격 증명 공급자 AWS Directory Service 로 사용하려면 Directory Service API 권한도 필요합니다. 다음 권한이 필요하거나 제안됩니다.

- Transfer Family에서 디렉토리를 조회하려면 ds:DescribeDirectories가 필요합니다.
- Transfer Family에 대한 승인을 추가하려면 ds:AuthorizeApplication가 필요합니다.
- ds:UnauthorizeApplication에서는 서버 생성 프로세스 중에 문제가 발생할 경우를 대비하여 임시로 생성된 모든 리소스를 제거하는 것이 좋습니다.

Transfer Family 서버를 만드는 데 사용하는 역할에 이러한 권한을 추가하세요. 이러한 권한에 대한 자세한 내용은 [Directory Service API 권한: 작업, 리소스 및 조건 참조](#)를 참조하세요.

액티브 디렉터리 영역 사용

Active Directory 사용자가 AWS Transfer Family 서버에 액세스하도록 하는 방법을 고려할 때는 사용자의 영역과 해당 그룹의 영역을 염두에 두세요. 가장 바람직한 것은 사용자 영역과 그룹 영역이 일치해야 하는 것입니다. 즉, 사용자와 그룹이 모두 기본 영역에 있거나 둘 다 신뢰할 수 있는 영역에 속해 있다는 뜻입니다. 그렇지 않은 경우 Transfer Family에서 사용자를 인증할 수 없습니다.

사용자를 테스트하여 구성이 올바른지 확인할 수 있습니다. 자세한 내용은 [테스트 사용자](#)를 참조하세요. 사용자/그룹 영역에 문제가 있는 경우 사용자 그룹에 연결된 액세스 권한을 찾을 수 없습니다라는 오류 메시지가 표시됩니다.

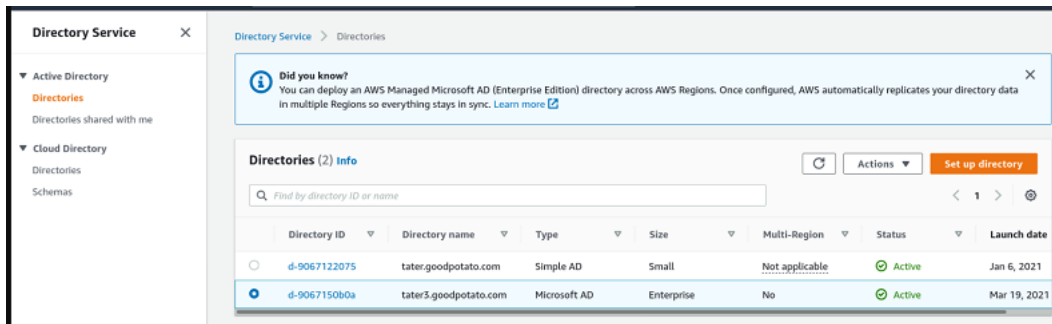
자격 증명 공급자 AWS Managed Microsoft AD 로 선택

이 섹션에서는를 서버와 AWS Directory Service for Microsoft Active Directory 함께 사용하는 방법을 설명합니다.

Transfer Family AWS Managed Microsoft AD 와 함께를 사용하려면

1. 에 로그인 AWS Management Console 하고 <https://console.aws.amazon.com/directoryservicev2/> Directory Service 콘솔을 엽니다.

Directory Service 콘솔을 사용하여 하나 이상의 관리형 디렉터리를 구성합니다. 자세한 내용을 알아보려면 Directory Service 관리 안내서의 [AWS Managed Microsoft AD](#) 단원을 참조하세요.

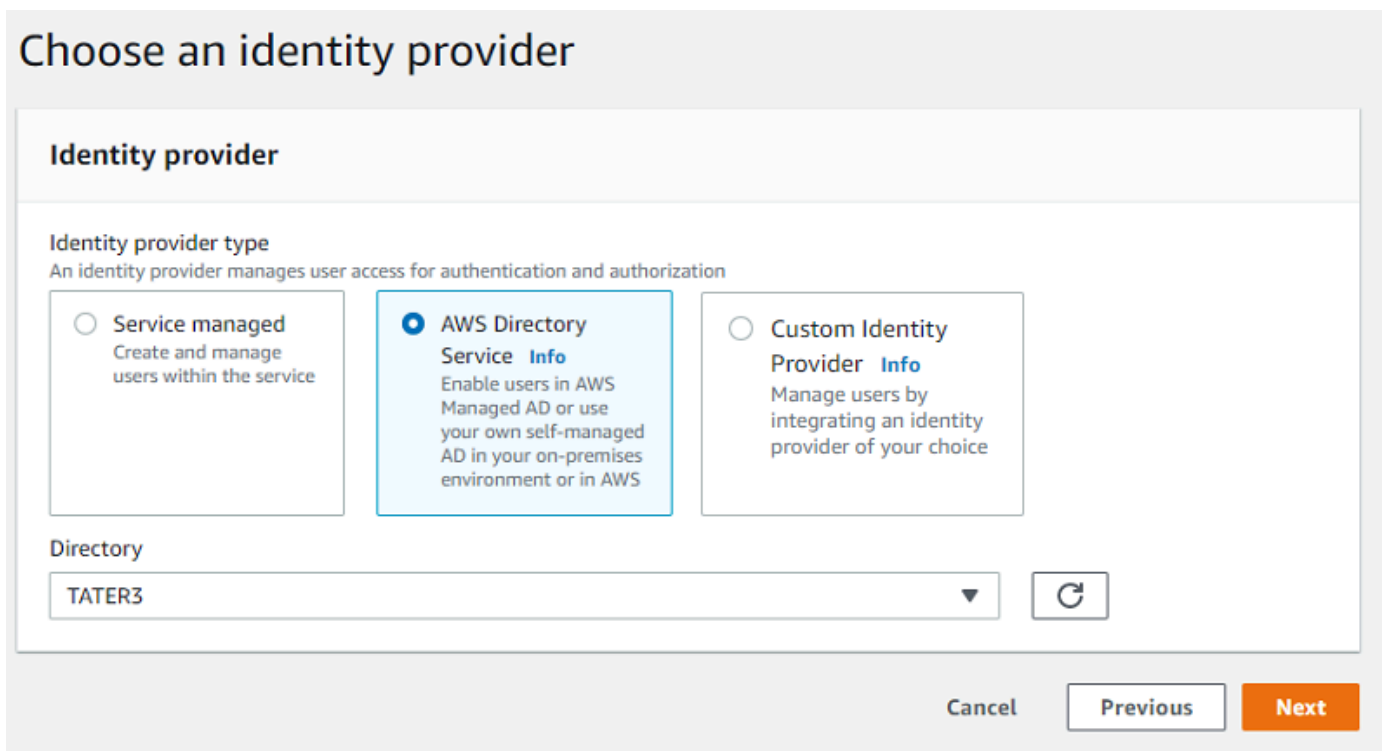


2. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 열고 서버 생성을 선택합니다.
3. 프로토콜 선택 페이지의 목록에서 하나 이상의 프로토콜을 선택합니다.

Note

FTPS를 선택하는 경우 AWS Certificate Manager 인증서를 제공해야 합니다.

4. 자격 증명 제공자 선택에서 AWS Directory Service를 선택합니다.



5. 디렉터리 목록에는 구성된 모든 관리 디렉터리가 포함됩니다. 목록에서 디렉터리를 선택하고 다음을 선택합니다.

 Note

- 교차 계정 및 공유 디렉터리는에서 지원되지 않습니다 AWS Managed Microsoft AD.
- Directory Service를 자격 증명 공급자로 사용하여 서버를 설정하려면 몇 가지 Directory Service 권한을 추가해야 합니다. 자세한 내용은 [사용을 시작하기 전에 AWS Directory Service for Microsoft Active Directory](#)을 참조하세요.

6. 서버 생성을 완료하려면 다음 절차 중 하나를 사용합니다.

- [SFTP 지원 서버 생성](#)
- [FTPS 지원 서버 생성](#)
- [FTP 지원 서버 생성](#)

해당 절차에서 ID 제공자 선택 다음 단계를 계속 진행하세요.

 Important

Transfer Family 서버에서 Microsoft AD 디렉터리를 사용한 Directory Service 경우에서 삭제할 수 없습니다. 먼저 서버를 삭제해야 디렉터리를 삭제할 수 있습니다.

온프레미스 Microsoft Active Directory에 연결

이 섹션에서는 AD 커넥터를 사용하여 AWS 디렉터리를 설정하는 방법을 설명합니다.

AD Connector를 사용하여 AWS 디렉터리를 설정하려면

1. [디렉터리 서비스](#) 콘솔을 열고 디렉터리를 선택합니다.
2. 디렉터리 설정을 선택합니다.
3. 디렉터리 타입으로는 AD Connector를 선택합니다.
4. 디렉터리 크기를 선택하고 다음을 선택한 다음 VPC와 서브넷을 선택합니다.
5. 다음을 선택하고 다음과 같이 필드를 채웁니다.
 - 디렉터리 DNS 이름: Microsoft Active Directory에 사용 중인 도메인 이름을 입력합니다.
 - DNS IP 주소: Microsoft Active Directory IP 주소를 입력합니다.

- 서버 계정 사용자 이름 및 암호: 사용할 서비스 계정의 세부 정보를 입력합니다.

6. 화면을 완료하여 디렉터리 서비스를 생성합니다.

다음 단계는 SFTP 프로토콜과 AWS Directory Service의 자격 증명 공급자 유형을 사용하여 Transfer Family 서버를 생성하는 것입니다. 디렉터리 드롭다운 목록에서 이전 절차에서 추가한 디렉터리를 선택합니다.

그룹에 액세스 권한 부여

서버를 생성한 후에는를 사용하여 활성화된 프로토콜을 통해 파일을 업로드하고 다운로드할 수 있는 액세스 권한이 있는 디렉터리의 그룹을 선택해야 합니다 AWS Transfer Family. 액세스 권한을 생성하여 이 작업을 수행할 수 있습니다.

Note

AWS Transfer Family 의 기본 제한은 서버당 Active Directory 그룹 100개입니다. 사용 사례에 100개 이상의 그룹이 필요한 경우 용 사용자 지정 자격 증명 공급자를 [사용한 Active Directory 인증 간소화에 설명된 대로 사용자 지정 자격 증명 공급자 AWS Transfer Family](#) 솔루션을 사용하는 것이 좋습니다.

Note

사용자는 액세스 권한을 부여하는 그룹에 직접 속해야 합니다. 예를 들어 Bob이 사용자이고 groupA에 속하며 groupA 자체가 groupB에 포함되어 있다고 가정합니다.

- 그룹 A에 액세스 권한을 부여하면 Bob에게 액세스 권한이 부여됩니다.
- 그룹 A가 아닌 그룹 B에 액세스 권한을 부여하는 경우 Bob은 액세스 권한을 갖지 않습니다.

그룹에 액세스 권한을 부여하려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 서버 세부 정보 페이지로 이동합니다.
3. 액세스 섹션에서 액세스 추가를 선택합니다.
4. 이 서버에 액세스할 AWS Managed Microsoft AD 디렉터리의 SID를 입력합니다.

Note

그룹의 SID를 찾는 방법에 대한 자세한 내용은 [the section called “사용을 시작하기 전에 AWS Directory Service for Microsoft Active Directory”](#)를 참조하세요.

5. 액세스에서 그룹의 AWS Identity and Access Management (IAM) 역할을 선택합니다.
6. 정책 섹션에서 정책을 선택합니다. 기본 설정은 없음입니다.
7. 홈 디렉터리에서 그룹의 홈 디렉터리에 해당하는 Amazon S3 버킷을 선택합니다.

Note

세션 정책을 생성하여 사용자에게 표시되는 버킷 부분을 제한할 수 있습니다. 예를 들어, /filetest 디렉터리 아래에 있는 자신의 폴더로만 사용자를 제한하려면 상자에 다음 텍스트를 입력합니다.

```
/filetest/${transfer:UserName}
```

세션 정책 생성에 대한 자세한 설명은 [Amazon S3 버킷을 위한 세션 정책 생성](#) 섹션을 참조하세요.

8. 추가를 선택하여 연결을 생성합니다.
9. 서버를 선택합니다.
10. 액세스 추가를 선택합니다.
 - 그룹의 SID를 입력합니다.

Note

SID를 찾는 방법에 대한 자세한 내용은 [the section called “사용을 시작하기 전에 AWS Directory Service for Microsoft Active Directory”](#)를 참조하세요.

11. 액세스 추가를 선택합니다.

액세스 섹션에는 서버에 대한 액세스가 나열됩니다.

The screenshot displays the AWS Transfer Family console interface. At the top, the 'Endpoint configuration' section shows the 'Availability Zone' as 'us-east-1a', 'Subnet ID' as 'subnet-...', and 'Private IPv4 Address' as '172.31.80.36'. Below this, the 'Accesses (1)' section features a search bar, an 'Actions' dropdown menu, and an 'Associate access' button. A table lists the access details with columns for 'External Id', 'Home directory', and 'Role'. The first entry shows 'S...', '/padbucket3', and 'ADGuy_S3_And_EFS'. The 'Additional details' section at the bottom includes 'Logging role' (Server activity not logged to Amazon CloudWatch), 'Server host key', 'Security Policy' (TransferSecurityPolicy-2018-11), and 'Domain' (Amazon S3). An 'Edit' button is located in the top right of this section.

테스트 사용자

사용자가 서버의 AWS Managed Microsoft AD 디렉터리에 액세스할 수 있는지 테스트할 수 있습니다.

Note

사용자는 엔드포인트 구성 페이지의 액세스 섹션에 나열된 정확히 하나의 그룹(외부 ID)에 속해야 합니다. 사용자가 그룹에 속하지 않거나 둘 이상의 그룹에 속해 있는 경우 해당 사용자에게 액세스 권한이 부여되지 않습니다.

특정 사용자에게 액세스 권한이 있는지 테스트하려면

1. 서버 세부 정보 페이지에서 작업을 선택한 다음 테스트를 선택합니다.
2. 자격 증명 제공자 테스트의 경우 액세스 권한이 있는 그룹 중 하나에 속하는 사용자의 로그인 자격 증명을 입력합니다.
3. 테스트를 선택합니다.

ID 제공자 테스트에 성공하면 선택한 사용자에게 서버 액세스 권한이 부여되었음을 알 수 있습니다.

Identity provider testing

User configuration [Info](#)

Username

transferuser1

Password

Response

```
{
  "Response": "
  {\"homeDirectory\":\"/padbucket3\", \"homeDirectoryDetails\":null, \"homeDirectoryType\":\"PATH\", \"posixProfile\",
  \"null\", \"publicKeys\":null, \"role\":\"arn:aws:iam::195886157073:role/ADGuy_53_Ard_EF5\", \"policy\":\"null\", \"userName\": \"transferuser1\", \"identityProviderType\":null, \"userConfigMessage\":null}",
  "StatusCode": 200,
  "Message": ""
}
```

Cancel

Test

사용자가 액세스 권한이 있는 두 개 이상의 그룹에 속해 있는 경우 다음과 같은 응답을 받게 됩니다.

```
"Response": "",
"StatusCode": 200,
"Message": "More than one associated access found for user's groups."
```

그룹의 서버 액세스 삭제

그룹의 서버 액세스 삭제

1. 서버 세부 정보 페이지에서 작업을 선택한 다음 액세스 삭제를 선택합니다.
2. 대화 상자에서 이 그룹에 대한 액세스 권한을 제거합니다.

서버 세부 정보 페이지로 돌아가면 이 그룹에 대한 액세스 권한이 더 이상 나열되지 않는 것을 볼 수 있습니다.

SSH(보안 셸)를 사용하여 서버에 연결

서버와 사용자를 구성한 후 SSH를 사용하여 서버에 연결하고 액세스 권한이 있는 사용자의 정식 사용자 이름을 사용할 수 있습니다.

```
sftp user@active-directory-domain@vpc-endpoint
```

예를 들어 `transferuserexample@mycompany.com@vpce-0123456abcdef-789xyz.vpc-svc-987654zyxabc.us-east-1.vpce.amazonaws.com`입니다.

이 형식은 페더레이션 검색을 대상으로 하므로 크기가 클 수 있는 Active Directory의 검색을 제한합니다.

Note

간단한 사용자 이름을 지정할 수 있습니다. 하지만 이 경우 Active Directory 코드가 페더레이션의 모든 디렉터리를 검색해야 합니다. 이로 인해 검색이 제한될 수 있으며 사용자에게 액세스 권한이 있어야 하는 경우에도 인증이 실패할 수 있습니다.

인증이 완료되면 사용자는 사용자를 구성할 때 지정한 홈 디렉터리에 위치해 있습니다.

포리스트 및 트러스트를 사용하여 자체 관리형 Active Directory AWS Transfer Family에 연결

Directory Service에는 자체 관리형 Active Directory에 연결하는 데 사용할 수 있는 다음 옵션이 있습니다.

- 단방향 포리스트 신뢰(온프레미스 Active Directory에서 송 AWS Managed Microsoft AD 수신)는 루트 도메인에서만 작동합니다.
- 하위 도메인의 경우 다음 중 하나를 사용할 수 있습니다.
 - AWS Managed Microsoft AD와 온프레미스 Active Directory 간의 양방향 신뢰 사용
 - 각 하위 도메인에는 단방향 외부 트러스트를 사용합니다.

예를 들어 `transferuserexample@mycompany.com` 신뢰할 수 있는 도메인을 사용하여 서버에 연결하는 경우 사용자는 신뢰할 수 있는 도메인을 지정해야 합니다.

Entra ID 도메인 서비스에 AWS 디렉터리 서비스 사용

SFTP 전송만 필요하고 도메인을 관리하지 않으려는 고객을 위한 Simple Active Directory가 있습니다. 또는 완전 관리형 서비스에서 Active Directory와 고가용성의 이점을 누리고자 하는 고객은 AWS 관리형 Microsoft AD를 사용할 수 있습니다. 마지막으로, SFTP 전송을 위해 기존 Active Directory 포리스트를 활용하려는 고객을 위해 Active Directory Connector가 있습니다.

다음 사항에 유의하세요.

- SFTP 전송 요구 사항에 맞게 기존 액티브 디렉터리 포리스트를 활용하려면 [Active Directory Connector](#)를 사용할 수 있습니다.
- 완전 관리형 서비스에서 Active Directory의 이점과 고가용성을 활용하려면 AWS Directory Service for Microsoft Active Directory를 사용할 수 있습니다. 자세한 내용은 [Microsoft Active Directory용 AWS Directory Service 사용](#)을 참조하세요.

이 주제에서는 Active Directory 커넥터 및 [Entra ID\(이전 Azure AD\) 도메인 서비스](#)를 사용하여 Entra ID로 SFTP Transfer 사용자를 인증하는 방법을 설명합니다.

주제

- [Entra ID 도메인 서비스용 AWS 디렉터리 서비스 사용을 시작하기 전에](#)
- [1단계: Entra ID 도메인 서비스 추가](#)
- [2단계: 서비스 계정 생성](#)
- [3단계: AD 커넥터를 사용하여 AWS 디렉터리 설정](#)
- [4단계: AWS Transfer Family 서버 설정](#)
- [5단계: 그룹에 액세스 권한 부여](#)
- [6단계: 사용자 테스트](#)

Entra ID 도메인 서비스용 AWS 디렉터리 서비스 사용을 시작하기 전에

Note

AWS Transfer Family의 기본 제한은 서버당 Active Directory 그룹 100개입니다. 사용 사례에 100개 이상의 그룹이 필요한 경우 용 사용자 지정 자격 증명 공급자를 [사용한 Active Directory 인증 간소화에 설명된 대로 사용자 지정 자격 증명 공급자 AWS Transfer Family](#)솔루션을 사용하는 것이 좋습니다.

의 AWS경우 다음이 필요합니다.

- Transfer Family 서버를 사용하는 AWS 리전의 Virtual Private Cloud(VPC)
- VPC에 최소 두 개의 프라이빗 서브넷이 있어야 함
- VPC는 인터넷에 연결되어 있어야 함
- Microsoft Entra와의 site-to-site VPN 연결을 위한 고객 게이트웨이 및 가상 프라이빗 게이트웨이

Microsoft Entra의 경우 다음이 필요합니다.

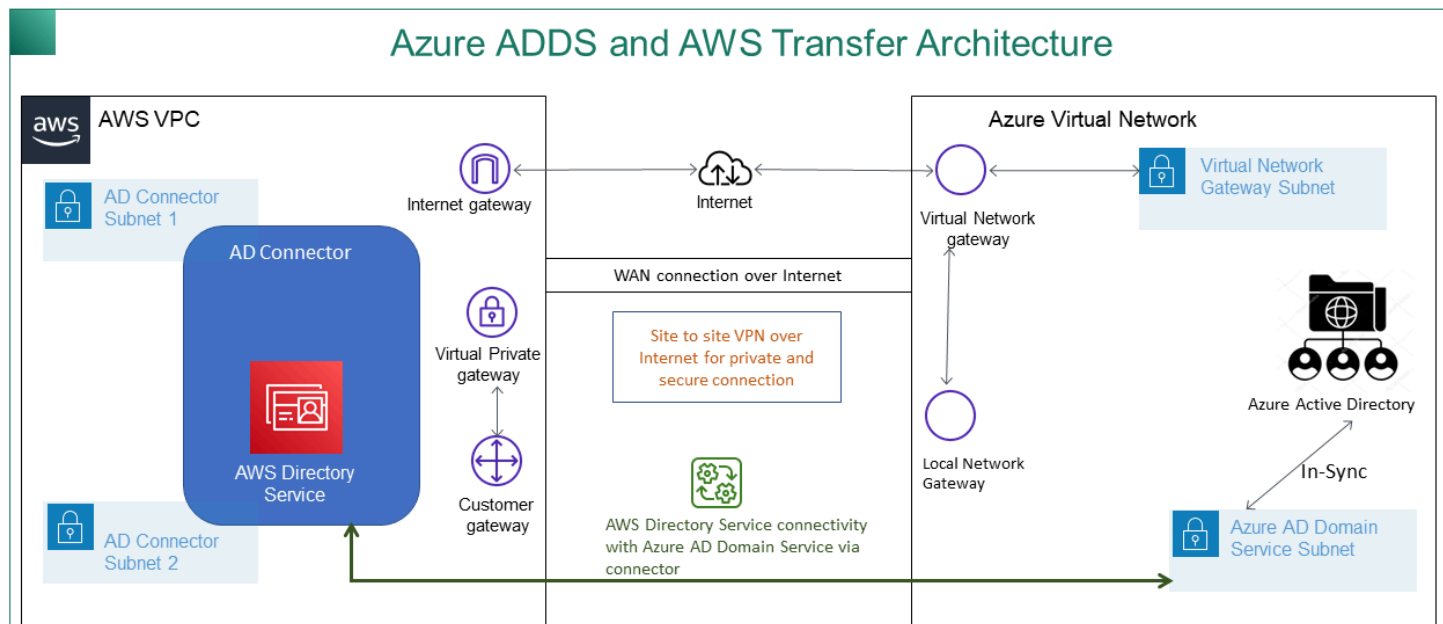
- Entra ID 및 Active Directory 도메인 서비스
- Entra 리소스 그룹
- Entra 가상 네트워크
- Amazon VPC와 Entra 리소스 그룹 간의 VPN 연결

 Note

네이티브 IPSEC 터널을 통하거나 VPN 어플라이언스를 사용하여 연결할 수 있습니다. 이 주제에서는 Entra 가상 네트워크 게이트웨이와 로컬 네트워크 게이트웨이 간에 IPSEC 터널을 사용합니다. Entra Domain Service 엔드포인트와 AWS VPC를 수용하는 서브넷 간의 트래픽을 허용하도록 터널을 구성해야 합니다.

- Microsoft Entra와의 site-to-site VPN 연결을 위한 고객 게이트웨이 및 가상 프라이빗 게이트웨이

다음 다이어그램은 시작하기 전에 필요한 구성을 보여줍니다.



1단계: Entra ID 도메인 서비스 추가

Entra ID는 기본적으로 도메인 조인 인스턴스를 지원하지 않습니다. 도메인 조인과 같은 작업을 수행하고 그룹 정책과 같은 도구를 사용하려면 관리자가 Entra ID 도메인 서비스를 활성화해야 합니다. Entra DS를 아직 추가하지 않았거나 기존 구현이 SFTP Transfer 서버에서 사용할 도메인과 연결되어 있지 않은 경우 새 인스턴스를 추가해야 합니다.

Entra ID 도메인 서비스 활성화에 대한 자세한 내용은 [자습서: Microsoft Entra 도메인 서비스 관리형 도메인 생성 및 구성을 참조하세요](#).

Note

Entra DS를 활성화할 때 SFTP Transfer 서버를 연결하려는 리소스 그룹 및 Entra 도메인에 대해 구성되었는지 확인합니다.


bob.us

 Azure AD Domain Services

«
Refresh
Delete

Overview

 Activity log

 Access control (IAM)

 Tags

 Settings

 Properties

 Configuration issues for your managed domain were detected. Run configuration diagnostics


bob.us

Running

View health

2단계: 서비스 계정 생성

Entra에는 Entra DS의 관리자 그룹에 속하는 서비스 계정이 하나 있어야 합니다. 이 계정은 AWS Active Directory 커넥터와 함께 사용됩니다. 이 계정이 Entra DS와 동기화되어 있는지 확인합니다.

[Home](#) > [Default Directory](#) > [Users](#) > [bobatusa](#)


bobatusa | Profile

 User

«
Edit
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

bobatusa

bobsmith@xyz.com

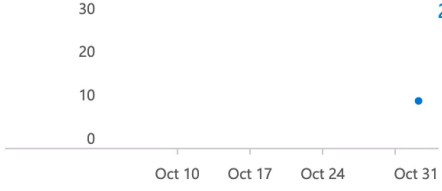


 Creation time
 10/6/2021, 1:32:27 AM

Identity

Name	First name	Last name
bobatusa	Bob	Smith
User Principal Name	User type	
bobsmith@xyz.com	Member	

User Sign-ins



Group memberships



Activity

- Sign-in logs
- Audit logs

Tip

SFTP 프로토콜을 사용하는 Transfer Family 서버에서는 Entra ID에 대한 다중 인증이 지원되지 않습니다. Transfer Family 서버는 사용자가 SFTP에 인증한 후에는 MFA 토큰을 제공할 수 없습니다. 연결을 시도하기 전에 MFA를 비활성화해야 합니다.

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

3단계: AD 커넥터를 사용하여 AWS 디렉터리 설정

Entra DS를 구성하고 AWS VPC와 Entra 가상 네트워크 간에 IPSEC VPN 터널이 있는 서비스 계정을 생성한 후에는 모든 AWS EC2 인스턴스에서 Entra DS DNS IP 주소를 ping하여 연결을 테스트할 수 있습니다.

연결이 활성화되었는지 확인한 후 아래에서 계속할 수 있습니다.

AD Connector를 사용하여 AWS 디렉터리를 설정하려면

1. [디렉터리 서비스](#) 콘솔을 열고 디렉터리를 선택합니다.
2. 디렉터리 설정을 선택합니다.
3. 디렉터리 타입으로는 AD Connector를 선택합니다.
4. 디렉터리 크기를 선택하고 다음을 선택한 다음 VPC와 서브넷을 선택합니다.
5. 다음을 선택하고 다음과 같이 필드를 채웁니다.
 - 디렉터리 DNS 이름: Entra DS에 사용 중인 도메인 이름을 입력합니다.
 - DNS IP 주소: Entra DS IP 주소를 입력합니다.
 - 서버 계정 사용자 이름 및 암호: 2단계: 서비스 계정 생성에서 만든 서비스 계정의 세부 정보를 입력합니다.

6. 화면을 완료하여 디렉터리 서비스를 생성합니다.

이제 디렉터리 상태가 활성화되어야 하며 SFTP 전송 서버에서 사용할 준비가 되었습니다.

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](#)

Find by directory ID or name

Refresh Actions **Set up directory**

< 1 > ⚙

	Directory ID	Directory name	Type	Size	Multi-Region	Status	Launch date
☐	d-906752c0d7	██████████	AD Connector	Small	Not applicable	Active	Nov 3, 2021

4단계: AWS Transfer Family 서버 설정

SFTP 프로토콜과 AWS Directory Service의 자격 증명 공급자 유형을 사용하여 Transfer Family 서버를 생성합니다. 디렉터리 드롭다운 목록에서 3단계: AD Connector를 사용하여 AWS 디렉터리 설정에서 추가한 디렉터를 선택합니다.

Note

Transfer Family 서버에서 사용한 Microsoft AD 디렉터리는 AWS Directory Service에서 삭제할 수 없습니다. 먼저 서버를 삭제해야 디렉터를 삭제할 수 있습니다.

5단계: 그룹에 액세스 권한 부여

서버를 생성한 후에는를 사용하여 활성화된 프로토콜을 통해 파일을 업로드하고 다운로드할 수 있는 디렉터리의 그룹을 선택해야 합니다 AWS Transfer Family. 액세스 권한을 생성하여 이 작업을 수행할 수 있습니다.

Note

사용자는 액세스 권한을 부여하는 그룹에 직접 속해야 합니다. 예를 들어 Bob이 사용자이고 groupA에 속하며 groupA 자체가 groupB에 포함되어 있다고 가정합니다.

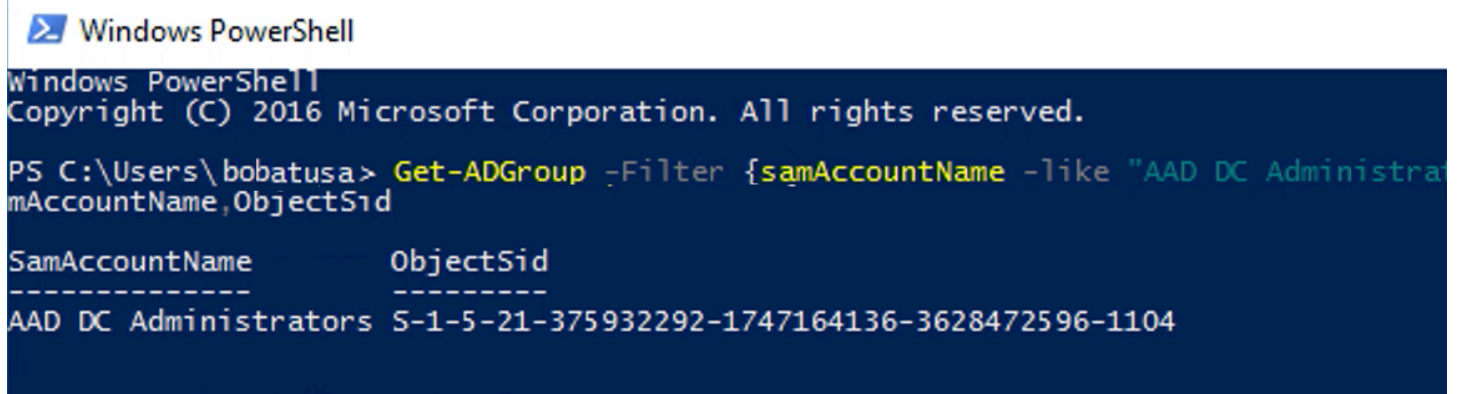
- 그룹 A에 액세스 권한을 부여하면 Bob에게 액세스 권한이 부여됩니다.

- 그룹 A가 아닌 그룹 B에 액세스 권한을 부여하는 경우 Bob은 액세스 권한을 갖지 않습니다.

액세스 권한을 부여하려면 그룹의 SID를 검색해야 합니다.

다음 Windows PowerShell 명령을 사용하여 그룹의 SID를 검색하고 *YourGroupName*을 그룹 이름으로 바꿉니다.

```
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 Administrators"} -Properties * | Select SamAccountName, ObjectSid

SamAccountName      ObjectSid
-----
AAD DC Administrators S-1-5-21-375932292-1747164136-3628472596-1104
```

그룹에 액세스 권한을 부여하려면

1. <https://console.aws.amazon.com/transfer/> 엽니다.
2. 서버 세부 정보 페이지로 이동한 다음 액세스 섹션에서 액세스 추가를 선택합니다.
3. 이전 절차의 출력에서 받은 SID를 입력합니다.
4. 액세스에서 그룹의 AWS Identity and Access Management 역할을 선택합니다.
5. 정책 섹션에서 정책을 선택합니다. 기본값은 없음입니다.
6. 홈 디렉터리에서 그룹의 홈 디렉터리에 해당하는 Amazon S3 버킷을 선택합니다.
7. 추가를 선택하여 연결을 생성합니다.

전송 서버의 세부 정보는 다음과 유사해야 합니다.

Protocols	Identity provider
Protocols over which clients can connect to your server's endpoint SFTP	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	/smb/windows	sftp-user-role ↗

6단계: 사용자 테스트

사용자가 서버의 AWS Managed Microsoft AD 디렉터리에 액세스할 수 있는지 테스트([테스트 사용자](#))할 수 있습니다. 사용자는 엔드포인트 구성 페이지의 액세스 섹션에 나열된 정확히 하나의 그룹(외부 ID)에 속해야 합니다. 사용자가 그룹에 속하지 않거나 둘 이상의 그룹에 속해 있는 경우 해당 사용자에게 액세스 권한이 부여되지 않습니다.

논리적 디렉터리를 사용하여 Transfer Family 디렉터리 구조를 단순화합니다.

논리적 디렉터리는 AWS Transfer Family 서버 디렉터리 구조를 단순화합니다. 논리적 디렉터리를 사용하면 Amazon S3 버킷 또는 Amazon EFS 파일 시스템에 연결할 때 사용자가 탐색할 수 있는 사용자 친화적인 이름으로 가상 디렉터리 구조를 생성할 수 있습니다. 이렇게 하면 사용자가 실제 디렉터리 경로, 버킷 이름 및 파일 시스템 이름을 볼 수 없습니다.

Note

최종 사용자가 수행하도록 허용하는 작업만 수행할 수 있도록 세션 정책을 사용해야 합니다. 논리적 디렉터리를 사용하여 최종 사용자를 위한 사용자 친화적인 가상 디렉터리를 생성하고 버킷 이름을 추상화해야 합니다. 논리적 디렉터리 매핑은 사용자가 지정된 논리적 경로 및 하위 디렉터리에만 액세스하고 논리적 루트를 통과하는 상대 경로를 금지할 수 있도록 허용합니다.

Transfer Family는 상대 요소를 포함할 수 있는 모든 경로를 검증하고 이러한 경로를 Amazon S3에 전달하기 전에 이러한 경로가 해결되지 않도록 적극적으로 차단합니다. 이렇게 하면 사용자가 논리적 매핑을 벗어나지 않습니다.

Transfer Family는 최종 사용자가 논리적 디렉터리 외부의 디렉터리에 액세스하지 못하도록 하지만 고유한 역할 또는 세션 정책을 사용하여 스토리지 수준에서 최소 권한을 적용하는 것이 좋습니다.

chroot 및 디렉터리 구조 이해

chroot 작업을 사용하면 사용자의 루트 디렉터리를 스토리지 계층 구조의 모든 위치로 설정할 수 있습니다. 이렇게 하면 사용자가 구성된 홈 또는 루트 디렉터리로 제한되어 상위 수준 디렉터리에 액세스할 수 없습니다.

Amazon S3 사용자가 로 제한되는 경우를 생각해 보세요 `amzn-s3-demo-bucket/home/${transfer:UserName}`. 그루트가 없으면 일부 클라이언트에서 사용자가 `/amzn-s3-demo-bucket/home`까지 이동할 수 있으므로 적절한 디렉터리로 돌아가려면 로그아웃 및 로그인が必要です. chroot 작업을 수행하면이 문제가 방지됩니다.

여러 버킷 및 접두사에서 사용자 지정 디렉터리 구조를 생성할 수 있습니다. 이는 워크플로에 버킷 접두사만으로는 제공할 수 없는 특정 디렉터리 레이아웃이 필요한 경우에 유용합니다. 디렉터리 경로가 파일 시스템의 다른 위치를 참조하는 Linux 파일 시스템에 심볼릭 링크를 생성하는 것과 비슷하게 Amazon S3 내의 여러 비연속 위치에 연결할 수도 있습니다.

논리적 디렉터리 사용 규칙

이 섹션에서는 논리적 디렉터리 사용에 대한 몇 가지 규칙 및 기타 고려 사항을 설명합니다.

매핑 제한

- 가 Entry인 경우 하나의 매핑만 허용됩니다 `"/`(중첩 경로는 허용되지 않음).
- 논리적 디렉터리는 사용자 지정 IDP 및 AD 사용자의 경우 최대 2.1MB, 서비스 관리형 사용자의 경우 2,000개 항목의 매핑을 지원합니다. 다음과 같이 매핑 크기를 계산할 수 있습니다.
 - 형식으로 일반적인 매핑을 작성합니다. `{"Entry":"/entry-path", "Target":"/target-path"}` 여기서 *entry-path* 및 *target-path*는 사용할 실제 값입니다.
 - 해당 문자열의 문자 수를 계산한 다음 하나(1)를 추가합니다.
 - 해당 수에 서버에 대해 보유한 대략적인 매핑 수를 곱합니다.

3단계에서 예상한 수가 2.1MB 미만인 경우 매핑이 허용 한도 내에 있는 것입니다.

대상 경로 요구 사항

- 버킷 또는 파일 시스템 경로가 사용자 이름에 따라 파라미터화된 경우 `${transfer:UserName}` 변수를 사용합니다.
- 연결된 IAM 역할에 해당 스토리지 위치에 액세스하는 데 필요한 권한이 있는 한 다른 Amazon S3 버킷 또는 파일 시스템을 가리키도록 대상을 구성할 수 있습니다.
- 모든 대상은 슬래시(/)로 시작해야 하지만 슬래시로 끝낼 수는 없습니다. 예를 들어 `/amzn-s3-demo-bucket/images`는 정답이고 `amzn-s3-demo-bucket/images` 와 `/amzn-s3-demo-bucket/images/`는 그렇지 않습니다.

스토리지 고려 사항

- Amazon S3는 폴더가 가상 개념으로만 존재하는 객체 스토어입니다. Amazon S3 스토리지를 사용할 때 Transfer Family는 후행 슬래시가 있는 0바이트 객체가 없더라도 STAT 작업에서 접두사를 디렉터리로 보고합니다. 후행 슬래시가 있는 적절한 0바이트 객체도 STAT 작업의 디렉터리로 보고됩니다. 이 동작은 [Amazon Simple Storage Service 사용 설명서의 폴더를 사용하여 Amazon S3 콘솔에서 객체 구성](#)에 설명되어 있습니다.
- 파일과 폴더를 구분해야 하는 애플리케이션의 경우 Amazon Elastic File System(Amazon EFS)을 Transfer Family 스토리지 옵션으로 사용합니다.
- 사용자의 논리적 디렉터리 값을 지정하는 경우 사용하는 파라미터는 사용자 유형에 따라 다릅니다.
 - 서비스 관리 사용자의 경우 HomeDirectoryMappings에서 논리적 디렉터리 값을 제공하세요.
 - 사용자 지정 자격 증명 공급자 사용자의 경우에 논리적 디렉터리 값을 제공합니다
다HomeDirectoryDetails.

사용자 디렉터리 값

- 논리적 디렉터리 값을 지정하기 위한 파라미터는 사용자 유형에 따라 다릅니다.
 - 서비스 관리 사용자의 경우 HomeDirectoryMappings에서 논리적 디렉터리 값을 제공하세요.
 - 사용자 지정 자격 증명 공급자 사용자의 경우에 논리적 디렉터리 값을 제공합니다
다HomeDirectoryDetails.

- LOGICAL HomeDirectoryType을 사용하는 경우 서비스 관리형 사용자, Active Directory 액세스 및 응답에 HomeDirectoryDetails가 제공되는 사용자 지정 자격 증명 공급자 구현에 대한 HomeDirectory 값을 지정할 수 있습니다. 지정하지 않으면 HomeDirectory의 기본값은 입니다/.

논리적 디렉터리를 구현하는 방법에 대한 자세한 내용은 섹션을 참조하세요 [논리적 디렉터리 구현](#).

논리적 디렉터리 구현

Important

루트 디렉터리 요구 사항

- Amazon S3 성능 최적화 설정을 사용하지 않는 경우 시작 시 루트 디렉터리가 있어야 합니다.
- Amazon S3의 경우 이는 슬래시(/)로 끝나는 0바이트 객체를 생성하는 것을 의미합니다/.
- 이 요구 사항을 방지하려면 서버를 생성하거나 업데이트할 때 Amazon S3 성능 최적화를 활성화하는 것이 좋습니다.
- LOGICAL HomeDirectoryType을 사용하여 HomeDirectory를 지정할 때 값은 논리적 디렉터리 매핑 중 하나에 매핑되어야 합니다. 이 서비스는 사용자 생성 및 업데이트 중에 이를 검증하여 작동하지 않는 구성을 방지합니다.

논리적 홈 디렉터리 구성

LOGICAL을 HomeDirectoryType으로 사용하는 경우 다음 사항에 유의하세요.

- HomeDirectory 값은 기존 논리적 디렉터리 매핑 중 하나와 일치해야 합니다.
- 시스템은 사용자 생성 및 업데이트 중에 이를 자동으로 검증합니다.
- 이 검증은 액세스 문제를 일으키는 구성을 방지합니다.

논리적 디렉터리 활성화

사용자의 논리적 디렉터를 사용하려면 HomeDirectoryType 파라미터를 로 설정합니다LOGICAL. 새 사용자를 생성하거나 기존 사용자를 업데이트할 때 이 작업을 수행합니다.

```
"HomeDirectoryType": "LOGICAL"
```

사용자 **chroot**에 대해 활성화

chroot의 경우, 각 사용자에게 대한 단일 Entry 및 Target 쌍으로 구성된 디렉터리 구조를 만드세요. 항목 /는 루트 폴더를 나타내는 반면, 대상은 버킷 또는 파일 시스템의 실제 위치를 지정합니다.

Example for Amazon S3

```
[{"Entry": "/", "Target": "/amzn-s3-demo-bucket/jane"}]
```

Example for Amazon EFS

```
[{"Entry": "/", "Target": "/fs-faa1a123/jane"}]
```

이전 예와 같이 절대 경로를 사용하거나, 다음 예와 같이 사용자 이름을 동적으로 `${transfer:UserName}`로 대체할 수 있습니다.

```
[{"Entry": "/", "Target":  
"/amzn-s3-demo-bucket${transfer:UserName}"}]
```

위 예에서 사용자는 루트 디렉터리에 잠겨 있어 계층 구조에서 상위 위치로 이동할 수 없습니다.

가상 디렉터리 구조

가상 디렉터리 구조의 경우, 사용자의 IAM 역할 매핑에 액세스 권한이 있는 한, 여러 버킷 또는 파일 시스템을 포함하여 S3 버킷 또는 EFS 파일 시스템 어디에나 대상을 사용하여 여러 Entry Target 쌍을 생성할 수 있습니다.

다음 가상 구조 예제에서는 사용자가 AWS SFTP에 로그인할 때, /pics, 및 하위 디렉터리가 있는 루트 디렉터리에 있습니다/doc/reporting/anotherpath/subpath/financials.

Note

Amazon S3 디렉터리의 성능을 최적화하도록 선택하지 않는 한(서버를 생성하거나 업데이트 할 때) 사용자 또는 관리자는 디렉터리가 아직 없는 경우 디렉터리를 생성해야 합니다. 이 문제를 피하는 것은 Amazon S3 성능 최적화를 고려해야 하는 이유입니다.
Amazon EFS의 경우 관리자가 논리적 매핑 또는 / 디렉터리를 생성해야 합니다.

[

```
{
  "Entry": "/pics", "Target": "/amzn-s3-demo-bucket1/pics"},
  "Entry": "/doc", "Target": "/amzn-s3-demo-bucket1/anotherpath/docs"},
  "Entry": "/reporting", "Target": "/amzn-s3-demo-bucket2/Q1"},
  "Entry": "/anotherpath/subpath/financials", "Target": "/amzn-s3-demo-bucket2/financials"}]
```

Note

매핑한 특정 폴더에만 파일을 업로드할 수 있습니다. 즉, 이전 예에서는 /anotherpath 또는 anotherpath/subpath 디렉터리에는 업로드할 수 없고; anotherpath/subpath/financials에만 업로드할 수 있습니다. 또한 경로가 겹칠 수 없으므로 이러한 경로에 직접 매핑할 수 없습니다.

예를 들어, 다음 매핑을 형성한다고 가정하겠습니다:

```
{
  "Entry": "/pics",
  "Target": "/amzn-s3-demo-bucket/pics"
},
{
  "Entry": "/doc",
  "Target": "/amzn-s3-demo-bucket/mydocs"
},
{
  "Entry": "/temp",
  "Target": "/amzn-s3-demo-bucket2/temporary"
}
```

해당 버킷에만 파일을 업로드할 수 있습니다. sftp를 통해 처음 연결하면 루트 디렉터리인 /(으)로 이동됩니다. 해당 디렉터리에 파일을 업로드하려고 하면 업로드가 실패합니다. 다음 명령은 예 순서를 나타냅니다:

```
sftp> pwd
Remote working directory: /
sftp> put file
Uploading file to /file
remote open("/file"): No such file or directory
```

임의의 directory/sub-directory 위치에 업로드하려면 경로를 sub-directory에 명시적으로 매핑해야 합니다.

다운로드하여 사용할 수 있는 AWS CloudFormation 템플릿을 포함하여 논리적 디렉터리 구성 및 사용자에게 chroot 대한 자세한 내용은 AWS 스토리지 블로그의 [chroot 및 논리적 디렉터리를 사용하여 AWS SFTP 구조 간소화](#)를 참조하세요.

논리적 디렉터리 구성 예제

이 예에서는 사용자를 생성하고 두 개의 논리적 디렉터리를 할당합니다. 다음 명령을 실행하면 (기존 Transfer Family 서버의 경우) 논리적 디렉터리 pics 및 doc 를 사용하여 새 사용자가 만들어집니다.

```
aws transfer create-user \
  --user-name marymajor \
  --server-id s-11112222333344445 \
  --role arn:aws:iam::1234abcd5678:role/marymajor-role \
  --home-directory-type LOGICAL \
  --home-directory-mappings "[{\"Entry\":\"/pics\", \"Target\":\"/amzn-s3-demo-bucket1/pics\"}, {\"Entry\":\"/doc\", \"Target\":\"/amzn-s3-demo-bucket2/test/mydocs\"}]" \
  --ssh-public-key-body file://~/.ssh/id_rsa.pub
```

marymajor가 기존 사용자이고 홈 디렉터리 타입이 PATH인 경우 이전 사용자와 비슷한 명령을 사용하여 그것을 LOGICAL로 변경할 수 있습니다.

```
aws transfer update-user \
  --user-name marymajor \
  --server-id s-11112222333344445 \
  --role arn:aws:iam::1234abcd5678:role/marymajor-role \
  --home-directory-type LOGICAL \
  --home-directory-mappings "[{\"Entry\":\"/pics\", \"Target\":\"/amzn-s3-demo-bucket1/pics\"}, {\"Entry\":\"/doc\", \"Target\":\"/amzn-s3-demo-bucket2/test/mydocs\"}]"
```

유념할 사항:

- 디렉터리 /amzn-s3-demo-bucket1/pics 및 /amzn-s3-demo-bucket2/test/mydocs 없는 경우 사용자 (또는 관리자) 가 해당 디렉터리를 만들어야 합니다.

Note

이러한 디렉터리는 최적화된 디렉터리를 구성한 경우 Transfer Family 서버에서 자동으로 생성됩니다.

- 가 서버에 **marymajor** 연결하고 `ls -l` 명령을 실행하면 Mary는 다음을 확인합니다.

```
drwxr--r--  1      -      -      0 Mar 17 15:42 doc
drwxr--r--  1      -      -      0 Mar 17 16:04 pics
```

- marymajor**은(는) 이 수준에서는 파일이나 디렉터리를 만들 수 없습니다. 하지만 `pics` 및 `doc` 내에서 하위 디렉토리를 추가할 수는 있습니다.
- Mary가 `pics` 밑에 추가하는 파일은 `/amzn-s3-demo-bucket2/test/mydocs` 각각 Amazon S3 경로 `/amzn-s3-demo-bucket1/pics` 밑에 `doc` 추가됩니다.
- 이 예에서는 이러한 가능성을 설명하기 위해 서로 다른 두 개의 버킷을 지정합니다. 하지만 사용자에 대해 지정한 여러 논리 디렉터리 또는 모든 논리적 디렉터리에 동일한 버킷을 사용할 수 있습니다.

이 예제에서는 논리적 홈 경로에 대한 대체 구성을 제공합니다.

```
aws transfer create-user \
  --user-name marymajor \
  --server-id s-11112222333344445 \
  --role arn:aws:iam::1234abcd5678:role/marymajor-role \
  --home-directory-type LOGICAL \
  --home-directory /home/marymajor \
  --home-directory-mappings "[{\"Entry\":\"/home/marymajor/pics\", \"Target\":\"/amzn-s3-demo-bucket1/pics\"}, {\"Entry\":\"/home/marymajor/doc\", \"Target\":\"/amzn-s3-demo-bucket2/test/mydocs\"}]" \
  --ssh-public-key-body file:///~/.ssh/id_rsa.pub
```

다음 사항에 유의하세요.

- 매핑은 두 논리적 경로의 첫 번째 부분 `/home/marymajor`인 공통 경로를 제공합니다. 그런 다음 파일을 `pics` 및 `doc` 폴더에 추가할 수 있습니다.
- 이전 예제와 마찬가지로 홈 디렉터리 `/home/marymajor`는 읽기 전용입니다.

Amazon EFS를 위한 논리적 디렉터리 구성

Transfer Family 서버에서 Amazon EFS를 사용하는 경우, 사용자가 논리적 홈 디렉터리에서 작업할 수 있으려면 먼저 읽기 및 쓰기 권한이 있는 사용자의 홈 디렉터리를 생성해야 합니다. 사용자는 논리적 홈 디렉터리 상의 `mkdir`에 대한 권한이 없기 때문에 이 디렉터를 직접 만들 수는 없습니다.

사용자의 홈 디렉터리가 존재하지 않고 사용자가 1s 명령을 실행하면 시스템은 다음과 같이 응답합니다:

```
sftp> ls
remote readdir ("/"): No such file or directory
```

상위 디렉터리에 대한 관리 액세스 권한이 있는 사용자는 사용자의 논리적 홈 디렉터리를 생성해야 합니다.

사용자 지정 AWS Lambda 응답

맞춤 ID 공급자에 연결하는 Lambda 함수와 함께 논리적 디렉터리를 사용할 수 있습니다. 이렇게 하려면 Lambda 함수에서 HomeDirectoryType를 **LOGICAL**(으)로 지정하고, HomeDirectoryDetails 파라미터에 Entry와(과) Target를 추가합니다. 예제:

```
HomeDirectoryType: "LOGICAL"
HomeDirectoryDetails: "[{\"Entry\": \"\\\", \"Target\": \"\\\"/amzn-s3-demo-bucket/
theRealFolder\"}]\"
```

다음 코드는 맞춤 Lambda 인증 호출로부터의 응답 성공의 예입니다.

```
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": "{\n\"Role\": \"arn:aws:iam::123456789012:role/bob-usa-role\",
    \"HomeDirectoryType\": \"LOGICAL\",
    \"HomeDirectoryDetails\": \"[{\\\"Entry\\\":\\\"/myhome\\\",\\\"
\\\"Target\\\":\\\"/amzn-s3-demo-bucket/theRealFolder\\\"]\\\",
    \"PublicKeys\": \"[ssh-rsa myrsapubkey]\"}",
  "StatusCode": 200
}
```

Note

이 "Ur1": 행은 API Gateway 메서드를 맞춤 ID 공급자로 사용하는 경우에만 반환됩니다.

Transfer Family를 사용하여 FSx for NetApp ONTAP 파일 시스템에 액세스

목차

- [개요](#)
- [사전 조건](#)
 - [FSx for NetApp ONTAP 요구 사항](#)
 - [필수 IAM 권한](#)
- [FSx 스토리지가 Transfer Family와 작동하는 방식](#)
 - [파일 시스템 사용자 자격 증명](#)
- [FSx용 S3 액세스 포인트 생성](#)
 - [액세스 포인트 이름 지정](#)
 - [FSx for NetApp ONTAP에 대한 액세스 포인트 생성](#)
 - [파일 시스템 권한 구성](#)
- [FSx에서 S3 액세스 포인트 별칭 사용](#)
 - [액세스 포인트 별칭 정보](#)
 - [액세스 포인트 별칭 찾기](#)
- [FSx 스토리지용 Transfer Family 구성](#)
 - [IAM 역할 생성](#)
- [FSx 스토리지 사용자 관리](#)
 - [사용자 생성](#)
 - [여러 디렉터리 매핑 구성](#)
- [파일 전송 클라이언트 구성](#)
 - [WinSCP 구성](#)
 - [기타 SFTP 클라이언트](#)
- [FSx 스토리지 문제 해결](#)
 - [파일 작업 문제](#)

개요

Transfer Family는 S3 액세스 포인트를 통해 Amazon FSx for NetApp ONTAP을 지원합니다. Amazon FSx for NetApp ONTAP은 NetApp의 인기 있는 ONTAP 파일 시스템에 구축된 매우 안정적이고 확장 가능하며 성능이 뛰어나고 기능이 풍부한 파일 스토리지를 제공하는 완전 관리형 서비스입니다. FSx 파일 시스템으로 Transfer Family를 구성하면 사용자는 표준 파일 전송 클라이언트를 사용하여 Transfer Family 엔드포인트에 연결합니다. Transfer Family는 데이터가 FSx 파일 시스템에 남아 있는 동안 FSx 볼륨에 연결된 S3 액세스 포인트를 통해 파일 작업을 라우팅합니다. FSx for NetApp ONTAP에 대한 자세한 내용은 [Amazon FSx for NetApp ONTAP이란 무엇입니까?](#)를 참조하세요.

이 통합을 통해 다음을 수행할 수 있습니다.

- SFTP, FTPS 또는 FTP 프로토콜을 사용하여 파일을 엔터프라이즈급 파일 스토리지로 전송
- 여러 프로토콜(SFTP, NFS, SMB)을 통해 동일한 데이터에 액세스
- 스냅샷, 백업 및 데이터 계층화와 같은 FSx 기능 사용

Important

이름 바꾸기 및 추가 작업을 포함하여 Transfer Family에서 FSx 파일 시스템을 사용할 때는 일부 파일 작업이 지원되지 않습니다. 업로드 작업의 경우 파일 크기는 5GB로 제한됩니다. 제한 사항의 전체 목록은 [액세스 포인트 호환성을 참조하세요](#).

사전 조건

Amazon FSx로 Transfer Family를 구성하기 전에 다음 요구 사항을 충족해야 합니다.

FSx for NetApp ONTAP 요구 사항

Transfer Family와 함께 FSx for NetApp ONTAP을 사용하려면 다음이 필요합니다.

- ONTAP 버전 9.17.1 이상을 실행하는 FSx for NetApp ONTAP 파일 시스템
- 동일한 AWS 리전의 파일 시스템 및 S3 액세스 포인트
- 파일 시스템과 액세스 포인트를 모두 소유한 동일한 AWS 계정

자세한 내용은 [Amazon FSx for NetApp ONTAP 시작하기를 참조하세요](#).

필수 IAM 권한

각 S3 액세스 포인트를 사용하여 이루어진 모든 요청에 S3가 적용하는 고유한 권한 및 네트워크 제어를 사용하여 각 S3 액세스 포인트를 구성할 수 있습니다. S3 액세스 포인트는 리소스, 사용자 또는 기타 조건별로 액세스 포인트 사용을 제어하는 데 사용할 수 있는 IAM 리소스 정책을 지원합니다. 애플리케이션 또는 사용자가 액세스 포인트를 통해 파일에 액세스하려면 액세스 포인트와 기본 볼륨 모두에서 요청을 허용해야 합니다. 자세한 내용은 [IAM 액세스 포인트 정책을](#) 참조하세요.

FSx용 Amazon S3 액세스 포인트는 IAM 권한과 파일 시스템 수준 권한을 결합하는 이중 계층 권한 부여 모델을 사용합니다. 이 접근 방식을 사용하면 AWS 서비스 수준과 기본 파일 시스템 수준 모두에서 데이터 액세스 요청이 제대로 승인됩니다.

애플리케이션 또는 사용자가 액세스 포인트를 통해 데이터에 성공적으로 액세스하려면 S3 액세스 포인트 정책과 기본 FSx 볼륨 모두 요청을 허용해야 합니다.

이 통합을 생성하고 구성하려면 다음 권한이 필요합니다.

- fsx:CreateAndAttachS3AccessPoint
- s3:CreateAccessPoint
- s3:GetAccessPoint
- s3:PutAccessPointPolicy (선택 사항인 액세스 포인트 정책을 생성하는 경우)

FSx 스토리지가 Transfer Family와 작동하는 방식

FSx 파일 시스템으로 Transfer Family를 구성하면 다음 구성 요소가 함께 작동하여 파일 전송을 활성화합니다.

1. 사용자는 SFTP, FTPS 또는 FTP 클라이언트를 사용하여 Transfer Family 서버에 연결합니다.
2. Transfer Family는 서비스 관리형 자격 증명, 사용자 지정 자격 증명 공급자 또는를 사용하여 사용자를 인증합니다 AWS Directory Service for Microsoft Active Directory. 인증되면 Transfer Family는 사용자와 연결된 IAM 역할을 수임합니다.
3. 각 파일 작업에 대해 Transfer Family는 사용자의 수임된 IAM 역할을 사용하여 S3 액세스 포인트에 요청하는 표준 S3 API 클라이언트 역할을 하며 S3 액세스 포인트 정책에 대한 권한을 확인합니다.
4. FSx 파일 시스템은 액세스 포인트와 연결된 파일 시스템 사용자에게 요청된 작업을 수행할 권한이 있는지 확인합니다. 그런 다음 FSx 볼륨에서 파일 작업이 수행됩니다.

파일 작업이 성공하려면 두 권한 부여 계층 모두 요청을 허용해야 합니다.

Note

S3 액세스 포인트를 FSx 볼륨에 연결해도 NFS 또는 SMB를 통해 직접 액세스할 때 볼륨의 동작 방식은 변경되지 않습니다. 기존 파일 프로토콜 액세스는 변경되지 않고 계속 작동합니다.

파일 시스템 사용자 자격 증명

각 액세스 포인트는 액세스 포인트를 생성할 때 지정하는 파일 시스템 사용자 자격 증명을 사용합니다. 이 자격 증명은 해당 액세스 포인트를 통해 이루어진 모든 파일 액세스 요청을 승인합니다. 파일 시스템 사용자는 기본 Amazon FSx 파일 시스템의 사용자 계정입니다. 파일 시스템 사용자에게 읽기 전용 액세스 권한이 있는 경우 액세스 포인트를 사용하여 수행된 읽기 요청만 승인되고 쓰기 요청은 차단됩니다. 파일 시스템 사용자에게 읽기-쓰기 액세스 권한이 있는 경우 액세스 포인트를 사용하여 만든 연결된 볼륨에 대한 읽기 및 쓰기 요청이 모두 승인됩니다.

FSx용 S3 액세스 포인트 생성

Transfer Family를 구성하기 전에 FSx 볼륨에 연결된 S3 액세스 포인트를 생성해야 합니다. S3 액세스 포인트는 버킷 또는 Amazon FSx for ONTAP 볼륨과 같은 데이터 소스에 연결된 명명된 네트워크 엔드포인트입니다. Amazon FSx 콘솔, AWS CLI 또는 API를 사용하여 액세스 포인트를 생성하고 FSx for NetApp ONTAP에 연결할 수 있습니다. 연결되면 S3 객체 API를 사용하여 파일 데이터에 액세스할 수 있습니다. 데이터는 Amazon FSx 파일 시스템에 계속 상주하며 기존 워크로드에서 직접 액세스할 수 있습니다. 백업, 스냅샷, 사용자 및 그룹 할당량, 압축 등 모든 FSx for NetApp ONTAP 스토리지 관리 기능을 사용하여 스토리지를 계속 관리합니다.

자세한 내용은 [액세스 포인트 생성](#)을 참조하십시오.

액세스 포인트 이름 지정

액세스 포인트의 이름을 지정할 때 다음 지침을 따릅니다.

- 액세스 포인트 이름은 AWS 계정 및 리전 내에서 고유해야 합니다.
- 이름은 -ext-s3alias (별칭에 대해 예약됨)로 끝날 수 없습니다.
- 민감한 정보는 DNS에 게시되므로 이름에 포함하지 마세요.

이름 지정 규칙의 전체 목록은 [액세스 포인트 이름 지정 규칙, 제한 및 제한](#)을 참조하세요.

FSx for NetApp ONTAP에 대한 액세스 포인트 생성

다음 절차에 따라 FSx for NetApp ONTAP 볼륨에 대한 S3 액세스 포인트를 생성합니다.

액세스 포인트를 생성하려면(콘솔)

1. <https://console.aws.amazon.com/fsx/>에서 Amazon FSx 콘솔을 엽니다.
2. 탐색 창에서 파일 시스템을 선택합니다.
3. FSx for NetApp ONTAP 파일 시스템을 선택합니다.
4. 볼륨 탭을 선택합니다.
5. 연결할 볼륨을 선택합니다.
6. 작업에서 S3 액세스 포인트 생성을 선택합니다.
7. 액세스 포인트 이름에 설명이 포함된 이름(예: transfer-family-ap)을 입력합니다.
8. 파일 시스템 사용자 자격 증명 유형에서 다음 중 하나를 선택합니다.
 - UNIX 자격 증명 - UNIX 보안 스타일이 있는 볼륨의 경우
 - Windows 자격 증명 - NTFS 보안 스타일이 있는 볼륨의 경우
9. (선택 사항) 액세스 포인트 정책에 이 액세스 포인트를 통해 액세스하는 객체에 대해 수행할 수 있는 IAM 보안 주체를 정의하는 IAM 정책을 입력합니다. 자세한 내용은 [액세스 포인트 액세스 관리를 참조](#)하세요.
10. 생성(Create)을 선택합니다.
11. 생성 후 Transfer Family 구성에 사용할 액세스 포인트 별칭을 기록해 둡니다.

Note

가 연결된 SFTP/FTPS 사용자를 대신하여 S3 리소스에 AWS Transfer Family 액세스하는 경우 요청은 VPC가 아닌 AWS Transfer Family 인프라에서 시작됩니다. 따라서 VPC 네트워크 오리진으로 구성된 S3 액세스 포인트는 이러한 요청을 거부합니다. 그러나 인터넷 네트워크 오리진으로 구성된 액세스 포인트를 사용하더라도 Transfer Family와 액세스 포인트 간의 모든 트래픽은 비공개로 유지되며 AWS 백본 네트워크를 통해 이동하므로 퍼블릭 인터넷을 통과하지 않습니다.

파일 시스템 권한 구성

지정하는 파일 시스템 사용자는 Transfer Family 사용자가 수행할 수 있는 작업을 결정합니다. FSx 볼륨에 대한 적절한 권한을 구성해야 합니다.

UNIX 예제:

```
# Create a directory for Transfer Family users
mkdir -p /vol1/transfer-users

# Set ownership to match the access point user
chown 1001:1001 /vol1/transfer-users

# Set permissions
chmod 755 /vol1/transfer-users
```

Windows 예:

```
# Create a directory for Transfer Family users
New-Item -Path "D:\vol1\transfer-users" -ItemType Directory

# Set permissions for the file system user associated with the access point
# Replace DOMAIN\TransferUser with your Windows user identity
icacls "D:\vol1\transfer-users" /grant "DOMAIN\TransferUser:(OI)(CI)M" /T

# Verify permissions
icacls "D:\vol1\transfer-users"
```

FSx에서 S3 액세스 포인트 별칭 사용

Transfer Family에서 FSx 파일 시스템을 사용하는 경우 S3 액세스 포인트 별칭을 사용해야 합니다. Transfer Family는 FSx 스토리지에 대한 액세스 포인트 ARNs 또는 기타 참조 방법 사용을 지원하지 않습니다.

Important

AWS Transfer Family 는 FSx 파일 시스템을 사용할 때만 S3 액세스 포인트 별칭을 지원합니다. 액세스 포인트 ARNs 또는 virtual-hosted-style URIs 사용할 수 없습니다.

⚠ Important

액세스 포인트는 볼륨과 동일한 리전에 있어야 합니다.

액세스 포인트 별칭 정보

FSx 볼륨에 연결된 S3 액세스 포인트를 생성하면 Amazon S3가 액세스 포인트 별칭을 자동으로 생성합니다. 이 별칭은 S3 버킷 이름을 사용하는 모든 곳에서 사용할 수 있는 고유 식별자입니다.

FSx 볼륨에 연결된 액세스 포인트의 경우 별칭은 다음 형식을 사용합니다.

```
access-point-name-metadata-ext-s3alias
```

예제 별칭:

```
my-fsx-ap-aqfqprnstn7aefdfbarligizwggyfouse1a-ext-s3alias
```

i Note

접미사는 -ext-s3alias FSx 액세스 포인트 별칭용으로 예약되어 있습니다. 액세스 포인트 이름에는이 접미사를 사용할 수 없습니다.

액세스 포인트 별칭 찾기

액세스 포인트를 생성한 후 액세스 포인트 별칭을 찾을 수 있습니다.

액세스 포인트 별칭을 찾으려면(콘솔)

1. <https://console.aws.amazon.com/fsx/>에서 Amazon FSx 콘솔을 엽니다.
2. 탐색 창에서 파일 시스템을 선택합니다.
3. 파일 시스템을 선택합니다.
4. 볼륨 탭을 선택하고 액세스 포인트를 생성한 볼륨을 선택합니다.
5. S3 액세스 포인트 세부 정보 열로 이동합니다.
6. 별칭은 별칭 열에 표시됩니다.

액세스 포인트 별칭을 찾으려면(CLI)

`describe-s3-access-point-attachments` 명령을 사용합니다.

```
aws fsx describe-s3-access-point-attachments \
  --filters Name=file-system-id,Values=fs-0123456789abcdef0
```

응답에는 별칭이 포함됩니다.

```
{
  "S3AccessPointAttachments": [
    {
      "S3AccessPoint": {
        "ResourceARN": "arn:aws:s3:us-east-1:111122223333:accesspoint/my-fsx-
ap",
        "Alias": "my-fsx-ap-aqfqprnstn7aefdfbarligizwgyfouse1a-ext-s3alias"
      }
    }
  ]
}
```

Transfer Family 사용자를 구성할 때 홈 디렉터리 매핑에서 액세스 포인트 별칭을 사용합니다.

홈 디렉터리 형식:

```
/access-point-alias/path/to/directory
```

예:

```
/my-fsx-ap-aqfqprnstn7aefdfbarligizwgyfouse1a-ext-s3alias/users/jsmith
```

FSx 스토리지용 Transfer Family 구성

S3 액세스 포인트를 생성한 후 이를 사용하도록 Transfer Family 서버를 구성합니다.

IAM 역할 생성

Transfer Family에 S3 액세스 포인트에 대한 액세스 권한을 부여하는 IAM 역할을 생성해야 합니다.

Important

IAM 정책에는 별칭이 아닌 액세스 포인트 ARN 형식이 필요합니다.

`arn:aws:s3:region:account-id:accesspoint/access-point-name` IAM 정책 리소

스 문의 형식을 사용합니다. 액세스 포인트 별칭(로 끝남 -ext-s3alias)은 홈 디렉터리 매핑에만 사용됩니다.

IAM 역할을 만들려면

1. IAM 콘솔(<https://console.aws.amazon.com/iam/>)을 엽니다.
2. 탐색 창에서 역할을 선택한 다음 역할 생성을 선택합니다.
3. 신뢰할 수 있는 엔터티 유형에서 AWS 서비스를 선택합니다.
4. 사용 사례에서 전송을 선택합니다.
5. 다음을 선택합니다.
6. 정책 생성을 선택하고 정책을 입력합니다(아래 샘플 정책 참조).
7. 정책을 역할에 연결하고 역할 생성을 선택합니다.

IAM 정책의 예:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowFileOperations",
      "Effect": "Allow",
      "Action": [
        "s3:GetObject",
        "s3:PutObject",
        "s3:DeleteObject",
        "s3:GetObjectTagging",
        "s3:PutObjectTagging"
      ],
      "Resource": "arn:aws:s3:us-east-2:111122223333:accesspoint/my-fsx-ap/object/*"
    },
    {
      "Sid": "AllowDirectoryOperations",
      "Effect": "Allow",
      "Action": [
        "s3:ListBucket",
        "s3:GetBucketLocation"
      ],

```

```

    "Resource": "arn:aws:s3:us-east-2:111122223333:accesspoint/my-fsx-ap"
  }
]
}

```

FSx 스토리지 사용자 관리

S3 액세스 포인트 별칭을 사용하는 홈 디렉터리 매핑을 사용하여 Transfer Family 사용자를 생성합니다.

사용자 생성

FSx 스토리지 사용자를 생성할 때는 홈 디렉터리 매핑에서 액세스 포인트 별칭을 사용합니다.

서비스 관리형 사용자를 생성하려면(콘솔)

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 탐색 창에서 서버를 선택합니다.
3. 서버를 선택합니다.
4. 사용자 섹션에서 사용자 추가를 선택합니다.
5. 사용자 이름에 사용자 이름을 입력합니다.
6. 역할에서 생성한 IAM 역할을 선택합니다.
7. 홈 디렉터리에서 제한을 선택합니다.
8. 홈 디렉터리 매핑의 경우 액세스 포인트 별칭을 사용하여 매핑을 추가합니다.

```

[{"Entry": "/", "Target": "/my-fsx-ap-aqfqprnstn7aefdfbarligizwgfyfouse1a-ext-s3alias/users/jsmith"}]

```

사용자를 생성하려면(CLI)

create-user 명령을 사용합니다. 액세스 포인트 별칭을 별칭으로 바꿉니다.

```

aws transfer create-user \
  --server-id s-0123456789abcdef0 \
  --user-name jsmith \
  --role arn:aws:iam::111122223333:role/TransferFamilyFSxRole \
  --home-directory-type LOGICAL \

```

```
--home-directory-mappings '[
  {
    "Entry": "/",
    "Target": "/my-fsx-ap-aqfqprnstn7aefdfbarligizwggyfouse1a-ext-s3alias/users/
jsmith"
  }
]'
```

여러 디렉터리 매핑 구성

여러 가상 디렉터리를 FSx 볼륨의 다른 경로에 매핑할 수 있습니다.

예: 별도의 업로드 및 다운로드 디렉터리

```
aws transfer create-user \
  --server-id s-0123456789abcdef0 \
  --user-name jsmith \
  --role arn:aws:iam::111122223333:role/TransferFamilyFSxRole \
  --home-directory-type LOGICAL \
  --home-directory-mappings '[
    {
      "Entry": "/inbox",
      "Target": "/my-fsx-ap-aqfqprnstn7aefdfbarligizwggyfouse1a-ext-s3alias/users/
jsmith/inbox"
    },
    {
      "Entry": "/outbox",
      "Target": "/my-fsx-ap-aqfqprnstn7aefdfbarligizwggyfouse1a-ext-s3alias/users/
jsmith/outbox"
    }
  ]'
```

파일 전송 클라이언트 구성

Transfer Family에서 FSx 파일 시스템을 사용하는 경우 지원되지 않는 기능을 비활성화하도록 파일 전송 클라이언트를 구성해야 합니다.

WinSCP 구성

WinSCP는 기본적으로 FSx용 S3 액세스 포인트에서 지원되지 않는 임시 이름 바꾸기 기능을 사용합니다.

 Warning

WinSCP에서 임시 이름 바꾸기 기능을 비활성화하지 않으면 파일 업로드가 실패합니다.

WinSCP에서 임시 이름 바꾸기를 비활성화하려면

1. WinSCP를 엽니다.
2. 로그인 대화 상자에서 편집을 선택하여 세션 설정을 수정합니다.
3. 고급을 선택합니다.
4. 왼쪽 탐색의 전송에서 내구성을 선택합니다.
5. 임시 파일 이름으로 전송 재개/전송 활성화에서 비활성화를 선택합니다.
6. 확인을 선택하여 설정을 저장합니다.

또는 기존 세션에 대해이 설정을 비활성화할 수 있습니다.

1. Transfer Family 서버에 연결합니다.
2. 옵션을 선택한 다음 기본 설정을 선택합니다.
3. 전송을 선택한 다음 내구성을 선택합니다.
4. 임시 파일 이름으로 전송 재개/전송 활성화에서 비활성화를 선택합니다.
5. 확인을 선택합니다.

기타 SFTP 클라이언트

다른 SFTP 클라이언트의 경우 사용 가능한 경우 다음 기능을 비활성화합니다.

- 임시 파일 업로드(임시 파일로 업로드한 다음 이름 바꾸기)
- 임시 파일을 사용하여 전송 재개
- 이름 바꾸기 작업을 사용한 원자성 업로드
- 업로드를 위한 추가 모드

특정 구성 단계는 클라이언트 설명서를 참조하세요.

FSx 스토리지 문제 해결

이 섹션에서는 FSx 파일 시스템에서 Transfer Family를 사용할 때 발생하는 일반적인 문제를 식별하고 해결하는 방법을 설명합니다.

파일 작업 문제

권한 거부됨

권한 거부 오류가 발생하는 경우:

1. IAM 역할에 액세스 포인트 별칭에 대한 올바른 권한이 있는지 확인합니다. S3 APIs.
2. 액세스 포인트 정책이 IAM 역할을 허용하는지 확인합니다.
3. 파일 시스템 사용자에게 대상 경로에 대한 권한이 있는지 확인합니다.
4. 홈 디렉터리 매핑이 올바른 액세스 포인트 별칭을 사용하는지 확인합니다.

WinSCP에서 업로드 실패

WinSCP에서 파일 업로드에 실패하면 임시 이름 바꾸기를 비활성화합니다.

1. WinSCP에서 옵션을 선택한 다음 기본 설정을 선택합니다.
2. 전송을 선택한 다음 내구성을 선택합니다.
3. 임시 파일 이름으로 전송 재개/전송 활성화에서 비활성화를 선택합니다.

자세한 내용은 [the section called “파일 전송 클라이언트 구성”](#) 단원을 참조하십시오.

파일 업로드 실패

파일 업로드에 실패하는 경우:

1. 파일 크기가 5GB 미만인지 확인합니다.
2. FSx 볼륨에 사용 가능한 스토리지가 충분한지 확인합니다.
3. CloudWatch 지표의 제한을 모니터링합니다.

Transfer Family 웹 앱

웹 앱을 생성하여 웹 브라우저를 통해 Amazon Simple Storage Service(S3)와 데이터를 주고받는 간단한 인터페이스를 활성화할 수 있습니다. 따라서 서버를 생성하거나 프로비저닝 AWS Transfer Family 할 필요가 없습니다.

Transfer Family 웹 앱을 도입하기 전에 최종 사용자는 클라이언트, 사용자 지정 빌드 또는 타사 솔루션을 사용하여 Amazon S3의 데이터에 액세스해야 했습니다. 이는 고객과 파트너의 엄격한 보안 요구 사항과 클라이언트 앱이 기술적이지 않은 사용자가 운영하기 어렵기 때문입니다.

웹 앱이 출시되면 이제 최종 사용자가 Amazon S3에서 데이터를 검색, 업로드 및 다운로드할 수 있도록 안전하고 가용성이 높은 브랜드가 지정된 포털을 확장할 수 있습니다. 웹 앱은 기본적으로 AWS IAM Identity Center 및 Amazon S3 Access Grants와 통합됩니다. 즉, 인증된 사용자만 액세스 권한이 있는 데이터를 볼 수 있습니다. 웹 앱은 [Amazon S3용 Storage Browser](#)를 사용하여 빌드되며 코드를 작성하거나 자체 애플리케이션을 호스팅할 필요 없이 완전 관리형 제품에서 동일한 최종 사용자 기능을 제공합니다.

Transfer Family 웹 앱과 함께 AWS 서비스 사용하는 다른에 대한 자세한 내용은 다음 설명서를 참조하세요.

- [Amazon Simple Storage Service 사용 설명서의 S3 Access Grants로 액세스 관리](#)
- [AWS IAM Identity Center 사용 설명서](#)
- [Amazon S3 Access Grants 워크숍](#)
- [완전 관리형 Amazon S3 파일 전송을 위한 AWS Transfer Family 웹 앱 발표](#)

Transfer Family 웹 앱을 시작하는 데 도움이 되는 다음 리소스를 사용할 수 있습니다.

- 사용 설명서는 Transfer Family 웹 앱 설정에 대한 자세한 step-by-step 안내를 제공합니다. [자습서: 기본 Transfer Family 웹 앱 설정](#).
- AWS 시작하기 리소스 센터는 웹 [AWS Transfer Family 앱 시작하기](#) 자습서를 제공합니다.
- [웹 앱 시작하기를 AWS Transfer Family](#)봅니다.

AWS 리전 Transfer Family 웹 앱용

AWS Transfer Family 웹 앱은 아시아 태평양(뉴질랜드) 및를 제외하고 리전 [AWS 표에 나열된 대로 Transfer Family가 지원하는 모든 리전](#)에서 사용할 수 있습니다 AWS European Sovereign Cloud.

웹 애플리케이션 VPC 엔드포인트는 웹 애플리케이션을 사용할 수 있는 모든 AWS 리전에서 지원됩니다.

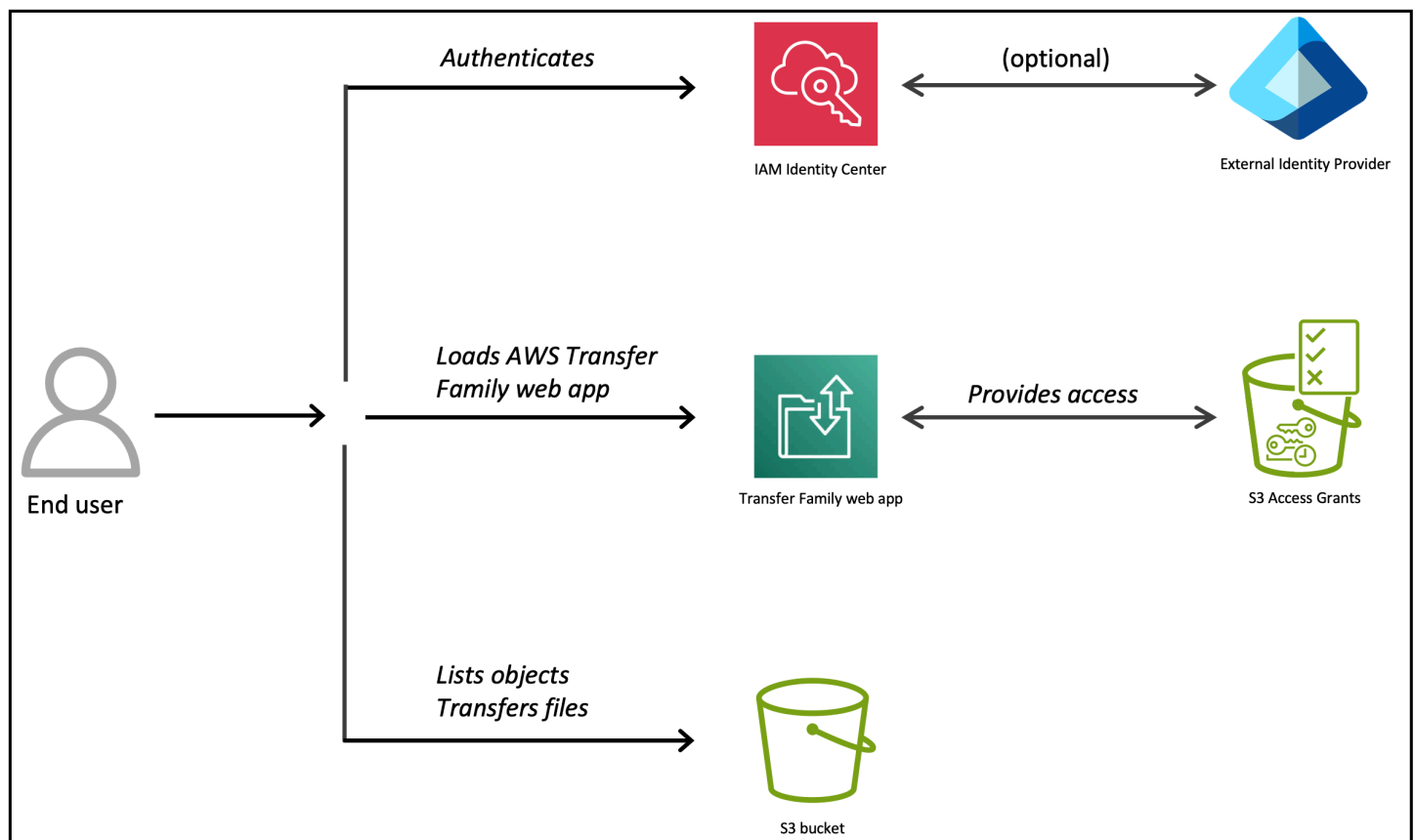
AWS Transfer Family 웹 애플리케이션의 브라우저 호환성

Transfer Family 웹 애플리케이션은 다음 브라우저를 지원합니다.

브라우저	버전	호환성
Microsoft Edge	최근 세 버전	호환됨
Mozilla Firefox	최근 세 버전	호환됨
Google Chrome	최근 세 버전	호환됨
Apple Safari	최근 세 버전	호환됨

Transfer Family 웹 애플리케이션을 생성하는 방법

다음 다이어그램은 Transfer Family 웹 애플리케이션 아키텍처를 보여줍니다.



다이어그램을 기반으로 Transfer Family 웹 앱이 다음과 상호 작용하는지 확인할 수 있습니다. AWS 서비스

- Amazon S3 for storage 및 Amazon S3 Access Grants를 사용하여 세션 자격 증명을 획득합니다.
- AWS IAM Identity Center 를 페더레이션 자격 증명 공급자로 지정합니다.
- 웹 앱에 대한 사용자 지정 URL을 구성하는 경우 Amazon CloudFront.

웹 앱을 사용할 때 다음과 같은 제한 사항에 유의하세요.

- 쿼리당 최대 검색 결과 수: 10,000
- Transfer Family 웹 앱에서 사용하는 Amazon S3 버킷은 웹 앱 자체와 동일한 계정에 있어야 합니다. 교차 계정 버킷은 현재 지원되지 않습니다.
- 쿼리당 최대 검색 범위: 검색된 파일 10,000개
- 파일당 최대 업로드 크기: 160GB(149GiB)
- 복사를 위한 최대 크기 파일: 5.36GB(5GiB)
- 점(.)으로 시작하거나 끝나는 폴더 이름은 지원되지 않습니다.

사전 조건

에서 필요한 역할을 AWS Identity and Access Management 구성합니다. 지침에 제공된 코드 블록을 붙여 넣습니다. 필요한 역할 구성에 대한 자세한 내용은 섹션을 참조하세요 [Transfer Family 웹 앱에 대한 IAM 역할 구성](#).

- 자격 증명 보유자 역할을 생성합니다.
- S3 Access Grants에서 사용할 IAM 역할을 생성합니다. S3 Access Grants는 이 IAM 역할을 수입하여 등록된 Amazon S3 위치의 피부여자에게 임시 자격 증명을 제공합니다.

Transfer Family 웹 앱을 생성하는 프로세스

웹 앱을 생성하고 최종 사용자를 시작하고 실행하려면 다음 작업을 수행합니다.

1. 페더레이션 ID 제공업체 역할을 하도록 IAM Identity Center를 구성합니다. IAM Identity Center에서 다음 작업을 수행합니다. IAM Identity Center 구성에 대한 자세한 내용은 섹션을 참조하세요 [Transfer Family 웹 앱에 대한 자격 증명 공급자 구성](#).
 - a. IAM Identity Center 인스턴스가 아직 없는 경우 인스턴스를 생성합니다.

- b. 자격 증명 소스를 확인합니다. 기본 IAM Identity Center 디렉터리 또는 타사 공급자(예: Okta)일 수 있습니다.
 - c. 웹 앱을 사용할 사용자 또는 그룹을 생성하거나 식별합니다.
 - d. 자격 증명 소스에 IAM Identity Center 디렉터리를 사용하는 경우 생성한 사용자 또는 그룹 IDs 기록해 둡니다. 나중에 S3 Access Grants를 사용하여 액세스 권한을 생성할 때 필요합니다.
2. Amazon S3에서 Amazon S3 Access Grants를 구성합니다. S3 Access Grants에 대한 자세한 내용은 [Transfer Family 웹 앱에 대한 Amazon S3 Access Grants 구성](#) 섹션을 참조하세요.
 - S3 Access Grants 인스턴스가 아직 없는 경우 해당 인스턴스를 생성합니다 AWS 리전.
 - IAM 역할을 사용하여 위치를 등록합니다.
 - 액세스 권한 부여를 생성합니다.
3. Transfer Family에서 다음 작업을 수행합니다.
 - a. Transfer Family 웹 앱을 생성합니다. Transfer Family 웹 앱을 생성하는 방법에 대한 자세한 내용은 섹션을 참조하세요 [Transfer Family 웹 앱 구성](#).

Important

웹 앱에서 사용하는 모든 Amazon S3 버킷에 대해 교차 오리진 리소스 공유(CORS)를 설정합니다. CORS 설정에 대한 자세한 내용은 섹션을 참조하세요 [버킷에 대한 교차 오리진 리소스 공유\(CORS\) 설정](#).

- b. 웹 앱에 사용자 또는 그룹을 할당합니다. 사용자 및 그룹을 할당하는 방법에 대한 자세한 내용은 섹션을 참조하세요 [Transfer Family 웹 앱에 사용자 또는 그룹 할당 또는 추가](#).
- c. (선택 사항) 웹 앱의 액세스 엔드포인트를 사용자 지정 URL로 업데이트합니다. 사용자 지정 URL 생성에 대한 자세한 내용은 섹션을 참조하세요 [사용자 지정 URL로 액세스 엔드포인트 업데이트](#).
- d. 최종 사용자에게 웹 앱에 로그인하고 상호 작용할 수 있는 액세스 엔드포인트 URL을 제공합니다.

Transfer Family 웹 앱에 대한 자격 증명 공급자 구성

다음 섹션에서는 자격 증명 공급자를 구성하는 방법을 설명합니다.

시작하려면 자격 증명 소스가 있어야 합니다. IAM Identity Center 디렉터리, AWS Directory Service for Microsoft Active Directory 또는 외부 자격 증명 공급자를 사용할 수 있습니다. Transfer Family는 사용자 자격 증명을 저장하고 여러 조직의 사용자를 인증하는 시스템인 연합 자격 증명 공급자로 IAM Identity Center를 사용합니다.

IAM Identity Center 디렉터리를 ID 소스로 사용하지 않는 경우 다음 주제를 참조하세요.

- [외부 자격 증명 공급자 관리](#)
- [Microsoft AD 디렉터리에 연결](#)
- [IAM Identity Center의 조직 및 계정 인스턴스](#)
- [IAM Identity Center ID 소스 자습서](#)

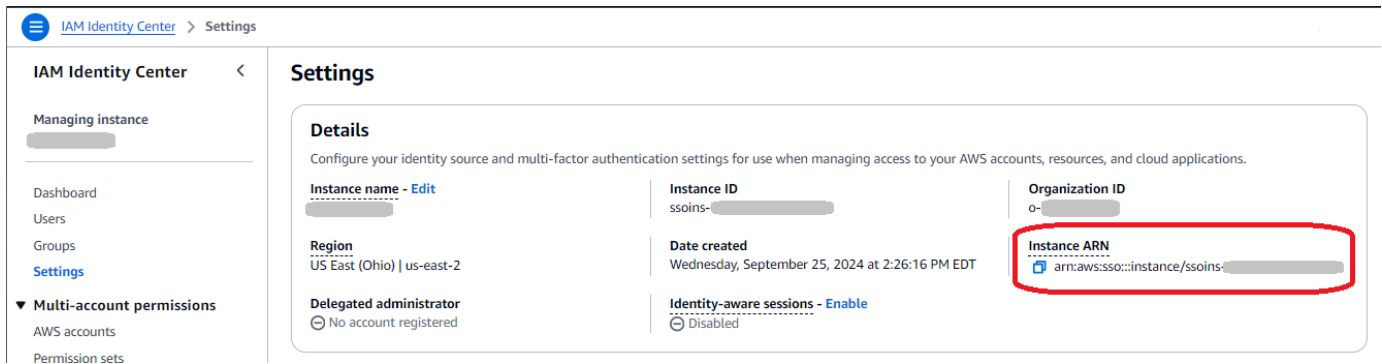
Note

IAM Identity Center에는 인스턴스당 1개의 ID 소스만 가질 수 있습니다 AWS 리전. 자세한 내용은 [IAM Identity Center 사전 조건 및 고려 사항을 참조하세요](#).

IAM Identity Center 디렉터리를 ID 소스로 사용하려는 경우 이 주제를 건너뛰고 로 이동하여 마법사에서 IAM Identity Center 인스턴스를 [Transfer Family 웹 앱 생성](#) 생성할 수 있습니다.

Transfer Family 웹 앱과 함께 사용하도록 AWS IAM Identity Center 를 구성하려면

1. 에 로그인 AWS Management Console 하고 <https://console.aws.amazon.com/singlesignon/> AWS IAM Identity Center 콘솔을 엽니다.
2. 계정 인스턴스 또는 조직 인스턴스를 생성하고 사용할 수 있습니다 AWS IAM Identity Center.
 - 계정 인스턴스에 대한 자세한 내용은 [의 계정 인스턴스 생성을 AWS IAM Identity Center](#) 참조하세요. IAM Identity Center의 계정 인스턴스를 사용하면 지원되는 AWS 관리형 애플리케이션과 OpenID Connect(OIDC) 기반 고객 관리형 애플리케이션을 배포할 수 있습니다. 계정 인스턴스는 IAM Identity Center 인력 ID 및 액세스 포털 기능을 AWS 계정 활용하여 단일에서 애플리케이션의 격리된 배포를 지원합니다.
 - 조직 인스턴스에 대한 자세한 내용은 [IAM Identity Center의 조직 인스턴스](#)를 참조하세요. 단일 조직 인스턴스를 사용하여 사용자 및 그룹의 액세스를 중앙에서 관리할 수 있습니다.
3. IAM Identity Center 설정 페이지에서 인스턴스 ARN을 기록해 둡니다. Amazon S3 Access Grant 인스턴스를 생성할 때 이 값이 필요합니다.



4. Transfer Family 웹 앱과 함께 사용할 사용자 및 선택적으로 그룹을 하나 이상 생성합니다. IAM Identity Center 디렉터리를 자격 증명 공급자로 사용하는 경우 웹 앱 자체에서 직접 사용자를 추가할 수도 있습니다. 자세한 내용은 [Transfer Family 웹 앱에 사용자 또는 그룹 할당 또는 추가](#) 단원을 참조하십시오.

Transfer Family 웹 앱에 대한 IAM 역할 구성

두 가지 역할이 필요합니다. 하나는 웹 앱의 자격 증명 보유자 역할로 사용하고 다른 하나는 액세스 권한 부여를 구성하는 데 사용합니다. 자격 증명 보유자 역할은 인증된 사용자의 자격 증명을 세션에 포함하는 역할입니다. 사용자를 대신하여 데이터 액세스를 위해 S3 Access Grants에 요청하는 데 사용됩니다.

Note

자격 증명 보유자 역할을 생성하는 절차를 건너뛸 수 있습니다. Transfer Family 서비스가 자격 증명 보유자 역할을 생성하도록 하는 방법에 대한 자세한 내용은 섹션을 참조하세요 [Transfer Family 웹 앱 생성](#).

액세스 권한 부여 역할 생성 절차를 건너뛸 수 있습니다. 액세스 권한 부여 생성 절차에서 S3 위치를 등록하는 단계에서 새 역할 생성을 선택합니다.

자격 증명 보유자 역할 생성

1. 에 로그인 AWS Management Console 하고 <https://console.aws.amazon.com/iam/> IAM 콘솔을 엽니다.
2. 역할을 선택한 다음 역할 생성을 선택합니다.
3. 사용자 지정 신뢰 정책을 선택한 다음 다음 코드를 붙여 넣습니다.

```
{
```

```

    "Version": "2012-10-17",
    "Statement": [
      {
        "Effect": "Allow",
        "Principal": {
          "Service": "transfer.amazonaws.com"
        },
        "Action": [
          "sts:AssumeRole",
          "sts:SetContext"
        ]
      }
    ]
  }
}

```

4. 다음을 선택한 다음 권한 추가를 건너뛰고 다음을 다시 선택합니다.
5. 와 같은 이름을 입력합니다 web-app-identity-bearer.
6. 역할 생성을 선택하여 자격 증명 보유자 역할을 생성합니다.
7. 목록에서 방금 생성한 역할을 선택한 다음 권한 정책 패널에서 권한 추가 > 인라인 정책 생성을 선택합니다.
8. 정책 편집기에서 JSON을 선택한 다음 다음 코드 블록에 붙여 넣습니다.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetDataAccess",
        "s3:ListCallerAccessGrants",
        "s3:ListAccessGrantsInstances"
      ],
      "Resource": "*"
    }
  ]
}

```

9. 정책 이름에 AllowS3AccessGrants를 입력한 다음 정책 생성을 선택합니다.

다음으로 S3 Access Grants가 피부여자에게 임시 자격 증명을 제공하기 위해 수임하는 역할을 생성합니다.

Note

서비스가 자격 증명 보유자 역할을 생성하도록 허용하는 경우 해당 역할은 혼동된 대리자 보호를 설정합니다. 따라서 코드는 여기에 표시된 것과 다릅니다.

액세스 권한 부여 역할 생성

1. 에 로그인 AWS Management Console 하고 <https://console.aws.amazon.com/iam/> IAM 콘솔을 엽니다.
2. 역할을 선택한 다음 역할 생성을 선택합니다. 이 역할에는의 S3 데이터에 액세스할 수 있는 권한이 있어야 합니다 AWS 리전.
3. 사용자 지정 신뢰 정책을 선택한 다음 다음 코드를 붙여 넣습니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "access-grants.s3.amazonaws.com"
      },
      "Action": [
        "sts:AssumeRole",
        "sts:SetContext"
      ]
    }
  ]
}
```

4. 다음 [위치 등록에 설명된 대로 최소 정책 추가](#)를 선택합니다. 권장되지는 않지만 AmazonS3FullAccess 관리형 정책을 추가할 수 있으며, 이는 요구 사항에 너무 허용적일 수 있습니다.
5. 다음을 선택하고 이름(예: access-grants-location)을 입력합니다.
6. 역할 생성을 선택하여 역할을 생성합니다.

Note

서비스가 액세스 권한 부여 역할을 생성하도록 허용하는 경우 해당 역할은 혼동된 대리자 보호를 설정합니다. 따라서 코드는 여기에 표시된 것과 다릅니다.

Transfer Family 웹 앱 구성

이 섹션에서는 Transfer Family 웹 앱을 생성하는 절차를 설명합니다. 사용할 수 있는 사용자 및 그룹을 할당하려면 섹션을 참조하세요 [Transfer Family 웹 앱에 사용자 또는 그룹 할당 또는 추가](#).

Note

이 절차를 반복하여 웹 앱을 추가합니다. 이전에 생성한 IAM 역할을 재사용할 수 있습니다. 각 버킷의 교차 오리진 리소스 공유(CORS) 정책에 새 웹 앱의 액세스 엔드포인트를 추가해야 합니다.

Transfer Family 웹 앱 생성

Note

자격 증명 공급자에 IAM Identity Center 디렉터리를 사용하지 않는 경우에 설명된 대로 IAM Identity Center를 이미 설정하고 타사 자격 증명 공급자를 구성해야 웹 앱을 만들 수 있습니다 [Transfer Family 웹 앱에 대한 자격 증명 공급자 구성](#).

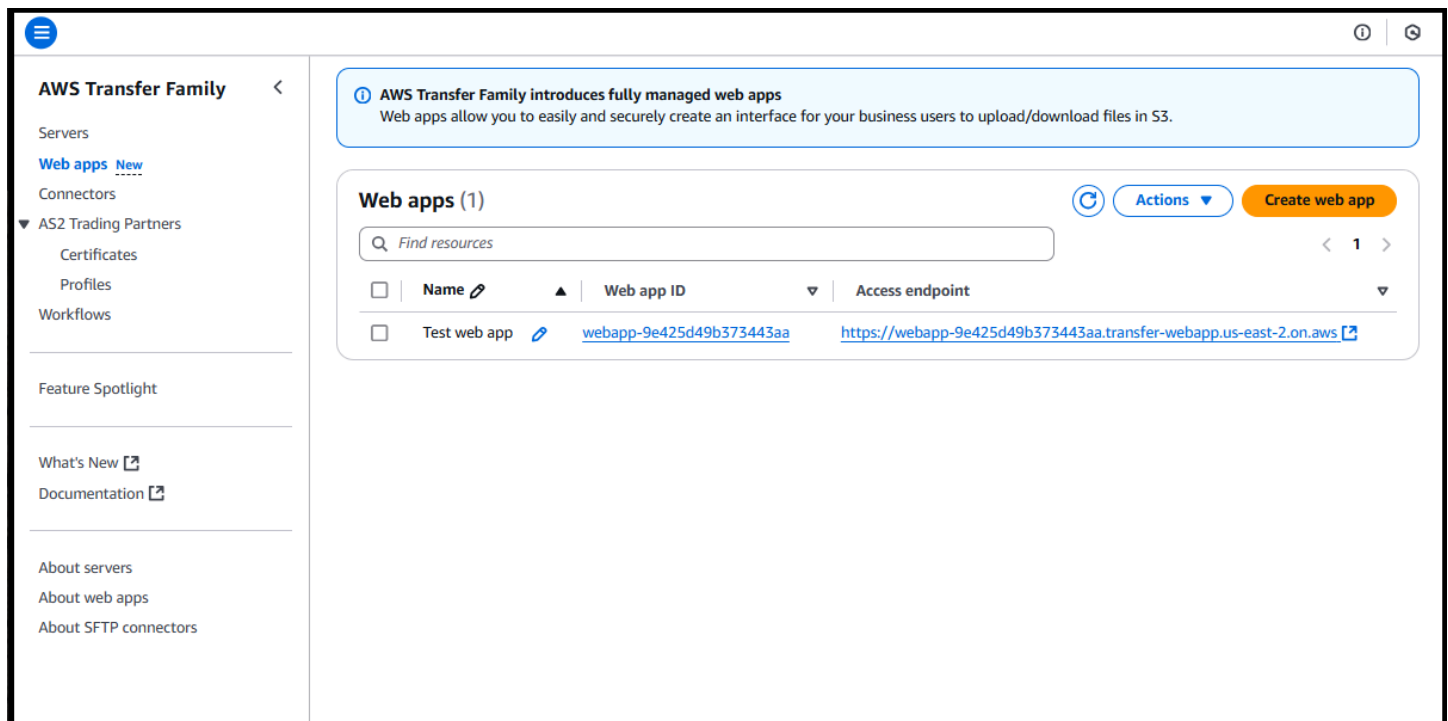
Transfer Family 웹 앱을 생성하려면 다음 단계를 완료하세요.

Transfer Family 웹 앱을 생성하려면

1. 에 로그인 AWS Management Console 하고 <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 웹 앱을 선택합니다.
3. 웹 앱 생성을 선택합니다.

인증 액세스의 경우 창은 다음과 같이 채워집니다.

- 에서 조직 또는 계정 인스턴스를 이미 생성한 경우 애플리케이션이 IAM Identity Center의 계정 인스턴스에 AWS Transfer Family 연결되었다는 메시지가 AWS IAM Identity Center 표시됩니다.
 - 이미 계정 인스턴스가 있고 조직 인스턴스의 멤버인 경우 연결할 인스턴스를 선택할 수 있습니다.
 - 아직 계정 인스턴스가 없거나 조직 인스턴스의 멤버인 경우 계정 인스턴스를 생성하는 옵션이 표시됩니다.
4. 엔드포인트 타입으로는 공적으로 액세스 가능 엔드포인트 타입을 선택합니다. VPC 호스팅 엔드포인트에 대한 내용은 [VPC에서 Transfer Family 웹 앱 생성](#) 섹션을 참조하세요.
 5. 권한 유형 창에서 이전에 생성한 역할을 사용하거나 서비스가 역할을 생성하도록 할 수 있습니다.
 - 자격 증명 보유자 역할을 이미 생성한 경우 기존 역할 사용을 선택하고 기존 역할 선택 메뉴에서 역할을 선택합니다.
 - 서비스가 역할을 생성하도록 하려면 새 서비스 역할 생성 및 사용을 선택합니다.
 6. 웹 앱 단위 창에서 값을 선택합니다. 하나의 웹 앱 단위는 최대 250개의 고유한 세션에서 웹 앱 활동을 허용합니다. 웹 앱을 생성할 때 예상 피크 워크로드 볼륨에 따라 필요한 단위 수를 프로비저닝합니다. 웹 앱 단위를 변경하면 결제에 영향을 미칩니다. 요금에 대한 자세한 내용은 [AWS Transfer Family 요금](#)을 참조하세요.
 7. 에서 Transfer Family를 사용하는 경우 FIPS 활성화 창에서 FIPS 활성화 엔드포인트 확인란을 선택할 AWS GovCloud (US) Region 수 있습니다. 다른 모든 경우에는 AWS 리전이 옵션을 사용할 수 없습니다.
 8. (선택 사항) 웹 앱을 구성하는 데 도움이 되는 태그를 추가합니다. 이름이 키이고 설명이 포함된 이름이 값으로 있는 태그를 추가하는 것이 좋습니다.
 9. 다음을 선택합니다. 이 화면에서 웹 앱의 제목을 선택적으로 제공할 수 있습니다. 제목을 제공하지 않으면 Transfer Web App의 기본 제목이 제공됩니다. 로고 및 파비콘의 이미지 파일을 업로드할 수도 있습니다.
 10. 다음을 선택한 다음 웹 앱 생성을 선택합니다.



Note

웹 앱 엔드포인트에서 액세스하는 모든 버킷에 대해 교차 오리진 리소스 공유(CORS) 정책을 설정해야 합니다.

IP 주소 지정

Transfer Family 웹 앱은 듀얼 스택 엔드포인트를 사용하여 IPv4 및 IPv6 연결을 모두 지원합니다. 인증을 위해 웹 앱 AWS IAM Identity Center은 이중 스택 엔드포인트를 통해 IPv6도 지원하는를 사용합니다.

듀얼 스택 IAM Identity Center 엔드포인트 지원이 시작된 경우 계정에 리전에 웹 앱이 있는 경우 해당 계정 및 리전의 모든 웹 앱은 이전 버전과의 호환성을 위해 IPv4-only IAM Identity Center 엔드포인트를 계속 사용합니다. 이는 기존 웹 앱과 새로 생성된 웹 앱 모두에 적용됩니다. 시작 시 웹 앱이 없는 계정은 IAM Identity Center 듀얼 스택 엔드포인트를 사용합니다. 웹 앱에서 사용하는 IAM Identity Center 엔드포인트를 변경해야 하는 경우 AWS Support에 문의하세요.

Note

웹 앱이 사용하는 엔드포인트를 확인하려면 IAM Identity Center 로그인 페이지로 리디렉션될 때 URL을 검사합니다.

- IPv4-only: [Region].signin.aws.amazon.com
- 듀얼 스택: [Region].sso.signin.aws

조직에서 방화벽 또는 네트워크 게이트웨이를 사용하는 경우 기존 IPv4 및 새 듀얼 스택 IAM Identity Center 엔드포인트를 모두 허용하여 웹 앱이 사용하는 엔드포인트에 관계없이 중단 없는 액세스를 보장합니다. 허용 목록에 추가할 도메인 및 URL 엔드포인트의 전체 목록은 IAM Identity Center 사용 설명서 [의 AWS 액세스 포털에 대한 액세스를 허용하도록 방화벽 및 게이트웨이 업데이트를 참조하세요](#).

VPC에서 호스팅되는 웹 앱의 경우 VPC 엔드포인트의 IP 주소 유형인 IPv4 전용 또는 듀얼 스택(IPv4 및 IPv6)을 선택할 수 있습니다. 기본값은 이전 버전과의 호환성을 위한 듀얼 스택입니다. VPC가 IPv4 IPv4-only를 선택합니다. IPv6 생성 후 웹 앱을 업데이트하여 IP 주소 유형을 변경할 수 있습니다.

VPC에서 Transfer Family 웹 앱 생성

이 섹션에서는 VPC에서 Transfer Family 웹 앱을 생성하는 절차를 설명합니다. 퍼블릭 인터넷을 통하지 않고 Amazon S3 버킷과 데이터를 주고받는 데 사용할 수 있도록 Virtual Private Cloud(VPC) 내에서 웹 앱의 엔드포인트를 호스팅할 수 있습니다. 웹 앱을 사용할 수 있는 사용자 및 그룹을 할당하려면 섹션을 참조하세요 [Transfer Family 웹 앱에 사용자 또는 그룹 할당 또는 추가](#).

Note

Transfer Family 웹 앱 VPC 엔드포인트를 사용할 때 프라이빗 end-to-end 데이터 흐름을 보장하려면 세 가지 추가 구성 요소를 구현해야 합니다. 먼저 Amazon S3 Access Grants API 호출에 필요한 Amazon S3 Control API 작업에 대한 PrivateLink 엔드포인트를 설정합니다. 둘째, PrivateLink Amazon S3 Gateway 엔드포인트(VPC 내 트래픽의 경우) 또는 Amazon S3 인터페이스 엔드포인트(VPN 또는 Direct Connect를 통해 온프레미스 네트워크에서 오는 트래픽의 경우)를 사용하여 Amazon S3 데이터 액세스를 위한 엔드포인트를 구성합니다. 셋째, 이러한 VPC 엔드포인트로부터의 트래픽만 허용하도록 버킷 정책을 업데이트하여 Amazon S3 버킷 액세스를 잠급니다. 이 조합을 통해 모든 데이터 전송이 프라이빗 네트워크 인프라 내에 유지되고 퍼블릭 인터넷을 통과하지 않도록 할 수 있습니다.

Transfer Family 웹 앱 생성

사전 조건

- AWS IAM Identity Center 구성된 자격 증명 공급자로 설정합니다. [Transfer Family 웹 앱에 대한 자격 증명 공급자 구성](#)(를) 참조하세요.
- VPC 및 네트워킹 구성 요소가 설정되었습니다. [VPC 생성](#)을 참조하세요.
- Amazon S3 Control 작업을 위한 API 엔드포인트 설정. [Amazon S3 인터페이스 엔드포인트 액세스](#)를 참조하세요.
- Amazon S3(게이트웨이 또는 인터페이스)용 VPC 엔드포인트 설정. [Amazon S3용 VPC 엔드포인트 유형](#)을 참조하세요. 인터페이스 엔드포인트를 사용하는 경우 프라이빗 DNS를 활성화해야 합니다. 예제는를 사용하여 [Amazon S3에 대한 프라이빗 DNS 지원 소개](#)를 AWS PrivateLink참조하세요.

Note

AWS IAM Identity Center 는 VPC 엔드포인트를 지원하지 않습니다. 모든 인증 요청은 퍼블릭 인터넷을 전송합니다. 또한 Transfer Family 웹 애플리케이션은 정적 콘텐츠(예: JavaScript, CSS 및 HTML 파일)를 로드하려면 인터넷 액세스가 필요합니다. 퍼블릭 인터넷 액세스에 대한 요구 사항은 데이터 액세스와 별개입니다. VPC 엔드포인트는 VPC 인프라를 통해 연결이 라우팅되도록 합니다.

Transfer Family 웹 앱을 생성하려면

1. 에 로그인 AWS Management Console 하고 <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 웹 앱을 선택합니다.
3. 웹 앱 생성을 선택합니다. 인증 액세스의 경우 창은 다음과 같이 채워집니다.
 - 에서 조직 또는 계정 인스턴스를 이미 생성한 경우 애플리케이션이 IAM Identity Center의 계정 인스턴스에 AWS Transfer Family 연결되었다는 메시지가 AWS IAM Identity Center표시됩니다.
 - 이미 계정 인스턴스가 있고 조직 인스턴스의 멤버인 경우 연결할 인스턴스를 선택할 수 있습니다.
 - 아직 계정 인스턴스가 없거나 조직 인스턴스의 멤버인 경우 계정 인스턴스를 생성하는 옵션이 표시됩니다.
4. 엔드포인트 구성 섹션에서 사용자가 웹 앱에 액세스하는 방법을 선택합니다.
 - 퍼블릭 액세스: HTTPS를 통해 퍼블릭을 통해 웹 앱 엔드포인트에 액세스할 수 있습니다. 이 옵션에는 VPC 구성이 필요하지 않으므로 설정이 간단하며 광범위한 퍼블릭 사용을 위한 애플리케이션에 적합합니다.

- VPC 호스팅: 웹 앱 엔드포인트는 Virtual Private Cloud(VPC) 내에서 호스팅되어 VPC 네트워크를 통한 프라이빗 네트워크 액세스 AWS Direct Connect 또는 VPN 연결을 제공합니다. 이 옵션은 네트워크 격리를 통해 향상된 보안을 제공하며 내부 애플리케이션에 권장됩니다.

VPC 호스팅 엔드포인트를 구성할 때 다음을 지정해야 합니다.

- VPC: 기존 VPC를 선택하거나 새 VPC를 생성합니다. VPC 생성 버튼을 사용할 수 있습니다.
- IP 주소 유형: VPC 엔드포인트의 IP 주소 유형을 선택합니다. IPv4 IPv4-only를 선택하고 IPv4 및 IPv6 연결의 경우 듀얼 스택을 선택합니다. 기본값은 듀얼 스택입니다.
- 가용 영역: 엔드포인트를 배포할 가용 영역을 선택합니다.
- 서브넷: 선택한 각 가용 영역 내에서 서브넷을 선택합니다.
- 보안 그룹: 소스 IP 주소를 기반으로 액세스를 제어할 보안 그룹을 선택하거나 생성합니다. 지정하지 않으면 VPC의 기본 보안 그룹이 사용됩니다. VPC 콘솔을 통해 보안 그룹을 관리합니다. TCP 포트 443에서 HTTPS를 통해 네트워크에서 들어오는 인바운드 트래픽을 허용하도록 VPC 보안 그룹을 구성합니다. 이는 IAM Identity Center 인증 및 웹 앱 정적 콘텐츠 로드에도 필요합니다.

Note

VPC 엔드포인트에 대해 액세스 엔드포인트를 사용자 지정할 수 없습니다. 사용자 지정 URL을 추가하려면 퍼블릭 엔드포인트를 사용합니다.

생성 후 단계

- 웹 앱 엔드포인트에서 액세스하는 모든 버킷에 대해 교차 오리진 리소스 공유(CORS) 정책을 설정해야 합니다. [CORS\(Cross-Origin Resource Sharing\) 정책](#)을(를) 참조하세요.
- VPC 엔드포인트를 통해 VPC에서 시작되는 트래픽만 허용하도록 버킷 정책을 업데이트합니다. [특정 VPC 엔드포인트에 대한 액세스 제한](#)을(를) 참조하세요.
- Transfer Family 웹 앱에 사용자 또는 그룹을 할당하거나 추가합니다. [Transfer Family 웹 앱에 사용자 또는 그룹 할당 또는 추가](#)을(를) 참조하세요.

CORS(Cross-Origin Resource Sharing) 정책

웹 앱에서 사용하는 모든 버킷에 대해 교차 오리진 리소스 공유(CORS)를 설정해야 합니다. CORS에 대한 자세한 내용은 [버킷에 대한 교차 오리진 리소스 공유\(CORS\) 설정](#) 섹션을 참조하십시오.

⚠ Important

다음 예제 정책을 사용하기 전에 허용된 오리진을 액세스 엔드포인트로 바꿉니다. 그렇지 않으면 최종 사용자가 웹 앱의 위치에 액세스하려고 할 때 오류가 발생합니다.

정책 예제:

```
[
  {
    "AllowedHeaders": [
      "*"
    ],
    "AllowedMethods": [
      "GET",
      "PUT",
      "POST",
      "DELETE",
      "HEAD"
    ],
    "AllowedOrigins": [
      "https://vpce-1234567-example.vpce-mq.transfer-webapp.us-east-1.on.aws"
    ],
    "ExposeHeaders": [
      "last-modified",
      "content-length",
      "etag",
      "x-amz-version-id",
      "content-type",
      "x-amz-request-id",
      "x-amz-id-2",
      "date",
      "x-amz-cf-id",
      "x-amz-storage-class",
      "access-control-expose-headers"
    ],
    "MaxAgeSeconds": 3000
  }
]
```

특정 VPC 엔드포인트에 대한 액세스 제한

다음은 ID가 vpce-1a2b3c4d인 VPC 엔드포인트에서만 특정 버킷 amzn-s3-demo-bucket에 대한 액세스를 제한하는 Amazon S3 버킷 정책의 예제입니다. 지정된 엔드포인트가 사용되지 않으면 이 정책은 버킷에 대한 모든 액세스를 거부합니다. aws:SourceVpce 조건은 엔드포인트를 지정합니다. aws:SourceVpce 조건은 VPC 엔드포인트 리소스의 ARN을 요구하지 않고 VPC 엔드포인트 ID만 요구합니다. VPC에서 발생하는 트래픽만 허용하도록 버킷 정책을 업데이트하는 방법에 대한 자세한 내용은 [버킷 정책을 사용하여 VPC 엔드포인트에서 액세스 제어를](#) 참조하세요. 정책에서 조건 사용에 대한 자세한 내용은 [조건 키를 사용한 버킷 정책 예제](#)를 참조하세요. 이 정책을 적용하기 위한 사전 조건으로 [Amazon S3 VPC 엔드포인트](#)를 생성해야 합니다.

Important

다음 예제 정책을 사용하기 전에 VPC 엔드포인트 ID를 사용 사례에 적합한 값으로 바꾸십시오. 그렇지 않으면 버킷에 액세스할 수 없습니다.

```
{
  "Version": "2012-10-17",
  "Id": "Policy1415115909152",
  "Statement": [
    {
      "Sid": "Access-to-specific-VPCE-only",
      "Principal": "*",
      "Action": "s3:*",
      "Effect": "Deny",
      "Resource": [
        "arn:aws:s3:::amzn-s3-demo-bucket",
        "arn:aws:s3:::amzn-s3-demo-bucket/*"
      ],
      "Condition": {
        "StringNotEquals": {
          "aws:SourceVpce": "vpce-1a2b3c4d"
        }
      }
    }
  ]
}
```

Transfer Family 웹 앱에 사용자 또는 그룹 할당 또는 추가

Transfer Family 웹 앱을 생성한 후 웹 앱에 액세스할 수 있는 사용자와 그룹을 할당할 수 있습니다. 이미 생성되어 IAM Identity Center에 저장된 사용자를 검색하거나 [새 사용자를 직접 추가할](#) 수 있습니다 (IAM Identity Center 디렉터리를 자격 증명 공급자로 사용하는 경우). 새 사용자를 추가하면 해당 사용자도 IAM Identity Center 인스턴스에 추가됩니다.

다음 사항에 유의하세요.

- IAM Identity Center 디렉터리를 ID 소스로 사용하고 적절한 권한이 있는 경우에만 새 사용자를 추가할 수 있습니다. 조직 인스턴스의 멤버인 경우 사용자를 추가하는 데 필요한 권한이 없을 수 있습니다.

Note

애플리케이션에 사용자 또는 그룹을 할당하지 않으면 사용자가 웹 앱에 로그인하려고 할 때 오류가 발생합니다.

- 새 사용자를 생성하는 경우 웹 앱의 데이터에 액세스할 수 있도록이 사용자에게 대한 S3 액세스 권한도 생성해야 합니다.
- 새 사용자를 생성한 후 해당 사용자는 IAM Identity Center로부터 진행 방법에 대한 지침이 포함된 온보딩 이메일을 받습니다.

Transfer Family 웹 앱에 사용자를 할당하려면

1. 웹 앱 목록으로 이동하여 편집할 목록을 선택합니다.
2. 사용자 및 그룹 할당을 선택합니다.

The screenshot shows the AWS IAM Identity Center console for a web app named 'webapp-'. At the top, there are buttons for 'Assign users and groups' (orange), 'Clear customization', and 'Delete'. Below this is the 'Web app details' section, which includes an 'Access endpoint' (a URL to the transfer-webapp.us-east-2.on.aws endpoint), a 'Web app endpoint' (a URL to the webapp.us-east-2.on.aws endpoint), and 'Web app units' (1). To the right of these details are the 'Identity provider' (AWS IAM Identity Center), 'IAM role', and 'Instance ARN'. Below the details section are tabs for 'Users' and 'Groups'. The 'Users' tab is selected, showing 'Users (2)'. Below this is a search bar labeled 'Find resources' and a 'Delete' button. A table lists two users with columns for 'Display name', 'User ID', and 'Username'. Each user entry has a checkbox on the left and a link icon on the right.

3. IAM Identity Center에서 이전에 생성한 사용자를 할당하려면 기존 사용자 및 그룹 할당을 선택합니다. 새 사용자를 생성하려면 4단계로 건너뛩니다.
 - a. 정보 화면이 나타납니다. 시작하기를 선택하여 계속 진행합니다.
 - b. 사용자를 검색합니다. 검색 기준을 입력하기 전까지는 사용자가 표시되지 않습니다. 다른 경우 사용자 이름이 아닌 표시 이름으로 검색해야 합니다. 정확한 일치 항목만 반환됩니다. 사용자를 찾을 수 없는 경우 IAM Identity Center 관리 콘솔로 이동하여 사용자를 찾은 다음 여기에 표시 이름을 복사하여 붙여 넣습니다.

Assign users and groups

Search: jane

Use "jane"

Users
Jane Doe jane@doe jane@doe@example.com

Choose to assign users and groups

[Manage users and groups from IAM Identity Center](#)

Back Add new users Done

- c. 추가할 사용자 및 그룹을 선택한 다음 할당을 선택합니다.
4. 새 사용자를 생성하려면 새 사용자 추가 및 할당을 선택합니다.
 - a. 정보 화면이 나타납니다. 시작하기를 선택하여 계속 진행합니다.
 - b. 새 사용자 추가를 선택합니다.
 - c. 대화 상자에 사용자 이름, 이름과 성, 이메일 주소 등의 사용자 세부 정보를 입력합니다.

Add new users

Newly-created users are enabled by default. To grant them access to Add User Widget, you must set their password in the IAM Identity Center console. [Manage user passwords in the IAM Identity Center console](#)

Username
A username is required for this person to sign in to the AWS access portal. You can't change the username later.

Maximum length of 128 characters. Can only contain alphanumeric characters or any of the following: + = , - @ _

First name

Last name

Email address

Confirm email address

Display name
This is typically the full name of the user (first and last name) and appears in the users list.

Cancel
Next

- d. 다음을 선택한 다음 추가를 선택하여 사용자를 추가하고 대화 상자를 닫거나 새 사용자 추가를 선택하여 다른 사용자를 생성합니다.

버킷에 대한 교차 오리진 리소스 공유(CORS) 설정

웹 앱에서 사용하는 모든 버킷에 대해 교차 오리진 리소스 공유(CORS)를 설정해야 합니다. CORS 구성은가 버킷에 액세스하도록 허용할 오리진을 식별하는 규칙을 정의하는 문서입니다. CORS에 대한 자세한 내용은 [CORS\(Cross-Origin Resource Sharing\) 구성](#)을 참조하세요.

⚠ Important

CORS를 설정하지 않으면 최종 사용자가 웹 앱의 위치에 액세스하려고 할 때 오류가 발생합니다.

Amazon S3 버킷에 대한 교차 오리진 리소스 공유(CORS)를 설정하려면

1. 에 로그인 AWS Management Console 하고 <https://console.aws.amazon.com/s3/> Amazon S3 콘솔을 엽니다.
2. 왼쪽 탐색 패널에서 버킷을 선택하고 검색 대화 상자에서 버킷을 검색한 다음 권한 탭을 선택합니다.
3. CORS(Cross-Origin Resource Sharing)에서 편집을 선택하고 다음 코드에 붙여 넣습니다.
*WebAppEndpoint*를 웹 앱의 실제 액세스 엔드포인트로 바꿉니다. 이는 웹 앱이 생성될 때 생성되는 VPC 호스팅 또는 퍼블릭 액세스 엔드포인트이거나, 생성할 경우 사용자 지정 액세스 엔드포인트일 수 있습니다. 사용자가 웹 앱에 로그인하려고 할 때 오류가 발생하므로 후행 슬래시를 입력하지 마십시오.

- 잘못된 예: `https://webapp-c7bf3423.transfer-webapp.us-east-2.on.aws/`
- 올바른 예제:
 - `https://webapp-c7bf3423.transfer-webapp.us-east-2.on.aws`
 - `https://vpce-05668789767a-fh45z079.vpce-mq.transfer-webapp.us-east-1.on.aws`

여러 웹 앱에 버킷을 재사용하는 경우 해당 엔드포인트를 AllowedOrigins 목록에 추가합니다.

```
[
  {
    "AllowedHeaders": [
      "*"
    ],
    "AllowedMethods": [
      "GET",
      "PUT",
      "POST",
      "DELETE",
      "HEAD"
    ]
  }
]
```

```

    "AllowedOrigins": [
      "https://WebAppEndpoint"
    ],
    "ExposeHeaders": [
      "last-modified",
      "content-length",
      "etag",
      "x-amz-version-id",
      "content-type",
      "x-amz-request-id",
      "x-amz-id-2",
      "date",
      "x-amz-cf-id",
      "x-amz-storage-class",
      "access-control-expose-headers"
    ],
    "MaxAgeSeconds": 3000
  }
]

```

4. 변경 사항 저장을 선택하여 CORS를 업데이트합니다.

CORS 구성을 테스트하려면 [CORS 테스트를](#) 참조하세요.

Transfer Family 웹 앱에 대한 Amazon S3 Access Grants 구성

이 주제에서는 Amazon S3 Access Grants를 사용하여 액세스 권한을 추가하는 방법을 설명합니다. 이 액세스 권한 부여는 회사 디렉터리의 사용자 및 그룹에 직접 데이터에 대한 액세스를 정의하고 권한 부여 just-in-time의 최소 권한 임시 자격 증명을 제공합니다. S3 Access Grants 인스턴스의 개별 권한 부여를 사용하면 회사 디렉터리의 특정 사용자 또는 그룹이 S3 Access Grants 인스턴스에 등록된 위치 내에서 액세스할 수 있습니다. 자세한 내용은 Amazon [S3 사용 설명서의 S3 Access Grants 개념](#)을 참조하세요. Amazon S3

Note

Transfer Family 웹 앱 이외의 S3 Access Grants에는 IAM Identity Center 디렉터리를 사용할 수 없습니다.

자격 증명 전파를 위해 Amazon S3 액세스 권한을 지정해야 합니다. Amazon S3 액세스 권한 부여는 최종 사용자가 액세스해야 하는 데이터를 저장합니다. 최종 사용자가 Transfer Family 웹 앱에 로그인 하면 S3 Access Grants는 사용자의 자격 증명을 신뢰할 수 있는 애플리케이션에 전달합니다. 이 섹션에서는 Amazon S3 액세스 권한 부여 인스턴스를 추가하고 구성한 다음 Amazon S3 버킷에 대한 액세스 권한을 부여하는 방법을 설명합니다.

Note

액세스 권한 설정을 완료하는 데 필요한 [IAM Identity Center 인스턴스 ARN](#)과 사용자 또는 그룹 ID를 준비합니다.

Amazon S3 Access Grants를 사용하여 권한 부여를 생성하려면

1. 에 로그인 AWS Management Console 하고 <https://console.aws.amazon.com/s3/> Amazon S3 콘솔을 엽니다.
2. 버킷을 생성하거나 웹 앱과 함께 사용할 기존 버킷을 기록해 둡니다. 버킷 생성에 대한 자세한 내용은 [Amazon S3 사용 설명서](#)를 참조하세요.
3. 왼쪽 탐색 창에서 Access Grants를 선택합니다.
4. S3 Access Grants 인스턴스 생성을 선택하고 다음 정보를 제공합니다.
 - **## ##**이 사용자의 리전인 리전에서 IAM Identity Center 인스턴스 추가# 선택합니다 AWS 리전. IAM Identity Center를 ID 제공업체로 사용하지 않는 경우 이 상자를 선택 취소합니다.
 - IAM Identity Center 인스턴스 ARN에 붙여 넣습니다.

[Amazon S3](#) > [Access Grants](#) > Set up Access Grants Instance

- Step 1
- Create S3 Access Grants instance**
- Step 2
- Register S3 Buckets or prefixes as locations
- Step 3
- Create grant
- Step 4
- Review and finish

Create S3 Access Grants instance Info

When you create your S3 Access Grants instance, you can use identities from your corporate directory or you can use IAM users and roles. This Access Grants instance serves as the container for your individual grants.

S3 Access Grants instance Info

AWS Region

You can create one S3 Access Grants instance per AWS Region per account.

US East (Ohio) us-east-2

☒ Add IAM Identity Center instance in US East (Ohio) us-east-2 - optional

You can use an IAM Identity Center instance to specify identities from corporate directories. If you don't have an existing IAM Identity Center instance, you can add one after creating your S3 Access Grants instance.

IAM Identity Center instance ARN

You can find the IAM Identity Center instance ARN in the [IAM Identity Center console](#).

arn:aws:sso::instance/ssoins-

Format: arn:aws:sso::instance/<instanceIdentifier>

다음을 선택하여 계속 진행합니다.

5. S3 버킷 또는 접두사를 위치로 등록합니다. 기본 위치인을 등록 `s3://` 하고 IAM 역할에 매핑하는 것이 좋습니다. 이 기본 경로의 위치는 계정의에 있는 모든 Amazon S3 버킷에 AWS 리전 대한 액세스를 포함합니다. 액세스 권한 부여를 생성할 때 범위를 기본 위치 내의 버킷, 접두사 또는 객체로 좁힐 수 있습니다.

다음 정보를 입력합니다.

- 범위에 버킷을 찾아보거나 버킷 이름과 선택적으로 접두사를 입력합니다.
- IAM 역할의 경우 새 역할 생성을 선택하여 서비스가 역할을 생성하도록 합니다.

또는에 설명된 대로 역할을 직접 생성한 [Transfer Family 웹 앱에 대한 IAM 역할 구성](#) 다음 여기에 해당 ARN을 입력할 수 있습니다.

The screenshot shows the 'Register location' page in the AWS IAM console. The breadcrumb trail is: Amazon S3 > Access Grants > US East (N. Virginia) us-east-1: default > Register location. The page title is 'Register location' with an 'Info' link. Below the title is a description: 'Registering a location for S3 your S3 Access Grants instance assigns an IAM role the permission to get temporary credentials to access resources within the specified scope.'

The 'Location' section has an 'Info' link. Below it is the 'Location scope' section with an 'Info' link. It states: 'You can't edit the location scope after registering the location.' The 'Scope' field contains 's3://'. To the right of the field are 'View' and 'Browse S3' buttons. Below the field, it provides format examples: 'Format for all buckets: s3://', 'Format for bucket: s3://<bucket>', and 'Format for prefix: s3://<bucket>/<prefix-with-path>'. The 'Permissions' section states: 'This IAM role allows S3 Access Grants to access your specified location scope.' Below this is a blue box with an information icon and text: 'To ensure the role has proper permissions to access the location scope, consult the [list of permissions](#).' The 'IAM role' section has an 'Info' link and states: 'This IAM role allows S3 Access Grants to access your specified location scope.' It has three radio buttons: 'Create new role' (selected), 'Choose from existing IAM roles', and 'Enter IAM role ARN'. At the bottom right are 'Cancel' and 'Register location' buttons.

다음을 선택하여 계속 진행합니다.

6. 권한 부여 생성 화면에서 다음 세부 정보를 제공합니다.
 - 권한에서 읽기 및 쓰기를 선택합니다. 액세스 권한 부여 권한은 읽기 전용 또는 읽기 및 쓰기일 수 있지만 쓰기 전용은 지원되지 않습니다.
 - 피부여자 유형에서 IAM Identity Center의 디렉터리 자격 증명을 선택합니다.

- 디렉터리 자격 증명 유형에서 지금 등록할 유형에 따라 사용자 또는 그룹을 선택합니다.
- IAM Identity Center 사용자/그룹 ID에서 사용자 또는 그룹의 ID를 붙여 넣습니다. 이 ID는 IAM Identity Center 콘솔과 사용자 및 그룹 테이블의 Transfer Family 웹 앱에서 사용할 수 있습니다.

Create Grant [Info](#)

Grant [Info](#)
 Grant a user or role a specific level of access to your S3 data.

Grant scope
 The grant scope is a combination of the location and the subprefix. Choose a registered location or register a new location. To narrow the scope, specify a subprefix. If you use the default "s3:///" location, you must specify a subprefix.

Location
 s3://test

Location ID [Info](#)
 [Empty field]

IAM role [Info](#)
[web-app-access-grants](#)

[Browse locations](#) [Register location](#)

Subprefix - optional [Info](#)
 Further scope the grant to a bucket, prefix, or an individual object.

Don't include any part of the grant already specified in the location.
 Format for bucket: <bucket>/*
 Format for prefix: <bucket>/<prefix-with-path>*
 Format for object: <bucket>/<prefix-with-path>/<object>

Grant scope [Info](#)
 s3://test/*

☐ The grant scope is an object

Permissions and access
 Grant a user or role a specific level of access to your S3 data.

⚠ The S3 console can't be used to access the data specified in the grant scope. To access the data, the grantee must do so through the AWS CLI, the application, or other AWS services.

Permissions
☒ Read
☒ Write

Grantee type [Info](#)
☒ Directory identity from IAM Identity Center
☐ IAM principal

Directory identity type [Info](#)
☒ User
☐ Group

IAM Identity Center user ID [Info](#)
 The user that you want to grant this level of access to. You can find the ID in the [IAM Identity Center console](#)

Format: a1b2c3d4-5678-90ab-cdef-EXAMPLE1111

▶ **Application - optional**

다음을 선택합니다.

7. 화면의 설정을 검토합니다. 모든 것이 올바르면 완료를 선택하여 액세스 권한을 생성합니다. 또는 취소 또는 이전을 선택하여 변경할 수 있습니다.

- Step 1
- Create S3 Access Grants instance
- Step 2
- Register S3 Buckets or prefixes as locations
- Step 3
- Create grant
- Step 4
- Review and finish

Review and finish

Step 1 - Create S3 Access Grants instance

S3 Access Grants instance

AWS Region

You can create one S3 Access Grants instance per AWS Region per account.

US East (Ohio) us-east-2

Add IAM Identity Center instance in US East (Ohio) us-east-2 - optional

You can use an IAM Identity Center instance to specify identities from corporate directories. If you don't have an existing IAM Identity Center instance, you can add one after creating your S3 Access Grants instance.

arn:aws:sso:::application/ssoins:/apl

Successfully created S3 Access Grants instance

Instance ARN: arn:aws:s3:us-east-2:::access-grants/default

You can grant other accounts access to this S3 Access Grants instance in the Resource Access Management console

Step 2 - Register S3 buckets or prefixes as locations

Location

Location scope

s3://

IAM role

web-app-beta-access-grants

Successfully registered location

Location ID:

Step 3 - Create grants

Grant

Grant a user or role a specific level of access to your S3 data.

Grant scope

s3:///*

IAM role

web-app-beta-access-grants

Location ID

Permissions and access

Access Type

READWRITE

Grantee type

Directory identity user

Grantee ID

Application - optional

-

Successfully created grant

Grant ARN: arn:aws:s3:us-east-2:::access-grants/default/grant/

Create another grant

Cancel

Previous

Finish

Amazon S3 > Access Grants > US East (Ohio) us-east-2: default

US East (Ohio) us-east-2: default [Info](#)[Share Instance](#)[Delete Instance](#)

S3 Access Grants provides scalable access control to data sets in your S3 buckets. To share an S3 Access Grants instance with external accounts by using the AWS Resource Access Manager console, choose Share instance. With S3 Access Grants, you can use identities from your corporate directory or from AWS Identity and Access Management (IAM) to control access. You can create one S3 Access Grants instance per AWS Region per account.

S3 Access Grants instance overview [Info](#)

Amazon Resource Name (ARN)

arn:aws:s3:us-east-2:/:access-grants/default

Creation date

October 25, 2024, 11:50:11 (UTC-04:00)

Account ID

IAM Identity Center [Info](#)[Deregister](#)[Add](#)

IAM Identity Center instance ARN

arn:aws:sso:::application/ssoins-:/apl

[Grants](#)[Locations](#)**Grants (1)** [Info](#)[Delete](#)[Create Grant](#)

< 1 > ⚙

Find by grant scope or grantee

Grant scope ▲	Grant ID ▼	Permission ▼	Grantee type ▼	Grantee ID 🔗 ▼	Creation date ▼	Location ID	Application ARN 🔗 ▼
☐ s3:///*		Read, Write	Directory identity user		October 25, 2024, 11:57:30 (UTC-04:00)		ALL

그러면 웹 앱에 대한 설정이 완료됩니다. 구성된 사용자 및 그룹은 액세스 포인트에서 웹 앱을 방문하여 로그인하고 파일을 업로드 및 다운로드할 수 있습니다.

사용자 지정 URL로 액세스 엔드포인트 업데이트

웹 앱으로 생성된 기본 액세스 엔드포인트에는 서비스 생성 식별자가 포함되어 있습니다. 브랜드 경험을 제공하기 위해 사용자가 Transfer Family 웹 앱에 액세스할 수 있는 사용자 지정 URL을 제공할 수 있습니다. 이 주제에서는 사용자 지정 URL로 액세스 엔드포인트를 업데이트하는 방법을 설명합니다.

Note

VPC 엔드포인트에 대해 액세스 엔드포인트를 사용자 지정할 수 없습니다. 사용자 지정 URL을 추가하려면 퍼블릭 엔드포인트를 사용합니다.

Note

다음 절차에서는 권장 [CloudFormation 스택 템플릿](#)을 사용합니다. 템플릿을 사용할 필요가 없습니다. [CloudFront 콘솔](#)을 직접 사용하여 배포를 생성할 수 있습니다.

그러나 제공된 템플릿은 프로세스를 간소화하고 잘못된 구성을 더 쉽게 방지할 수 있습니다.

CloudFormation 템플릿을 사용하지 않는 경우 다음 지침을 따라야 합니다.

- [오리진 요청 정책](#)은 쿼리 문자열과 쿠키를 오리진에 전달해야 하며 Host 헤더를 오리진에 전달해서는 안 됩니다.
- [캐시 정책](#)에는 캐시 키에 Host 헤더가 포함되어서는 안 됩니다.

웹 앱 URL을 사용자 지정하려면

1. Transfer Family 제공 AWS CloudFormation 템플릿인 [CloudFormation 스택 템플릿을 사용하여 CloudFront](#) CloudFront 배포를 생성합니다.
 - a. <https://console.aws.amazon.com/cloudformation> CloudFormation 콘솔을 엽니다.
 - b. 스택 생성을 선택하고 다음을 지정합니다.
 - 사전 조건 - 템플릿 준비 섹션에서 기존 템플릿 선택을 선택합니다.
 - 템플릿 지정 섹션에서 템플릿 파일 업로드를 선택합니다.
 - [CloudFormation 스택 템플릿](#) 파일을 저장한 다음 여기에 업로드합니다.
 - c. 다음을 선택하고 다음 정보를 제공합니다.
 - WebAppEndpoint: 웹 앱에서 값 복사
 - AccessEndpoint: 사용하려는 사용자 지정 도메인 이름을 제공합니다.
 - AcmCertificateArn:에 저장된 퍼블릭 또는 프라이빗 SSL/TLS 인증서의 ARN을 제공합니다.
AWS Certificate Manager
 - d. 새 스택이 생성될 때까지 CloudFormation 마법사를 완료합니다.
2. 웹 앱에서 액세스 엔드포인트를 편집하고 사용자 지정 URL을 사용하려는 URL로 업데이트합니다.

Transfer Family > Web apps > webapp- > Edit web app details

Edit web app details

Authentication access [Info](#)

Connect or create an Identity Center instance

Instance ARN

Select your instance ARN

Permission type [Info](#)

Create and use a new service role

Create and use an IAM role to allow the service to access the AWS resource it needs to create your application

Use an existing role

Use an existing IAM role to allow the service to access the AWS resources it needs to create your application

By selecting use existing role, you need to manually create an IAM Role in order for the Transfer Family web application to access your AWS services data. Please select an existing IAM role below or [create a new role in IAM console](#)

Select an existing role

If your newly created role is not showing up in the dropdown list, please try the refresh button on the right

Find your IAM role

Make sure you have attached the trust policy and permission policy to the role you selected.

[learn more about required policies](#)

Access endpoint [Info](#)

Map your own fully qualified domain name to this web app so your end users can use your custom URL, not the service-provided URL to access your web app. Note- you can only complete this step in CloudFront after you have created the web app.

Custom URL

https://my-custom-url

[Create a new distribution](#)

- DNS 레코드를 생성하여 사용자 지정 도메인 이름의 트래픽을 CloudFront 배포로 라우팅합니다. 영역에 Route 53를 사용하는 경우 CloudFront 배포 이름(예: xxxx.cloudfront.net)에 대한 별칭 또는 CNAME 레코드를 생성할 수 있습니다. CloudFront에서 Amazon Route 53를 사용하는 방법에 대한 자세한 내용은 [트래픽을 CloudFront 배포로 라우팅하도록 Amazon Route 53 구성을 참조하세요](#).
- AllowedOrigins 코드 블록에서 기본 액세스 엔드포인트를 다음 줄로 대체하여 교차 오리진 리소스 공유 정책을 업데이트합니다.

```
"https://custom-url"
```

웹 앱에서 사용하는 각 버킷에 대해 이 변경을 수행해야 합니다.

업데이트한 후 CORS 정책의 AllowedOrigins 섹션은 다음과 같아야 합니다.

```
"AllowedOrigins": [
  "https://custom-url",

```

각 Transfer Family 웹 앱에 대해 단일 AllowedOrigins 항목만 필요합니다.

자세한 내용은 [Amazon S3 버킷의 CORS\(Cross-Origin 리소스 공유\) 설정 절차를 참조하세요](#).

이제 사용자 지정 액세스 엔드포인트를 방문하여이 링크를 최종 사용자와 공유할 수 있습니다.

Transfer Family 웹 앱에 대한 CloudTrail 로깅

CloudTrail은 내에서 수행된 작업의 레코드를 AWS 서비스 생성하는 입니다 AWS 계정. 콘솔 로그인, AWS Command Line Interface 명령 및 SDK/API 작업과 같은 활동에 대한 API 작업을 지속적으로 모니터링하고 기록합니다. 이를 통해 누가 언제 어디서 어떤 조치를 취했는지에 대한 로그를 유지할 수 있습니다. CloudTrail은 AWS 환경의 모든 활동에 대한 기록을 제공하여 감사, 액세스 관리 및 규정 준수를 지원합니다.

Transfer Family 웹 앱의 경우 사용자가 수행한 인증 이벤트와 데이터 액세스 작업을 모두 추적할 수 있습니다. 포괄적인 로깅을 활성화하려면 다음을 수행해야 합니다.

1. 인증 활동을 추적하기 위한 관리 이벤트를 로깅하도록 CloudTrail을 구성합니다.
2. Amazon S3 데이터 이벤트를 활성화하여 웹 앱을 통해 수행된 파일 작업을 추적합니다.

참고 항목

- [IAM Identity Center의 CloudTrail 사용 사례](#)
- [IAM Identity Center 로그인 이벤트 이해](#)
- [CloudTrail userIdentity 요소](#)
- [S3 버킷 및 객체에 대한 CloudTrail 이벤트 로깅 사용 설정](#)
- [Amazon S3 CloudTrail 이벤트](#)

Amazon S3 데이터 이벤트 활성화

Amazon S3 버킷에서 Transfer Family 웹 앱을 통해 수행된 파일 작업을 추적하려면 해당 버킷에 대한 데이터 이벤트를 활성화해야 합니다. 데이터 이벤트는 객체 수준 API 활동을 제공하며 웹 앱 사용자가 수행하는 파일 업로드, 다운로드 및 기타 작업을 추적하는 데 특히 유용합니다.

Transfer Family 웹 앱에 대해 Amazon S3 데이터 이벤트를 활성화하려면:

1. <https://console.aws.amazon.com/cloudtrail/>에서 CloudTrail 콘솔을 엽니다.
2. 탐색 창에서 추적을 선택한 다음 기존 추적을 선택하거나 새 추적을 생성합니다.
3. 고급 이벤트 선택기에서 편집을 선택합니다.
4. 고급 이벤트 선택기 추가를 선택합니다.

5. 첫 번째 필드 선택기의 경우:

- 필드를 로 설정 `eventCategory`
- 연산자를 같음으로 설정
- 값을 로 설정 `Data`

6. 필드 추가를 선택하고 두 번째 필드 선택기를 선택합니다.

- 필드를 로 설정 `resources.type`
- 연산자를 같음으로 설정
- 값을 로 설정 `AWS::S3::Object`

7. (선택 사항) 특정 버킷에 대해서만 이벤트를 로깅하려면 필드 추가를 선택하고 다음을 추가합니다.

- 필드를 로 설정 `resources.ARN`
- 연산자를 로 시작으로 설정
- 값을 로 설정 `arn:aws:s3:::your-bucket-name/`

8. 변경 사항 저장을 선택합니다.

또는 레거시 데이터 이벤트 구성을 사용할 수 있습니다.

1. 데이터 이벤트에서 편집을 선택합니다.
2. 데이터 이벤트 유형에서 S3 버킷 및 객체 이벤트를 선택합니다.
3. 데이터 이벤트를 로깅할 Amazon S3 버킷을 선택합니다. 모든 현재 및 미래 S3 버킷을 선택하거나 개별 버킷을 지정할 수 있습니다.
4. 읽기 이벤트, 쓰기 이벤트 또는 둘 다를 로깅할지 여부를 선택합니다.
5. 변경 사항 저장을 선택합니다.

데이터 이벤트를 활성화한 후 CloudTrail용으로 구성된 Amazon S3 버킷에서 이러한 로그에 액세스할 수 있습니다. 로그에는 작업을 수행한 사용자, 작업 타임스탬프, 영향을 받는 특정 객체, Transfer Family 웹 앱을 통해 수행된 작업에 `userId` 대해를 추적하는 데 도움이 되는 `onBehalfOf` 필드와 같은 세부 정보가 포함됩니다.

로그 찾기 및 보기

Transfer Family 웹 앱에 대한 CloudTrail 로그를 찾고 보는 방법에는 여러 가지가 있습니다.

CloudTrail 콘솔 사용

최근 이벤트를 보는 가장 빠른 방법:

1. <https://console.aws.amazon.com/cloudtrail/>에서 CloudTrail 콘솔을 엽니다.
2. 이벤트 기록을 선택합니다.
3. 다음을 기준으로 이벤트를 필터링합니다.
 - 이벤트 소스: 웹 앱 이벤트signin.amazonaws.com용
 - 이벤트 소스: 파일 작업s3.amazonaws.com용
4. 이벤트를 클릭하면 자세한 정보를 볼 수 있습니다.

Amazon S3에서 로그 액세스

Amazon S3에 저장된 전체 로그 파일에 액세스하려면:

1. CloudTrail 추적의 Amazon S3 버킷을 식별합니다.

```
aws cloudtrail describe-trails --query 'trailList[*].[Name,S3BucketName]' --output
table
```

2. Amazon S3의 로그 파일로 이동합니다.

```
aws s3 ls s3://your-cloudtrail-bucket/AWSLogs/account-id/CloudTrail/region/YYYY/MM/
DD/
```

3. 웹 앱 ID에 대한 로그 파일을 다운로드하고 검색합니다.

```
aws s3 cp s3://your-cloudtrail-bucket/AWSLogs/account-id/CloudTrail/region/YYYY/MM/
DD/ . --recursive
gunzip *.json.gz
grep -l "webapp-1a2b3c4d5e6f7g8h9" *.json
```

AWS CLI 를 사용하여 이벤트 검색

AWS CLI다음을 사용하여 특정 웹 앱 이벤트를 검색합니다.

```
aws logs filter-log-events \
  --log-group-name /aws/cloudtrail/your-trail-name \
  --filter-pattern "webapp-1a2b3c4d5e6f7g8h9" \
  --start-time $(date -d "1 day ago" +%s)000
```

또는 인증 이벤트를 검색합니다.

```
aws logs filter-log-events \
  --log-group-name /aws/cloudtrail/your-trail-name \
  --filter-pattern "UserAuthentication" \
  --start-time $(date -d "1 day ago" +%s)000
```

인증 로그 예제

CloudTrail은 Transfer Family 웹 앱에 대한 인증 이벤트를 로깅하므로 성공 및 실패한 로그인 시도를 추적하는 데 도움이 될 수 있습니다. 이러한 로그는 보안 모니터링 및 규정 준수 목적에 특히 유용합니다.

주제

- [자격 증명 확인을 위한 로그 항목의 예](#)
- [로그인 인증을 위한 로그 항목 예제](#)
- [ListCallerAccessGrants의 로그 항목 예제](#)
- [GetDataAccess 이벤트에 대한 로그 항목 예제](#)

자격 증명 확인을 위한 로그 항목의 예

다음 예제는 인증 프로세스 중에 발생하는 자격 증명 확인 이벤트에 대한 CloudTrail 로그 항목을 보여줍니다.

```
{
  "eventVersion": "1.09",
  "userIdentity": {
    "type": "Unknown",
    "principalId": "123456789012",
    "arn": "",
    "accountId": "123456789012",
    "accessKeyId": "",
    "userName": "demo-user-2",
    "onBehalfOf": {
      "userId": "f12bb510-a011-702f-10dd-5607e2776dbc",
      "identityStoreArn": "arn:aws:identitystore::123456789012:identitystore/d-9a670c546e"
    },
    "credentialId": "58138a11-87e5-401d-8f0b-7161c9389112"
  },
  "eventTime": "2025-08-08T15:29:30Z",
```

```

    "eventSource": "signin.amazonaws.com",
    "eventName": "CredentialVerification",
    "awsRegion": "us-east-2",
    "sourceIPAddress": "192.0.2.224",
    "userAgent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/138.0.0.0 Safari/537.36",
    "requestParameters": null,
    "responseElements": null,
    "additionalEventData": {
        "AuthWorkflowID": "f304a48b-7b6d-41c8-b136-4f49c91c1f31",
        "CredentialType": "PASSWORD"
    },
    "requestID": "ff936828-4a81-453c-802d-81368b6bca1a",
    "eventID": "70cb7008-493d-42c2-a9eb-38bf168af6a8",
    "readOnly": false,
    "eventType": "AWS ServiceEvent",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "serviceEventDetails": {
        "CredentialVerification": "Success"
    },
    "eventCategory": "Management"
}

```

이 이벤트는 사용된 특정 자격 증명 ID 및 인증 워크플로 ID를 보여주는 인증 프로세스의 자격 증명 확인 단계에 대한 추가 세부 정보를 제공합니다.

로그인 인증을 위한 로그 항목 예제

다음 예제는 IAM Identity Center를 사용하여 웹 앱 로그인 중 성공적인 사용자 인증 이벤트에 대한 CloudTrail 로그 항목을 보여줍니다.

```

{
    "eventVersion": "1.09",
    "userIdentity": {
        "type": "Unknown",
        "principalId": "123456789012",
        "arn": "",
        "accountId": "123456789012",
        "accessKeyId": "",
        "userName": "demo-user-2",
        "onBehalfOf": {
            "userId": "f12bb510-a011-702f-10dd-5607e2776dbc",

```

```

    "identityStoreArn": "arn:aws:identitystore::123456789012:identitystore/d-9a670c546e"
  },
  "credentialId": "b41f0a02-1635-4d07-a414-aecf9e14b906"
},
"eventTime": "2025-08-07T14:09:07Z",
"eventSource": "signin.amazonaws.com",
"eventName": "UserAuthentication",
"awsRegion": "us-east-2",
"sourceIPAddress": "192.0.2.14",
"userAgent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/138.0.0.0 Safari/537.36",
"requestParameters": null,
"responseElements": null,
"additionalEventData": {
  "AuthWorkflowID": "7a4ef12c-7c4b-4bc3-b5bd-c2469afcc795",
  "LoginTo": "https://example.awsapps.com/start/",
  "CredentialType": "PASSWORD"
},
"requestID": "fc91bcf0-ac53-4454-a1a0-fb911eacc095",
"eventID": "18522007-1e60-4a71-b2b5-150baf504ab3",
"readOnly": false,
"eventType": "AWS ServiceEvent",
"managementEvent": true,
"recipientAccountId": "123456789012",
"serviceEventDetails": {
  "UserAuthentication": "Success"
},
"eventCategory": "Management"
}

```

이 예제에서는 다음 중요 필드를 기록해 둡니다.

- `eventSource: "signin.amazonaws.com"`를 표시합니다. 이는 IAM Identity Center 인증 이벤트를 나타냅니다.
- `userIdentity.onBehalfOf`: 웹 앱 사용자의 사용자 ID 및 자격 증명 스토어 ARN을 포함합니다.
- `additionalEventData.LoginTo`: 액세스 중인 IAM Identity Center 애플리케이션 URL을 표시합니다.
- `additionalEventData.CredentialType`: 사용된 인증 방법(PASSWORD)을 나타냅니다.
- `serviceEventDetails`: 인증 결과(성공)를 표시합니다.

ListCallerAccessGrants의 로그 항목 예제

다음 예제는 Transfer Family 웹 앱이 사용자에게 대해 사용 가능한 액세스 권한을 쿼리할 때 발생하는 ListCallerAccessGrants 이벤트에 대한 CloudTrail 로그 항목을 보여줍니다.

```
{
  "eventVersion": "1.11",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "AROEXAMPLEID:aws-transfer",
    "arn": "arn:aws:sts::123456789012:assumed-role/AWS
TransferWebAppIdentityBearer-us-east-2/aws-transfer",
    "accountId": "123456789012",
    "accessKeyId": "ASIAEXAMPLEKEY",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "AROEXAMPLEID",
        "arn": "arn:aws:iam::123456789012:role/service-role/AWS
TransferWebAppIdentityBearer-us-east-2",
        "accountId": "123456789012",
        "userName": "AWS TransferWebAppIdentityBearer-us-east-2"
      },
      "attributes": {
        "creationDate": "2025-08-08T15:29:34Z",
        "mfaAuthenticated": "false"
      }
    },
    "invokedBy": "transfer.amazonaws.com",
    "onBehalfOf": {
      "userId": "f12bb510-a011-702f-10dd-5607e2776dbc",
      "identityStoreArn": "arn:aws:identitystore::123456789012:identitystore/
d-9a670c546e"
    }
  },
  "eventTime": "2025-08-08T15:29:35Z",
  "eventSource": "s3.amazonaws.com",
  "eventName": "ListCallerAccessGrants",
  "awsRegion": "us-east-2",
  "sourceIPAddress": "transfer.amazonaws.com",
  "userAgent": "transfer.amazonaws.com",
  "requestParameters": {
    "Host": "123456789012.s3-control.dualstack.us-east-2.amazonaws.com",
```

```

        "allowedByApplication": "true",
        "maxResults": "100"
    },
    "responseElements": null,
    "additionalEventData": {
        "SignatureVersion": "SigV4",
        "CipherSuite": "TLS_AES_128_GCM_SHA256",
        "bytesTransferredIn": 0,
        "AuthenticationMethod": "AuthHeader",
        "x-amz-id-2": "1g34AaAELn/
fntxwriFVsr41VDl8dp5ygWFasHJFNVq5FDCWYfX0ye7s4tWHEJC8ppI5lLePYLIcw3iTXAgn5Q==",
        "bytesTransferredOut": 462
    },
    "requestID": "48485MTZEDWT0ANT",
    "eventID": "3de5dd60-b7cf-474c-a1ab-631467c1a5c3",
    "readOnly": true,
    "resources": [
        {
            "accountId": "123456789012",
            "type": "AWS:S3::AccessGrantsInstance",
            "ARN": "arn:aws:s3:us-east-2:123456789012:access-grants/default"
        }
    ],
    "eventType": "AWS ApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management"
}

```

이 예제에서는 다음 중요 필드를 기록해 둡니다.

- `eventName`: 사용 가능한 S3 액세스 권한 부여를 쿼리하는 `ListCallerAccessGrants` 이벤트를 보여줍니다.
- `requestParameters.allowedByApplication`: 쿼리가 애플리케이션에서 허용하는 권한 부여로 필터링됨을 나타냅니다.
- `requestParameters.maxResults`: 응답에 반환할 최대 권한 부여 수를 표시합니다.
- `userIdentity.onBehalfOf`: 요청을 특정 웹 앱 사용자에게 연결합니다.

이 이벤트는 Transfer Family 웹 앱이 사용자가 액세스할 수 있는 S3 리소스를 쿼리하는 시기를 추적하여 액세스 권한 부여 검색 작업에 대한 가시성을 제공합니다.

GetDataAccess 이벤트에 대한 로그 항목 예제

다음 예제는 Transfer Family 웹 앱이 사용자를 대신하여 S3 리소스에 대한 액세스 권한을 요청할 때 발생하는 GetDataAccess 이벤트에 대한 CloudTrail 로그 항목을 보여줍니다.

```
{
  "eventVersion": "1.11",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "AROASEQRAEABP7ADWEZA5:aws-transfer",
    "arn": "arn:aws:sts::123456789012:assumed-role/AWSTransferWebAppIdentityBearer-
ap-southeast-1/aws-transfer",
    "accountId": "123456789012",
    "accessKeyId": "ASIAEXAMPLEKEY",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "AROASEQRAEABP7ADWEZA5",
        "arn": "arn:aws:iam::123456789012:role/service-role/
AWSTransferWebAppIdentityBearer-ap-southeast-1",
        "accountId": "123456789012",
        "userName": "AWSTransferWebAppIdentityBearer-ap-southeast-1"
      },
      "attributes": {
        "creationDate": "2025-05-08T16:09:05Z",
        "mfaAuthenticated": "false"
      }
    },
    "invokedBy": "transfer.amazonaws.com",
    "onBehalfOf": {
      "identityStoreArn": "arn:aws:identitystore::123456789012:identitystore/
d-9667b0da7a",
      "userId": "191a35ec-10a1-70c1-e4ab-e2802411e13e"
    }
  },
  "eventTime": "2025-05-08T16:10:25Z",
  "eventSource": "s3.amazonaws.com",
  "eventName": "GetDataAccess",
  "awsRegion": "ap-southeast-1",
  "sourceIPAddress": "transfer.amazonaws.com",
  "userAgent": "transfer.amazonaws.com",
  "requestParameters": {
    "Host": "123456789012.s3-control.dualstack.ap-southeast-1.amazonaws.com",
```

```

        "durationSeconds": 900,
        "permission": "READWRITE",
        "target": "s3://amzn-s3-demo-bucket/users/john.doe/documents/*"
    },
    "responseElements": null,
    "additionalEventData": {
        "AuthenticationMethod": "AuthHeader",
        "CipherSuite": "TLS_AES_128_GCM_SHA256",
        "SignatureVersion": "SigV4",
        "bytesTransferredIn": 0,
        "bytesTransferredOut": 2244,
        "x-amz-id-2": "8ce8sZ0gNwsaj9w1mzagya
+cs0NjYl8FgEw4FGpE8DARi90aNC0RFWlTYNEn7ChqE9RCJrTzMvS+ru7Vz2xXHrkQt/1uQ9exZTZd1hX+/fM="
    },
    "requestID": "BXGSKKQXCWS5RAHB",
    "eventID": "c11db1d1-dfb8-431e-8625-48eba2ebadfe",
    "readOnly": true,
    "resources": [
        {
            "type": "AWS:S3::AccessGrantsInstance",
            "ARN": "arn:aws:s3:ap-southeast-1:123456789012:access-grants/default",
            "accountId": "123456789012"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "123456789012",
    "eventCategory": "Management"
}

```

이 예제에서는 다음 중요 필드를 기록해 둡니다.

- `eventName`: Transfer Family가 S3 리소스에 대한 액세스 권한을 요청할 때 발생하는 `GetDataAccess` 이벤트를 보여줍니다.
- `userIdentity.onBehalfOf`: 액세스 요청을 특정 웹 앱 사용자에게 연결하는 자격 증명 스토어 ARN 및 사용자 ID를 포함합니다.
- `requestParameters.target`: 액세스가 요청된 S3 경로 패턴을 표시합니다.
- `requestParameters.permission`: 요청된 액세스 유형(READWRITE, READ 또는 WRITE)을 나타냅니다.
- `requestParameters.durationSeconds`: 액세스 권한 부여가 유효한 기간을 표시합니다(일반적으로 900초/15분).

- sourceIPAddress 및 userAgent: 둘 다 "transfer.amazonaws.com"을 표시합니다. 이는 내부 서비스 요청임을 나타냅니다.

GetDataAccess 이벤트는 Transfer Family 웹 앱 사용자에게 특정 S3 리소스에 대한 액세스 권한이 부여되는 시점을 추적하는 데 특히 유용하므로 액세스 패턴을 모니터링하고 적절한 인증을 보장하는 데 도움이 됩니다.

CloudTrail 로그 항목 보기

Transfer Family 웹 앱의 CloudTrail 로그 항목을 보고 분석하는 방법에는 여러 가지가 있습니다.

CloudTrail 콘솔 사용

CloudTrail 콘솔은 로그 항목을 보고 필터링할 수 있는 사용자 친화적 인터페이스를 제공합니다.

1. <https://console.aws.amazon.com/cloudtrail/>에서 CloudTrail 콘솔을 엽니다.
2. 탐색 창에서 Event history(이벤트 내역)를 선택합니다.
3. 필터 옵션을 사용하여 이벤트의 범위를 좁힙니다.
 - Transfer Family 이벤트만 transfer.amazonaws.com 보려면 이벤트 소스를 로 설정합니다.
 - 이벤트 이름을 기준으로 필터링하여와 같은 특정 작업을 확인합니다UserAuthentication.
 - 시간 범위를 사용하여 특정 기간 내의 이벤트에 집중합니다.
4. 이벤트를 클릭하여 세부 정보를 확인합니다.

Amazon S3에서 로그 액세스

Amazon S3 버킷에 로그를 전송하도록 CloudTrail 추적을 구성한 경우 원시 로그 파일에 직접 액세스할 수 있습니다.

1. <https://console.aws.amazon.com/s3/>에서 S3 콘솔을 엽니다.
2. CloudTrail 로그가 저장되는 버킷 및 접두사로 이동합니다.
3. 로그는 연도, 월, 일 및 리전별로 구성됩니다. 적절한 디렉터리로 이동합니다.
4. JSON 형식의 로그 파일을 다운로드하여 엽니다.

웹 앱 문제 해결

Note

이러한 문제 해결 팁은 최종 사용자가 아닌 웹 앱 관리자를 위한 것입니다. 최종 사용자의 경우 문제가 발생하면 웹 앱 관리자에게 문의하세요. 다음 단락의 모든 인스턴스는 웹 앱 관리자를 참조합니다.

네트워크 오류 문제 해결

설명

웹 앱 엔드포인트를 로드하면 최종 사용자에게 네트워크 배너 네트워크 오류가 표시됩니다.

원인

가장 일반적인 문제는 다음과 같습니다.

- 관리자가 새 애플리케이션에 로그인하려는 사용자를 할당하지 않았습니다.
- 관리자가 IAM 역할에 필요한 작업을 추가하지 않았습니다.
- 사용자에게 할당된 S3 Access Grants 목록이 표시되지만 Amazon S3 버킷 또는 버킷에 대해 CORS가 올바르게 구성되지 않았습니다.

솔루션

- IAM Identity Center에서 사용자를 올바른 애플리케이션에 할당해야 합니다. 또는 그룹이 할당된 경우 로그인을 시도하는 사용자가 올바른 그룹에 속하는지 확인합니다. 이에 대한 설명은 [Transfer Family 웹 앱에 사용자 또는 그룹 할당 또는 추가](#)에 나와 있습니다.
- 역할에 sts:AssumeRole 및 작업 모두에 대한 사용자 지정 신뢰 정책에 필요한 sts:SetContext 작업이 포함되어 있는지 확인합니다. 이에 대한 설명은 [Transfer Family 웹 앱에 대한 IAM 역할 구성](#)에 나와 있습니다.
- 웹 앱에서 사용하는 모든 버킷에 대한 CORS 정책을 확인합니다. 이는 [Amazon S3 버킷에 대한 CORS\(Cross-Origin 리소스 공유\) 설정에 설명되어 있습니다](#).

구성된 버킷이 나타나지 않는 문제 해결

설명

모든 것이 올바르게 구성된 것처럼 보이지만 Amazon S3 버킷이 웹 앱에 표시되지 않습니다.

원인

가능한 원인 중 하나는 Amazon S3 버킷이 웹 앱과 동일한 계정에 있지 않기 때문입니다.

솔루션

Amazon S3 버킷이 웹 앱과 동일한 계정에 있는지 확인합니다. 교차 계정 버킷은 현재 지원되지 않습니다.

사용자 지정 URL 오류 문제 해결

설명

최종 사용자가 웹 앱에 로그인하면 권한 부여 실패: 권한 부여 코드 누락이라는 오류 메시지가 표시됩니다.

원인

제공된 CloudFormation 템플릿이 아닌 CloudFront를 직접 사용한 경우 쿼리 문자열을 전달하지 않도록 오리진 요청 정책을 잘못 구성했을 수 있습니다.

솔루션

오리진 요청 정책을 업데이트하여 쿼리 문자열과 쿠키를 오리진에 전달합니다.

설명

최종 사용자가 Transfer Family 웹 앱에 액세스하려고 하면 404 응답을 받습니다.

원인

제공된 CloudFormation 템플릿이 아닌 CloudFront를 직접 사용한 경우 캐시 키에 Host 헤더를 포함하도록 캐시 정책을 잘못 구성했거나 Host 헤더를 전달하도록 오리진 요청 정책을 잘못 구성했을 수 있습니다.

솔루션

- 캐시 정책이 캐시 키에 Host 헤더를 포함하지 않는지 확인합니다.
- 오리진 요청 정책이 Host 헤더를 전달하지 않는지 확인합니다.

기타 오류 문제 해결

설명

최종 사용자가 로그인할 수 없거나 버킷 또는 파일을 볼 수 없거나 다른 오류가 발생합니다.

원인

가능한 원인 중 하나는 IAM Identity Center 인스턴스 ARN이 권한 부여 ARN 또는 웹 앱 IAM Identity Center 인스턴스 ARN의 값과 일치하지 않기 때문입니다.

솔루션

다음 항목을 확인하여 일치하는지 확인합니다.

- IAM Identity Center에서 설정으로 이동하여 인스턴스 ARN을 확인합니다.

```
arn:aws:sso:::instance/ssoins-instance-identifier
```

- Amazon S3에서 Access Grants로 이동하여 IAM Identity Center 인스턴스 ARN을 확인합니다.

```
arn:aws:sso:::account-id:application/ssoins-instance-identifier/ap1-1234567890abcdef0
```

- Transfer Family에서 웹 앱 세부 정보 페이지로 이동하여 인스턴스 ARN을 확인합니다.

```
arn:aws:sso:::instance/ssoins-instance-identifier
```

값은 이러한 세 위치 모두에서 동일해야 합니다.

웹 앱에 표시되는 중복 S3 버킷

설명

사용자는 Transfer Family 웹 앱 인터페이스에 동일한 S3 버킷이 여러 번 나열됩니다.

원인

이는 사용자가 동일한 S3 버킷에 대한 중복 권한이 있는 여러 Active Directory 그룹에 속할 때 발생합니다. 웹 앱은 사용자에게 동일한 버킷 위치에 할당된 여러 권한 부여가 있는지 여부에 관계없이 사용자와 연결된 모든 최상위 권한 부여(UID 또는 GID)를 나열합니다.

솔루션

이 문제를 해결하려면 각 사용자가 각 S3 위치에 대해 하나의 권한만 갖도록 관리자가 권한 부여를 중복 제거해야 합니다. S3 Access Grants 구성을 검토하고 서로 다른 Active Directory 그룹에서 동일한 버킷에 대한 중복 권한 부여를 통합합니다.

Transfer Family 웹 앱에 대한 최종 사용자 지침

Note

이 주제에서는 웹 앱과 상호 작용하는 최종 사용자를 위한 정보입니다. 이 주제에서 사용자의 모든 인스턴스는 최종 사용자를 참조합니다.

이 주제에서는 사용 권한이 있는 AWS Transfer Family 웹 앱에 액세스하는 방법과 웹 앱과 상호 작용하는 방법을 설명합니다.

웹 앱 할당량

웹 앱을 사용할 때 다음과 같은 제한 사항에 유의하세요.

- 쿼리당 최대 검색 결과 수: 10,000
- Transfer Family 웹 앱에서 사용하는 Amazon S3 버킷은 웹 앱 자체와 동일한 계정에 있어야 합니다. 교차 계정 버킷은 현재 지원되지 않습니다.
- 쿼리당 최대 검색 범위: 검색된 파일 10,000개
- 파일당 최대 업로드 크기: 160GB(149GiB)
- 복사를 위한 최대 크기 파일: 5.36GB(5GiB)
- 점(.)으로 시작하거나 끝나는 폴더 이름은 지원되지 않습니다.

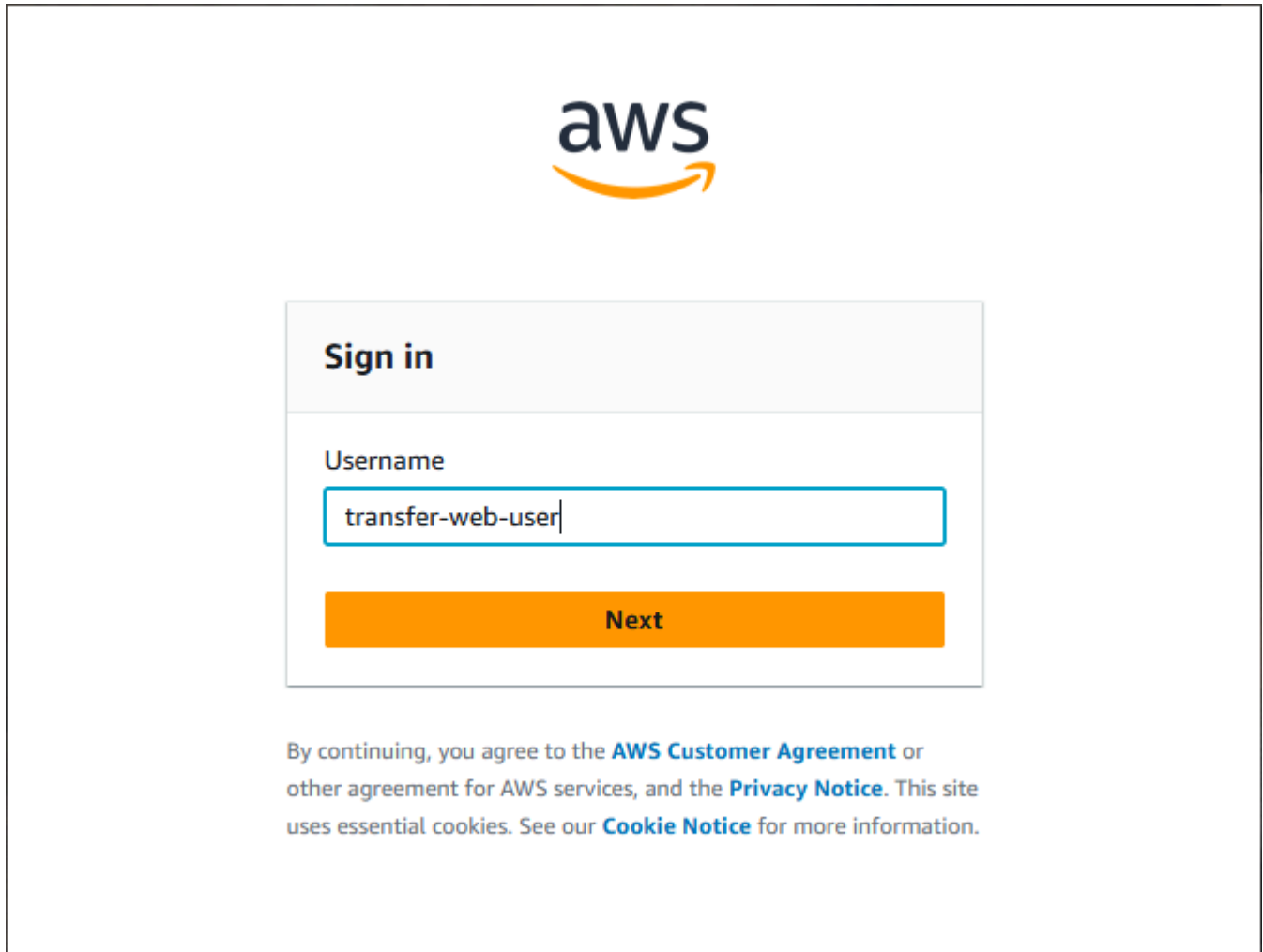
IAM Identity Center 사용자를 위한 사용자 경험

이 섹션에서는 조직에서 IAM Identity Center를 사용하여 사용자를 구성한 경우의 사용자 경험을 설명합니다.

Transfer Family 웹 앱에 액세스하려면

1. no-reply@login.awsapps.com에서 “참가 초대”라는 제목의 이메일을 받게 됩니다 AWS IAM Identity Center. 사용자 계정을 활성화하라는 초대를 수락합니다.
2. 메시지에서 AWS 액세스 포털 URL 아래의 URL을 선택합니다.

그러면 AWS 로그인 화면으로 이동합니다.

The image shows the AWS Sign in interface. At the top center is the AWS logo. Below it is a white box with a light gray header that says "Sign in". Inside the box, there is a "Username" label above a text input field containing "transfer-web-user". Below the input field is an orange button labeled "Next". At the bottom of the box, there is a line of text: "By continuing, you agree to the [AWS Customer Agreement](#) or other agreement for AWS services, and the [Privacy Notice](#). This site uses essential cookies. See our [Cookie Notice](#) for more information."

3. 자격 증명을 입력하고 로그인을 선택합니다.

그러면 사용 가능한 애플리케이션 목록이 AWS 액세스 포털표시된 로 이동합니다.

4. Transfer Family 웹 앱의 애플리케이션을 선택합니다.

타사 ID 제공업체 사용자를 위한 사용자 경험

조직에서 사용자를 구성하는 AWS IAM Identity Center 데를 사용하지 않은 경우 온보딩 환경은 최종 사용자를 구성하는 데 사용한 자격 증명 공급자 애플리케이션에 따라 달라집니다. 인증하고 로그인한 후 웹 앱 인터페이스는 다음 섹션에 설명된 것과 동일합니다.

Note

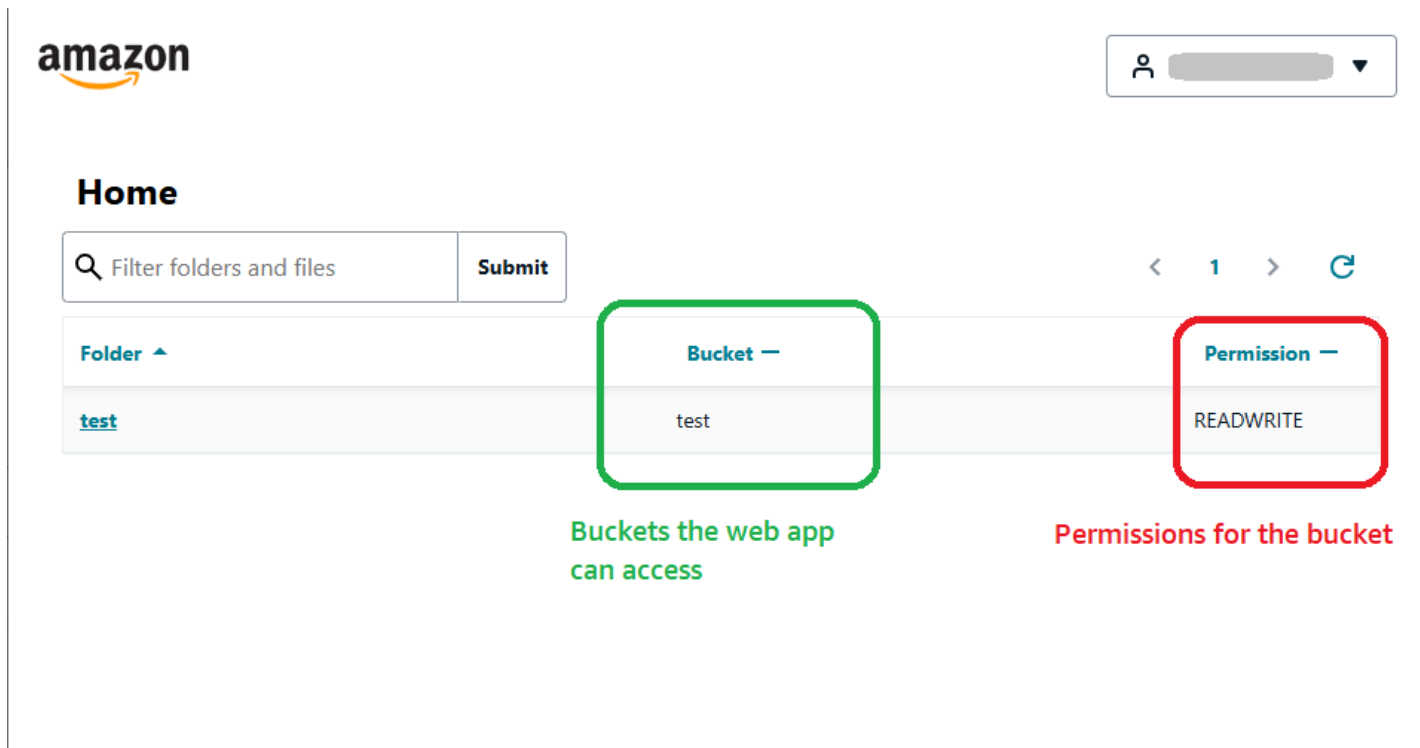
사용자가 동일한 Amazon S3 버킷에 대한 권한이 있는 여러 Active Directory 그룹에 속하는 경우 버킷은 웹 앱 인터페이스에 여러 번 나타납니다. 이는 웹 앱이 동일한 버킷에 대한 중복 권한 부여를 포함하여 사용자의 UID 또는 GID와 연결된 모든 최상위 권한 부여를 나열하기 때문에 발생합니다. 중복 등록을 방지하기 위해 관리자는 각 사용자에게 Amazon S3 위치당 하나의 권한만 부여하도록 여러 권한 부여를 통합할 수 있습니다.

Transfer Family 최종 사용자 인터페이스

인증하고 로그인한 후 웹 앱과 상호 작용할 수 있습니다.

네 가지 기본 보기가 있습니다.

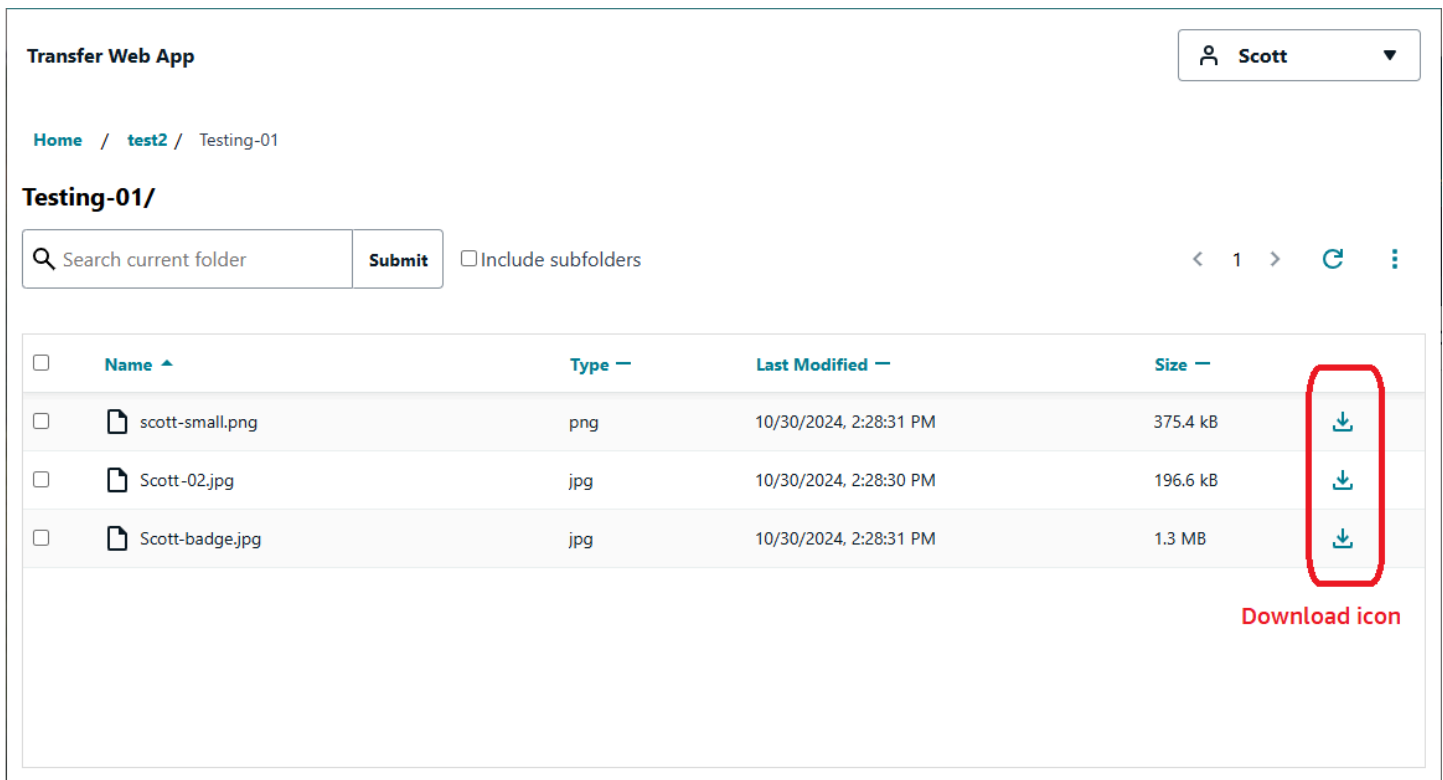
- 홈 페이지: 홈 페이지에는 액세스할 수 있는 S3 위치와 각 위치에 대한 권한이 나열됩니다. S3 위치는 S3 버킷 또는 접두사로, S3 Access Grants를 사용할 때 정의할 수 있습니다. 최종 사용자가 액세스할 수 있는 루트 수준 S3 리소스와 각 S3 위치에 대한 권한을 보여주는 사용자의 초기 보기입니다.



- 위치 세부 정보: 이 보기를 통해 사용자는 S3에서 파일 및 폴더를 둘러보고 파일을 업로드하거나 다운로드할 수 있습니다.
- 위치 작업: 작업(예: 업로드)을 선택하면 파일 위치의 다른 보기가 열립니다.
- 세로 줄임표: 세로 줄임표 아이콘이 작업 메뉴를 엽니다.

사용 가능한 작업

대부분의 작업은 작업 메뉴에서 사용할 수 있습니다. 파일을 다운로드하는 다른 주요 작업의 경우 파일을 선택한 후 다운로드 아이콘을 사용합니다(현재 한 번에 하나의 파일만 다운로드할 수 있음).



폴더에서 작업 메뉴를 사용하여 다음 작업 중 하나를 수행합니다.

- 하나 이상의 파일을 다른 위치에 복사합니다.
- 폴더를 만듭니다.
- 하나 이상의 파일을 삭제합니다.
- 하나 이상의 파일을 업로드합니다.
- 전체 폴더(있는 경우 하위 폴더 포함)를 업로드합니다.
- 폴더를 선택하고 해당 폴더로 이동합니다. 그런 다음 이전에 나열된 작업 중 하나를 수행할 수 있습니다.
- 페이지별로 정렬합니다.
- 폴더 및 하위 폴더당 파일 또는 폴더 이름을 기준으로 필터링합니다.

amazon

Home / test

test

Search current folder

Submit

☐ Include subfolders

< 1 > ↺ ⋮

Copy files

Create folder

Delete files

Upload files

Upload folder

Name ▲	Type —	Last Modified —
My First Folder/	Folder	
My Second Folder/	Folder	
Testing-01/	Folder	
Testing-02/	Folder	

사용 가능한 작업

314

AWS Transfer Family SFTP 커넥터

SFTP 커넥터는 AWS Transfer Family SFTP 프로토콜을 사용하여 Amazon 스토리지와 원격 서버 간에 파일을 전송하기 위해 원격 SFTP 서버와의 연결을 설정합니다. Amazon S3에서 파트너 소유의 외부 SFTP 서버로 파일을 보내거나, 파트너의 SFTP 서버에서 Amazon S3로 파일을 검색하거나, 원격 서버에서 파일을 나열, 삭제, 이름 바꾸기 또는 이동할 수 있습니다. SFTP 커넥터는 서비스 관리형 (AWS 관리형 인프라 사용)과 VPC(Amazon VPC Lattice를 사용하여 VPC를 통해 라우팅)의 두 가지 송신 유형을 지원합니다. SFTP 커넥터를 사용하면에서 자동화된 이벤트 기반 파일 전송 워크플로를 구축할 수 있습니다 AWS .

Transfer Family [AWS Transfer Family SFTP 커넥터](#)에 대한 간략한 소개는 SFTP 커넥터를 참조하세요.

주제

- [SFTP 커넥터 생성](#)
- [SFTP 커넥터에 대한 VPC 연결](#)
- [SFTP 커넥터 사용](#)
- [SFTP 커넥터 모니터링](#)
- [SFTP 커넥터 관리](#)
- [SFTP 커넥터의 크기 조정 및 할당량](#)
- [SFTP 커넥터를 사용한 참조 아키텍처](#)

SFTP 커넥터 생성

이 주제에서는 SFTP 커넥터를 생성하는 방법을 설명합니다. 각 커넥터는 하나의 원격 SFTP 서버에 연결할 수 있는 기능을 제공합니다. 다음과 같은 상위 수준 작업을 수행하여 SFTP 커넥터를 구성합니다.

Note

Virtual Private Cloud를 통해 트래픽을 라우팅하는 VPC 기반 커넥터는 섹션을 참조하세요 [VPC 기반 송신을 사용하여 SFTP 커넥터 생성](#).

1. 커넥터의 인증 자격 증명에 저장합니다 AWS Secrets Manager.

2. 보안 암호 ARN, 원격 서버의 URL 또는 리소스 구성 ARN, 커넥터에서 지원할 알고리즘이 포함된 보안 정책 및 기타 구성 설정을 지정하여 커넥터를 생성합니다.
3. 커넥터를 생성한 후 테스트하여 원격 SFTP 서버와의 연결을 설정할 수 있는지 확인할 수 있습니다.

SFTP 커넥터 송신 유형 선택

SFTP 커넥터를 생성할 때 "서비스 관리형"과 "VPC Lattice" 중에서 송신 유형을 선택합니다.

- 서비스 관리형(기본값): 커넥터는 NAT 게이트웨이와가 소유한 IP 주소를 사용하여 퍼블릭 인터넷 AWS Transfer Family 을 통해 연결을 라우팅합니다. 이 서비스는 연결을 설정하기 위해 원격 서버에 허용 목록에 등록해야 하는 커넥터에 대해 3개의 고정 IP 주소를 제공합니다.
- VPC Lattice: 커넥터는 Amazon VPC Lattice를 사용하여 VPC 환경을 통해 트래픽을 라우팅합니다. 다음 시나리오에서는 SFTP 커넥터에 VPC 연결을 사용합니다.
 - 프라이빗 SFTP 서버: VPC에서만 액세스할 수 있는 SFTP 서버에 연결
 - 온프레미스 연결: AWS Direct Connect 또는 AWS Virtual Private Network 연결을 통해 온프레미스 SFTP 서버에 연결
 - 사용자 지정 IP 주소: 자체 NAT 게이트웨이와 탄력적 IP 주소를 원격 서버에 제공합니다.
 - 중앙 집중식 보안 제어: 조직의 중앙 수신/송신 제어를 통해 파일 전송 라우팅

다음 매트릭스는 사용 사례에 적합한 커넥터 유형을 선택하는 데 도움이 됩니다.

SFTP 커넥터 송신 유형 매트릭스

기능	송신 유형 = 서비스 관리형	송신 유형 = VPC Lattice
퍼블릭 호스팅(인터넷 액세스 가능) SFTP 서버에 대한 연결	지원됨	지원 ¹
프라이빗 호스팅(온프레미스) SFTP 서버에 대한 연결	지원되지 않음	지원됨 ²
프라이빗 호스팅(VPC 내) SFTP 서버에 대한 연결	지원되지 않음	지원됨
원격 SFTP 서버에 표시되는 고정 IP 주소	서비스 제공 고정 IP 주소를 통해 지원됨	고객 소유 고정 IP 주소를 통해 지원됨

기능	송신 유형 = 서비스 관리형	송신 유형 = VPC Lattice
대역폭 사용 가능	계정당 50MBPS	고객 소유 Resource Gateway 및 NAT Gateway에서 사용할 수 있는 더 높은 대역폭
고객 소유 NAT 게이트웨이 및 네트워크 방화벽을 통해 인터넷으로 트래픽 라우팅	지원되지 않음. NAT 게이트웨이는 Transfer Family 서비스에 소유하고 관리합니다.	지원됨

¹ 송신 유형 = VPC Lattice를 사용하면 송신 VPCs의 송신 인프라(NAT 게이트웨이) 설정을 사용하여 공개적으로 호스팅되는 서버에 대한 연결이 지원됩니다.

² 송신 유형 = VPC Lattice를 사용하면 프라이빗 호스팅 서버에 대한 연결이 AWS Direct Connect 또는 VPN과 같은 VPC의 기존 네트워크를 사용하여 지원됩니다.

IP 주소 지정 모드 선택

서비스 관리형 송신으로 SFTP 커넥터를 생성할 때 두 IP 주소 지정 모드 중에서 선택할 수 있습니다.

- IPv4 전용(기본값): 커넥터는 원격 SFTP 서버에 연결하기 위해 IPv4 주소만 사용합니다. 콘솔, AWS CLI 또는 API를 통해 커넥터를 생성할 때 기본 모드입니다.
- 듀얼 스택: 커넥터는 IPv6 및 IPv4 주소를 모두 지원합니다. 듀얼 스택 모드에서 커넥터는 DNS 확인이 IPv6 결과를 반환할 때 IPv6를 선호하고 IPv4 DNS 결과만 반환될 때 IPv4를 사용합니다.

Note

IP 주소 지정 모드는 서비스 관리형 송신 유형의 커넥터에만 적용됩니다. VPC Lattice 송신을 사용하는 커넥터는 이 설정을 지원하지 않습니다.

주제

- [Secrets Manager에 SFTP 커넥터의 인증 자격 증명 저장](#)
- [서비스 관리형 송신을 사용하여 SFTP 커넥터 생성](#)
- [VPC 기반 송신을 사용하여 SFTP 커넥터 생성](#)
- [SFTP 커넥터를 테스트합니다.](#)

Secrets Manager에 SFTP 커넥터의 인증 자격 증명 저장

Secrets Manager를 사용하여 SFTP 커넥터의 사용자 자격 증명을 저장할 수 있습니다. 암호를 생성할 때 사용자 이름을 입력해야 합니다. 또한 암호, 프라이빗 키 또는 둘 다 제공할 수 있습니다. 자세한 내용은 [Transfer for SFTP 커넥터 할당량](#)을 참조하세요.

Note

Secrets Manager에 암호를 저장하면에 요금이 AWS 계정 발생합니다. 요금에 대한 자세한 내용은 [AWS Secrets Manager 요금](#)을 참조하세요.

SFTP 커넥터의 사용자 자격 증명을 Secrets Manager에 저장하려면

1. 에 로그인 AWS Management Console 하고 <https://console.aws.amazon.com/secretsmanager/> AWS Secrets Manager 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 암호를 선택합니다.
3. 암호 페이지에서 새 암호 저장을 선택합니다.
4. 암호 선택 페이지의 암호 타입에서 다른 타입의 암호를 선택합니다.
5. 보안 암호의 키/값 정보를 제공합니다. 사용자 이름과 프라이빗 키 또는 암호를 제공해야 합니다.
 - a. 키/값 쌍 섹션에서 키/값 탭을 선택합니다.
 - 키 - **Username**를 입력합니다.
 - value - 파트너의 서버에 연결할 권한이 있는 사용자의 이름을 입력합니다.
 - b. 키 페어를 제공하려면 행 추가를 선택하고 키/값 페어 섹션에서 키/값 탭을 선택합니다.
 - 키 - **PrivateKey**를 입력합니다.
 - value - 프라이빗 키에 붙여 넣습니다.

팁: 입력하는 프라이빗 키 데이터는 원격 SFTP 서버에이 사용자에게 대해 저장된 퍼블릭 키와 일치해야 합니다.

Note

AWS Transfer Family SFTP 커넥터를 사용한 인증에는 암호로 보호되는 프라이빗 키를 사용할 수 없습니다.

퍼블릭/프라이빗 키 페어를 생성하는 방법에 대한 자세한 내용은 섹션을 참조하세요 [macOS, Linux 또는 Unix에서 SSH 키 생성](#).

- c. 암호를 제공하려면 행 추가를 선택하고 키/값 쌍 섹션에서 키/값 탭을 선택합니다.
 - 키 - **Password**를 입력합니다.
 - 값 - 사용자의 암호를 입력합니다.
6. 다음을 선택합니다.
7. 암호 구성 페이지에 암호를 위한 명칭과 설명을 입력합니다. 이름에 **aws/transfer/**이라는 접두사를 사용하는 것이 좋습니다. 예컨대, 암호 **aws/transfer/connector-1**을 지정할 수 있습니다.
8. 다음을 선택한 후 교체 구성 페이지의 기본값을 그대로 사용합니다. 그 다음 다음을 선택합니다.
9. 검토 페이지에서 저장을 선택하여 암호를 만들고 저장합니다.

서비스 관리형 송신을 사용하여 SFTP 커넥터 생성

이 절차에서는 AWS Transfer Family 콘솔 또는를 사용하여 SFTP 커넥터를 생성하는 방법을 설명합니다 AWS CLI.

Console

SFTP 커넥터를 생성하려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 SFTP 커넥터를 선택한 다음 SFTP 커넥터 생성을 선택합니다.
3. 커넥터 구성 섹션의 송신 유형에서 서비스 관리를 선택합니다. 이 옵션은 AWS Transfer Family 관리형 송신 인프라를 사용합니다. Transfer Family 서비스는 각 SFTP 커넥터에 대한 고정 IP 주소를 제공하고 관리합니다.
4. 커넥터 구성 섹션에서 다음 정보를 제공합니다:

Connector configuration

Egress type [Info](#)
Choose your SFTP connector's egress type: either directly over public internet, or via your VPC environment

☒ **Service managed**
Connect to public endpoints over the internet. The connector will present service-provided static IP addresses to remote server.

☐ **VPC Lattice**
Connect to public or private endpoints through your VPC. The connector will present elastic IP address from your VPC's CIDR range to remote server.

URL
Enter the URL of remote server to which you need to connect using the SFTP connector

`sftp://sftp.example.com`

URL of the remote (target) SFTP server

Access role [Info](#)
IAM Role for Amazon S3 access and AWS Secrets Manager access

Choose an IAM role

Logging role - optional [Info](#)
IAM role for the connector to push events to your CloudWatch logs

Choose a logging role

- URL에는 원격 SFTP 서버의 URL을 입력합니다. 이 URL은 `sftp://partner-SFTP-server-url`와 같은 형식(예: `sftp://AnyCompany.com`)이어야 합니다.

Note

경우에 따라 URL에 포트 번호를 제공할 수 있습니다. 형식은 `sftp://partner-SFTP-server-url:port-number`입니다. 기본 포트 번호(지정된 포트가 없는 경우)는 포트 22입니다.

- 액세스 역할에서 사용할 (IAM) 역할의 Amazon 리소스 이름 AWS Identity and Access Management (ARN)을 선택합니다.
- 이 역할이 StartFileTransfer 요청에서 사용된 파일 위치의 상위 디렉터리에 대한 읽기 및 쓰기 액세스 권한을 이 역할이 제공하는지 확인하세요.
- 이 역할이 `secretsmanager:GetSecretValue`에 대해 암호에 액세스할 수 있는 권한을 제공하는지 확인하세요.

Note

정책에서 보안 암호의 ARN을 지정해야 합니다. ARN에는 보안 암호 이름이 포함되지만 이름에 임의의 영숫자 6자를 추가합니다. 보안 암호의 ARN 형식은 다음과 같습니다.

```
arn:aws:secretsmanager:region:account-id:secret:aws/
transfer/SecretName-6RandomCharacters
```

- 사용자의 이전 요청을 처리할 때 커넥터가 리소스에 액세스할 수 있도록 허용하는 신뢰 관계가 이 역할에 포함되어 있는지 확인하세요. 신뢰 관계 설정에 대한 자세한 내용은 [신뢰 관계를 구축하기 위해](#)을 참조하세요.

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

```

    "Resource": "arn:aws:secretsmanager:us-
west-2:111122223333:secret:aws/transfer/SecretName-6RandomCharacters"
  }
]
}

```

Note

액세스 역할의 경우, 예는 단일 암호에 대한 액세스 권한을 부여합니다. 하지만 와일드카드 문자를 사용하면 여러 사용자 및 암호에 대해 동일한 IAM 역할을 재사용하려는 경우 작업을 줄일 수 있습니다. 예컨대, 다음 리소스 명령문은 aws/transfer으로 시작하는 이름을 가진 모든 암호에 대한 권한을 부여합니다.

```

"Resource": "arn:aws:secretsmanager:region:account-id:secret:aws/
transfer/*"

```

SFTP 자격 증명이 포함된 암호를 다른 AWS 계정에 저장할 수도 있습니다. 교차 계정 보안 암호 액세스를 활성화하는 방법에 대한 자세한 내용은 [다른 계정의 사용자에 대한 AWS Secrets Manager 보안 암호 권한을 참조하세요](#).

5. 커넥터 구성을 완료합니다.

- (옵션) 로깅 역할에서 커넥터가 CloudWatch 로그에 이벤트를 푸시하는 데 사용할 IAM 역할을 선택합니다. 다음 예제 정책은 SFTP 커넥터에 대한 이벤트를 로깅하는 데 필요한 권한을 나열합니다.

```

{
  "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/*"
    }
  ]
}

```

```
]
}
```

6. SFTP 구성 섹션에 다음 정보를 제공합니다.

SFTP configuration Info

Connector credentials Info
 Select a secret from AWS Secrets Manager containing the username and SSH private key, or username and password, to be used to connect to the remote server

aws/transfer/sftp-connector1 ▼
 ↻
Store a new secret ↗

Trusted host keys - optional Info
 Enter the public portion of the host key(s) that is used to identify the remote server you need to connect to. You can enter the trusted host key(s) now, or add them after creating the connector by using the host key information returned by the TestConnection action.

ⓘ Connections to the remote server are established only if it's SSH fingerprint matches one of the provided trusted host key(s).

ssh-rsa AAAA:
 Remove

Add trusted host key

Maximum concurrent connections Info
 Select the maximum number of concurrent connections that your connector can create to the remote server

5 ▼

- 커넥터 자격 증명의 드롭다운 목록에서 SFTP 사용자의 프라이빗 키 또는 암호 AWS Secrets Manager 가 포함된의 보안 암호 이름을 선택합니다. 암호를 생성하여 특정 방식으로 저장해야 합니다. 자세한 내용은 [Secrets Manager에 SFTP 커넥터의 인증 자격 증명 저장](#)을 참조하세요.
- (선택 사항) TrustedHostKeys 파라미터를 비워 두고 커넥터를 생성할 수 있습니다. 그러나 커넥터 구성에이 파라미터를 제공할 때까지 커넥터가 원격 서버로 파일을 전송할 수 없습니다. 커넥터를 생성할 때 신뢰할 수 있는 호스트 키(들)를 입력하거나 나중에 TestConnection 콘솔 작업 또는 API 명령에서 반환된 호스트 키 정보를 사용하여 커넥터를 업데이트할 수 있습니다. 즉, 신뢰할 수 있는 호스트 키 텍스트 상자에 대해 다음 중 하나를 수행할 수 있습니다.
 - 커넥터를 생성할 때 신뢰할 수 있는 호스트 키(들)를 제공합니다. 외부 서버를 식별하는 데 사용되는 호스트 키의 퍼블릭 부분에 붙여 넣습니다. 신뢰 호스트 키 추가를 선택하여 키를 더 추가하면 키를 두 개 이상 추가할 수 있습니다. SFTP 서버에 대해 ssh-keyscan 명령을 사용하여 필요한 키를 검색할 수 있습니다. Transfer Family가 지원하는 신뢰 호스트 키의 형식 및 타입에 대한 자세한 설명은 [SFTPConnectorConfig](#) 섹션을 참조하세요.
 - 커넥터를 생성할 때 신뢰할 수 있는 호스트 키(들) 텍스트 상자를 비워 두고 나중에이 정보로 커넥터를 업데이트합니다. 커넥터를 생성할 때 호스트 키 정보가 없는 경우 지금은이 파라미터를 비워 두고 커넥터 생성을 진행할 수 있습니다. 커넥터를 생성한 후 새 커넥터

의 ID를 사용하여 AWS CLI 또는 커넥터의 세부 정보 페이지에서 TestConnection 명령을 실행합니다. 성공하면 TestConnection가 필요한 호스트 키 정보를 반환합니다. 그런 다음 콘솔을 사용하여(또는 UpdateConnector AWS CLI 명령을 실행하여) 커넥터를 편집하고 실행 시 반환된 호스트 키 정보를 추가할 수 있습니다 TestConnection.

Important

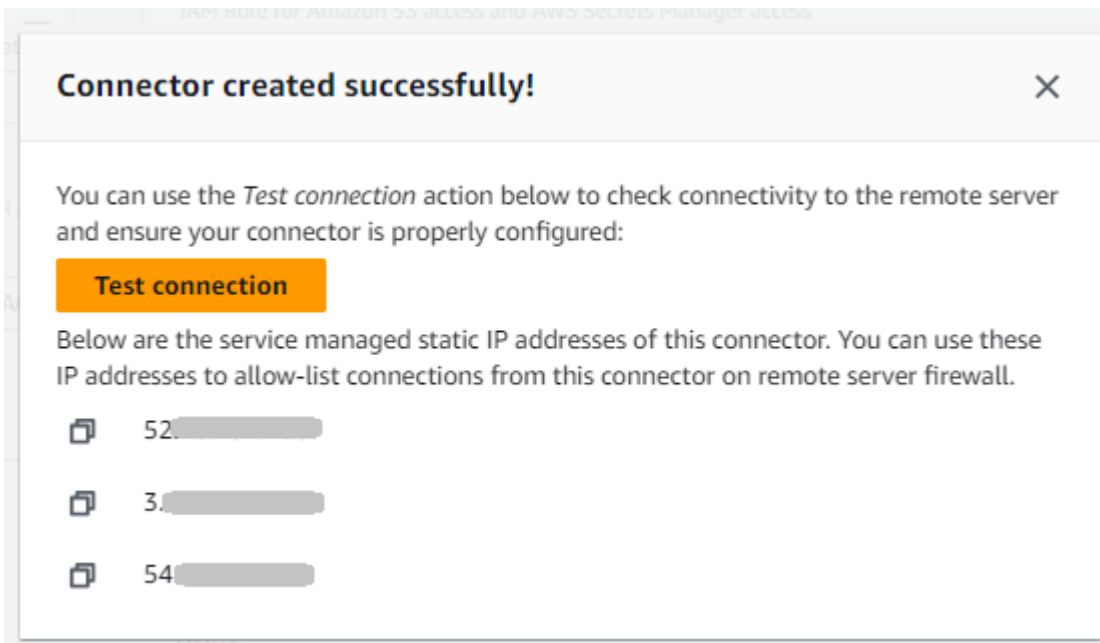
를 실행하여 원격 서버의 호스트 키를 검색하는 경우 반환된 키에 대해 out-of-band 검증을 수행해야 TestConnection합니다.

새 키를 신뢰할 수 있는 것으로 수락하거나 연결하려는 원격 SFTP 서버의 소유자로부터 받은 이전에 알려진 지문으로 제시된 지문을 확인해야 합니다.

- (선택 사항) 최대 동시 연결의 드롭다운 목록에서 커넥터가 원격 서버에 생성하는 동시 연결 수를 선택합니다. 콘솔의 기본 선택은 5입니다.

이 설정은 커넥터가 원격 서버와 동시에 설정할 수 있는 활성 연결 수를 지정합니다. 동시 연결을 생성하면 병렬 작업을 활성화하여 커넥터 성능을 향상시킬 수 있습니다.

7. 암호화 알고리즘 옵션 섹션의 보안 정책 필드의 드롭다운 목록에서 보안 정책을 선택합니다. 보안 정책을 사용하면 커넥터가 지원하는 암호화 알고리즘을 선택할 수 있습니다. 사용 가능한 보안 정책 및 알고리즘에 대한 자세한 내용은 섹션을 참조하세요 [AWS Transfer Family SFTP 커넥터에 대한 보안 정책](#).
8. (옵션) 태그 섹션에서 키 및 값에 하나 이상의 태그를 키-값 쌍을 입력합니다.
9. 모든 설정을 확인한 후 SFTP 커넥터 생성을 선택하여 SFTP 커넥터를 생성합니다. 커넥터가 성공적으로 생성되면 할당된 고정 IP 주소 목록과 연결 테스트 버튼이 있는 화면이 나타납니다. 버튼을 사용하여 새 커넥터의 구성을 테스트합니다.



새 SFTP 커넥터의 ID가 목록에 추가된 커넥터 페이지가 나타납니다. 커넥터에 대한 세부 정보를 보려면 [SFTP 커넥터 세부 정보 보기](#) 섹션을 참조하세요.

CLI

커넥터를 생성할 때는 [create-connector](#) 명령을 사용합니다. 이 명령을 사용하여 SFTP 커넥터를 생성하려면 다음 정보를 제공해야 합니다.

- 원격 SFTP 서버의 URL입니다. 이 URL은 `sftp://partner-SFTP-server-url`와 같은 형식 (예: `sftp://AnyCompany.com`)이어야 합니다.
- 액세스 역할입니다. 사용할 AWS Identity and Access Management (IAM) 역할의 Amazon 리소스 이름(ARN)을 선택합니다.
- 이 역할이 `StartFileTransfer` 요청에서 사용된 파일 위치의 상위 디렉터리에 대한 읽기 및 쓰기 액세스 권한을 이 역할이 제공하는지 확인하세요.
- 이 역할이 `secretsmanager:GetSecretValue`에 대해 암호에 액세스할 수 있는 권한을 제공하는지 확인하세요.

Note

정책에서 보안 암호의 ARN을 지정해야 합니다. ARN에는 보안 암호 이름이 포함되지만 이름에 임의의 영숫자 6자를 추가합니다. 보안 암호의 ARN 형식은 다음과 같습니다.

```
arn:aws:secretsmanager:region:account-id:secret:aws/
transfer/SecretName-6RandomCharacters
```

- 사용자의 이전 요청을 처리할 때 커넥터가 리소스에 액세스할 수 있도록 허용하는 신뢰 관계가 이 역할에 포함되어 있는지 확인하세요. 신뢰 관계 설정에 대한 자세한 내용은 [신뢰 관계를 구축하기 위해](#)을 참조하세요.

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

```

```

    "Resource": "arn:aws:secretsmanager:us-west-2:111122223333:secret:aws/
transfer/SecretName-6RandomCharacters"
  }
]
}

```

Note

액세스 역할의 경우, 예는 단일 암호에 대한 액세스 권한을 부여합니다. 하지만 와일드카드 문자를 사용하면 여러 사용자 및 암호에 대해 동일한 IAM 역할을 재사용하려는 경우 작업을 줄일 수 있습니다. 예컨대, 다음 리소스 명령문은 aws/transfer으로 시작하는 이름을 가진 모든 암호에 대한 권한을 부여합니다.

```

"Resource": "arn:aws:secretsmanager:region:account-id:secret:aws/transfer/
*"

```

SFTP 자격 증명이 포함된 암호를 다른 AWS 계정에 저장할 수도 있습니다. 교차 계정 보안 암호 액세스를 활성화하는 방법에 대한 자세한 내용은 [다른 계정의 사용자에게 대한 AWS Secrets Manager 보안 암호 권한을 참조하세요](#).

- (옵션) CloudWatch 로그에 이벤트를 푸시하는 데 사용할 커넥터의 IAM 역할을 선택합니다. 다음 예제 정책은 SFTP 커넥터에 대한 이벤트를 로깅하는 데 필요한 권한을 나열합니다.

```

{
  "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/*"
    }
  ]
}

```

- 다음 SFTP 구성 정보를 제공하세요.
 - SFTP 사용자의 프라이빗 키 또는 암호 AWS Secrets Manager 가 포함된의 보안 암호 ARN입니다.
 - 외부 서버를 식별하는 데 사용되는 호스트 키의 공개 부분입니다. 원하는 경우 신뢰 호스트 키를 여러 개 제공할 수 있습니다.

SFTP 정보를 제공하는 가장 쉬운 방법은 SFTP 정보를 파일에 저장하는 것입니다. 예컨대, 다음 예 텍스트를 testSFTPConfig.json이라는 파일에 복사합니다.

```
// 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="
  ]
}
```

- 커넥터의 보안 정책을 지정하고 보안 정책 이름을 입력합니다.

Note

SecretId는 전체 ARN이거나 암호 이름(이전 목록의 *example-username-key*)일 수 있습니다.

그런 다음 다음 명령을 실행하여 커넥터를 생성합니다.

```
aws transfer create-connector --url "sftp://partner-SFTP-server-url" \
--access-role your-IAM-role-for-bucket-access \
--logging-role arn:aws:iam::your-account-id:role/service-role/AWSTransferLoggingAccess \
--sftp-config file:///path/to/testSFTPConfig.json \
--security-policy-name security-policy-name \
--maximum-concurrent-connections integer-from-1-to-5
```

VPC 송신 유형 커넥터를 설명하면 응답에 새 필드가 포함됩니다.

```
{
  "Connector": {
```

```

    "AccessRole": "arn:aws:iam::123456789012:role/connector-role",
    "Arn": "arn:aws:transfer:us-east-1:123456789012:connector/
c-1234567890abcdef0",
    "ConnectorId": "c-1234567890abcdef0",
    "Status": "ACTIVE",
    "EgressConfig": {
      "VpcLattice": {
        "ResourceConfigurationArn": "arn:aws:vpc-lattice:us-
east-1:123456789012:resourceconfiguration/rcfg-12345678",
        "PortNumber": 22
      }
    },
    "EgressType": "VPC",
    "ServiceManagedEgressIpAddresses": null,
    "SftpConfig": {
      "TrustedHostKeys": [ "ssh-rsa AAAAB3NzaC..." ],
      "UserSecretId": "aws/transfer/connector-secret"
    },
    "Url": "sftp://my.sftp.server.com:22"
  }
}

```

트래픽ServiceManagedEgressIpAddresses은 AWS 관리형 인프라 대신 VPC를 통해 라우팅 되므로 VPC 송신 유형 커넥터의 경우는 null입니다.

VPC 기반 송신을 사용하여 SFTP 커넥터 생성

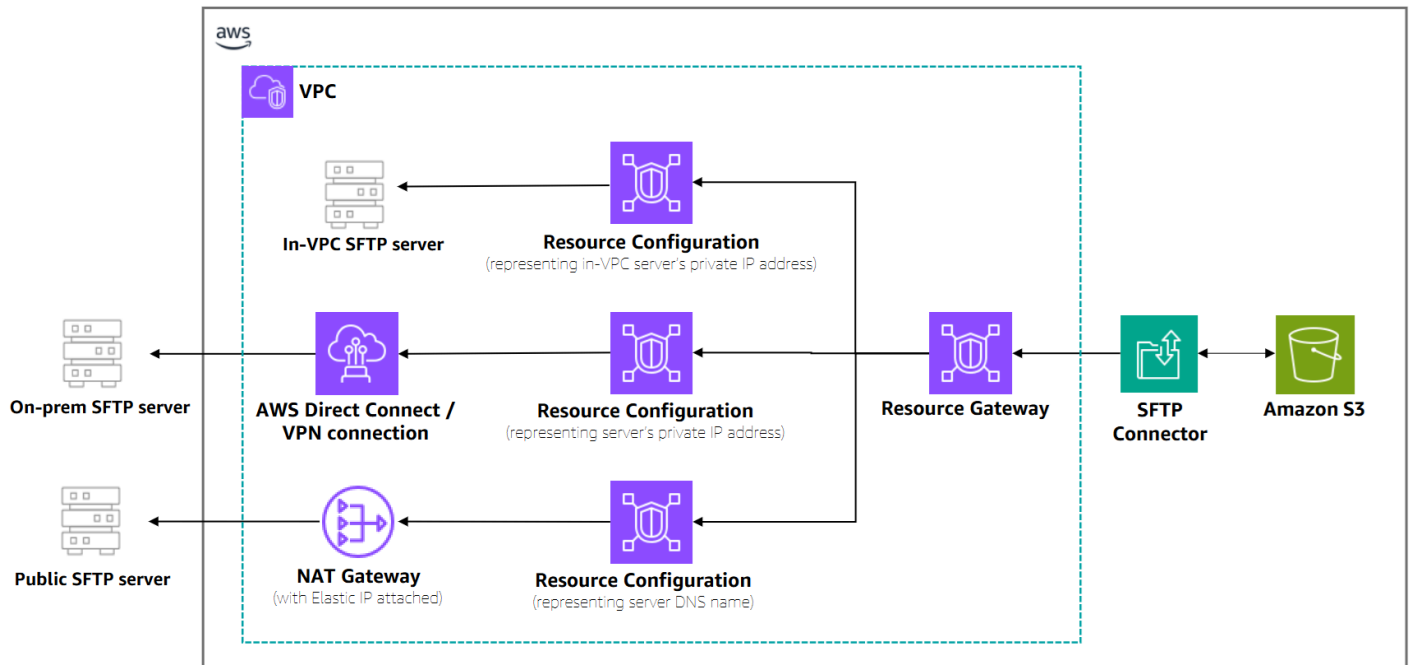
이 주제에서는 VPC 연결을 사용하여 SFTP 커넥터를 생성하기 위한 step-by-step 지침을 제공합니다. VPC_LATTICE 지원 커넥터는 Amazon VPC Lattice를 사용하여 가상 프라이빗 클라우드를 통해 트래픽을 라우팅하므로 프라이빗 엔드포인트에 안전하게 연결하거나 인터넷 액세스를 위해 자체 NAT 게이트웨이를 사용할 수 있습니다.

VPC 연결을 사용해야 하는 경우

다음 시나리오에서는 SFTP 커넥터에 VPC 연결을 사용합니다.

- 프라이빗 SFTP 서버: VPC에서만 액세스할 수 있는 SFTP 서버에 연결합니다.
- 온프레미스 연결: AWS Direct Connect 또는 AWS Site-to-Site VPN 연결을 통해 온프레미스 SFTP 서버에 연결합니다.
- 사용자 지정 IP 주소: BYOIP 시나리오를 포함하여 자체 NAT 게이트웨이와 탄력적 IP 주소를 사용합니다.

- 중앙 집중식 보안 제어: 조직의 중앙 수신/송신 제어를 통해 파일 전송을 라우팅합니다.



VPC_LATTICE 지원 SFTP 커넥터의 사전 조건

VPC_LATTICE 지원 SFTP 커넥터를 생성하기 전에 다음 사전 조건을 완료해야 합니다.

VPC 기반 연결 작동 방식

VPC Lattice를 사용하면 VPC 리소스를 다른 AWS 서비스와 안전하게 공유할 수 있습니다. AWS Transfer Family 는 서비스 네트워크를 사용하여 리소스 공유 프로세스를 간소화합니다. 주요 구성 요소는 다음과 같습니다.

- Resource Gateway: VPC에 대한 액세스 지점 역할을 합니다. 최소 2개의 가용 영역으로 VPC에서 이를 생성합니다.
- 리소스 구성: 연결하려는 SFTP 서버의 프라이빗 IP 주소 또는 퍼블릭 DNS 이름을 포함합니다.

VPC_LATTICE 지원 커넥터를 생성할 때는 전달 액세스 세션(FAS)을 AWS Transfer Family 사용하여 일시적으로 자격 증명을 가져오고 리소스 구성을 서비스 네트워크와 연결합니다.

필수 설정 단계

1. VPC 인프라: SFTP 서버 연결 요구 사항에 필요한 서브넷, 라우팅 테이블 및 보안 그룹이 있는 VPC가 올바르게 구성되어 있는지 확인합니다.
2. Resource Gateway: VPC Lattice create-resource-gateway 명령을 사용하여 VPC에서 Resource Gateway를 생성합니다. Resource Gateway는 두 개 이상의 가용 영역에 있는 서브넷과 연결되어야 합니다. 자세한 내용은 Amazon VPC Lattice 사용 설명서의 [리소스 게이트웨이](#)를 참조하세요.
3. 리소스 구성: VPC Lattice create-resource-configuration 명령을 사용하여 대상 SFTP 서버를 나타내는 리소스 구성을 생성합니다. 다음 중 하나를 지정할 수 있습니다.
 - 프라이빗 엔드포인트의 프라이빗 IP 주소
 - 퍼블릭 엔드포인트의 퍼블릭 DNS 이름(퍼블릭 엔드포인트에는 IP 주소가 지원되지 않음)
4. 인증 자격 증명:에 설명된 AWS Secrets Manager 대로 SFTP 사용자 자격 증명을에 저장합니다 [Secrets Manager에 SFTP 커넥터의 인증 자격 증명 저장](#).

Important

Resource Gateway 및 리소스 구성은 동일한 AWS 계정에서 생성해야 합니다. 리소스 구성을 생성할 때 먼저 Resource Gateway가 있어야 합니다.

VPC 리소스 구성에 대한 자세한 내용은 Amazon VPC Lattice 사용 설명서의 [리소스 구성](#)을 참조하세요.

Note

SFTP 커넥터에 대한 VPC 연결은 Amazon VPC Lattice 리소스를 사용할 수 AWS 리전 있는에서 사용할 수 있습니다. 자세한 내용은 [VPC Lattice FAQs](#). 가용 영역 지원은 리전에 따라 다르며 Resource Gateway에는 최소 2개의 가용 영역이 필요합니다.

VPC_LATTICE 지원 SFTP 커넥터 생성

사전 조건을 완료한 후, AWS CLI AWS 관리 콘솔 또는 AWS SDKs.

Console

VPC_LATTICE 지원 SFTP 커넥터를 생성하려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 SFTP 커넥터를 선택한 다음 SFTP 커넥터 생성을 선택합니다.
3. 커넥터 구성 섹션의 송신 유형에서 VPC Lattice를 선택합니다.

이 옵션은 VPC 간 리소스 액세스를 위해 Amazon VPC Lattice를 사용하여 VPC를 통해 트래픽을 라우팅합니다. 이 옵션을 사용하여 프라이빗 호스팅 서버 엔드포인트에 연결하거나, VPC의 보안 제어를 통해 트래픽을 라우팅하거나, 자체 NAT 게이트웨이 및 탄력적 IP 주소를 사용할 수 있습니다. 원격 SFTP 서버의 주소는 VPC에서 리소스 구성으로 표시됩니다. 리소스 구성에 대한 자세한 내용은 Amazon [VPC Lattice 사용 설명서의 VPC 리소스에 대한 리소스 구성](#)을 참조하세요.

4. 커넥터 구성을 완료합니다.
 - 액세스 역할에서 사용할 (IAM) 역할의 Amazon 리소스 이름 AWS Identity and Access Management (ARN)을 선택합니다.
 - 이 역할이 StartFileTransfer 요청에서 사용된 파일 위치의 상위 디렉터리에 대한 읽기 및 쓰기 액세스 권한을 이 역할이 제공하는지 확인하세요.
 - 이 역할이 secretsmanager:GetSecretValue에 대해 암호에 액세스할 수 있는 권한을 제공하는지 확인하세요.

Note

정책에서 보안 암호의 ARN을 지정해야 합니다. ARN에는 보안 암호 이름이 포함되지만 이름에 임의의 영숫자 6자를 추가합니다. 보안 암호의 ARN 형식은 다음과 같습니다.

```
arn:aws:secretsmanager:region:account-id:secret:aws/transfer/SecretName-6RandomCharacters
```

- 사용자의 이전 요청을 처리할 때 커넥터가 리소스에 액세스할 수 있도록 허용하는 신뢰 관계가 이 역할에 포함되어 있는지 확인하세요. 신뢰 관계 설정에 대한 자세한 내용은 [신뢰 관계를 구축하기 위해](#)을 참조하세요.

```
{
```



```

"Version": "2012-10-17",
"Statement": [
  {
    "Sid": "AllowListingOfUserFolder",
    "Action": [
      "s3:ListBucket",
      "s3:GetBucketLocation"
    ],
    "Effect": "Allow",
    "Resource": [
      "arn:aws:s3:::amzn-s3-demo-bucket"
    ]
  },
  {
    "Sid": "HomeDirObjectAccess",
    "Effect": "Allow",
    "Action": [
      "s3:PutObject",
      "s3:GetObject",
      "s3:DeleteObject",
      "s3:DeleteObjectVersion",
      "s3:GetObjectVersion",
      "s3:GetObjectACL",
      "s3:PutObjectACL"
    ],
    "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/*"
  },
  {
    "Sid": "GetConnectorSecretValue",
    "Effect": "Allow",
    "Action": [
      "secretsmanager:GetSecretValue"
    ],
    "Resource": "arn:aws:secretsmanager:us-  
west-2:111122223333:secret:aws/transfer/SecretName-6RandomCharacters"
  }
]
}

```

 Note

액세스 역할의 경우, 예는 단일 암호에 대한 액세스 권한을 부여합니다. 하지만 와일드카드 문자를 사용하면 여러 사용자 및 암호에 대해 동일한 IAM 역할을 재사용하려

는 경우 작업을 줄일 수 있습니다. 예컨대, 다음 리소스 명령문은 aws/transfer으로 시작하는 이름을 가진 모든 암호에 대한 권한을 부여합니다.

```
"Resource": "arn:aws:secretsmanager:region:account-id:secret:aws/transfer/*"
```

SFTP 자격 증명이 포함된 암호를 다른 AWS 계정에 저장할 수도 있습니다. 교차 계정 보안 암호 액세스 활성화에 대한 자세한 내용은 [다른 계정의 사용자에게 대한 AWS Secrets Manager 보안 암호 권한을 참조하세요](#).

- 리소스 구성 ARN에 SFTP 서버를 가리키는 VPC Lattice 리소스 구성의 ARN을 입력합니다.

```
arn:aws:vpc-lattice:region:account-id:resourceconfiguration/rcfg-12345678
```

- (옵션) 로깅 역할에서 커넥터가 CloudWatch 로그에 이벤트를 푸시하는 데 사용할 IAM 역할을 선택합니다.

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

5. SFTP 구성 섹션에 다음 정보를 제공합니다.

- 커넥터 자격 증명에서 SFTP 사용자의 프라이빗 키 또는 암호 AWS Secrets Manager가 포함된 보안 암호 이름을 선택합니다.
- 신뢰할 수 있는 호스트 키의 경우 외부 서버를 식별하는 데 사용되는 호스트 키의 퍼블릭 부분에 붙여넣거나 나중에 TestConnection 명령을 사용하여 구성하려면 비워 둡니다.

이 호스트 키는 VPC_LATTICE 커넥터용이므로 키에서 호스트 이름을 제거합니다.

- (선택 사항) 최대 동시 연결에서 커넥터가 원격 서버에 생성하는 동시 연결 수를 선택합니다 (기본값은 5).
- 6. 암호화 알고리즘 옵션 섹션의 드롭다운 목록에서 보안 정책을 선택합니다.
- 7. (선택 사항) 태그 섹션에서 태그를 키-값 페어로 추가합니다.
- 8. SFTP 커넥터 생성을 선택하여 VPC_LATTICE 지원 SFTP 커넥터를 생성합니다.

리소스 연결이 프로비저닝되는 PENDING 동안 커넥터가 상태로 생성되며, 일반적으로 몇 분 정도 걸립니다. 상태가 로 변경되면 커넥터를 사용할 준비가 ACTIVE된 것입니다.

CLI

다음 명령을 사용하여 VPC_LATTICE 지원 SFTP 커넥터를 생성합니다.

```
aws transfer create-connector \
  --url "sftp://my.sftp.server.com:22" \
  --access-role arn:aws:iam::123456789012:role/TransferConnectorRole \
  --sftp-config UserSecretId=my-secret-id,TrustedHostKeys="ssh-rsa AAAAB3NzaC..." \
  --egress-config VpcLattice={ResourceConfigurationArn=arn:aws:vpc-lattice:us-east-1:123456789012:resourceconfiguration/rcfg-1234567890abcdef0} \
  --security-policy-name TransferSecurityPolicy-2024-01
```

VPC 연결의 키 파라미터는 SFTP 서버 대상을 정의하는 리소스 구성 ARN을 --egress-config 지정하는입니다.

VPC 커넥터 상태 모니터링

VPC_LATTICE 지원 커넥터에는 비동기식 설정 프로세스가 있습니다. 생성 후 커넥터 상태를 모니터링합니다.

- 보류 중: 커넥터가 프로비저닝되고 있습니다. 서비스 네트워크 프로비저닝이 진행 중이며, 일반적으로 몇 분 정도 걸립니다.
- ACTIVE: 커넥터를 사용할 준비가 되었으며 파일을 전송할 수 있습니다.
- ERRORED: 커넥터가 프로비저닝되지 않았습니다. 오류 세부 정보에서 문제 해결 정보를 확인하세요.

`describe-connector` 명령을 사용하여 커넥터 상태를 확인합니다.

```
aws transfer describe-connector --connector-id c-1234567890abcdef0
```

PENDING 상태에서 `test-connection` API는 프로비저닝이 완료될 때까지 "Connector not available"를 반환합니다.

제한 사항 및 고려 사항

- 퍼블릭 엔드포인트: VPC를 통해 퍼블릭 엔드포인트에 연결할 때 리소스 구성에 DNS 이름을 제공해야 합니다. 퍼블릭 IP 주소는 지원되지 않습니다.
- 리전별 가용성: 일부에서 VPC 연결을 사용할 수 있습니다 AWS 리전. 교차 리전 리소스 공유는 지원되지 않습니다.
- 가용 영역 요구 사항: Resource Gateway는 두 개 이상의 가용 영역에 있는 서브넷과 연결되어야 합니다. 모든 가용 영역이 모든 리전에서 VPC Lattice를 지원하는 것은 아닙니다.
- 연결 제한: TCP 연결의 유효 제한 시간이 350초인 리소스당 최대 350개의 연결.

비용 고려 사항

일반 서비스 요금 AWS Transfer Family 외의 추가 요금은 없습니다. 그러나 고객은 Amazon Virtual Private Cloud 리소스 공유와 관련된 Amazon VPC Lattice의 추가 요금이 부과될 수 있으며, 인터넷으로 송신하기 위해 자체 NAT 게이트웨이를 사용하는 경우 NAT 게이트웨이 요금이 부과될 수 있습니다.

전체 AWS Transfer Family 요금 정보는 [AWS Transfer Family 요금 페이지](#)를 참조하세요.

SFTP 커넥터의 VPC 연결 예제

이 섹션에서는 다양한 시나리오에 대해 VPC 연결을 사용하여 SFTP 커넥터를 생성하는 예를 제공합니다. 이러한 예제를 사용하기 전에 VPC 연결 설명서에 설명된 대로 VPC 인프라 설정을 완료했는지 확인합니다.

예: 프라이빗 엔드포인트 연결

이 예제에서는 VPC에서만 액세스할 수 있는 프라이빗 SFTP 서버에 연결하는 SFTP 커넥터를 생성하는 방법을 보여줍니다.

사전 조건

1. VPC에서 Resource Gateway를 생성합니다.

```
aws vpc-lattice create-resource-gateway \
  --name my-private-server-gateway \
  --vpc-identifier vpc-1234567890abcdef0 \
  --subnet-ids subnet-1234567890abcdef0 subnet-0987654321fedcba0
```

2. 프라이빗 SFTP 서버에 대한 리소스 구성을 생성합니다.

```
aws vpc-lattice create-resource-configuration \
  --name my-private-server-config \
  --resource-gateway-identifier rgw-1234567890abcdef0 \
  --resource-configuration-definition ipResource={ipAddress="10.0.1.100"} \
  --port-ranges 22
```

VPC_LATTICE 지원 커넥터 생성

1. VPC 연결을 사용하여 SFTP 커넥터를 생성합니다.

```
aws transfer create-connector \
  --access-role arn:aws:iam::123456789012:role/TransferConnectorRole \
  --sftp-config UserSecretId=my-private-server-credentials,TrustedHostKeys="ssh-
rsa AAAAB3NzaC..." \
  --egress-config VpcLattice={ResourceConfigurationArn=arn:aws:vpc-lattice:us-
east-1:123456789012:resourceconfiguration/rcfg-1234567890abcdef0,PortNumber=22}
```

2. 커넥터 상태가 될 때까지 모니터링합니다ACTIVE.

```
aws transfer describe-connector --connector-id c-1234567890abcdef0
```

원격 SFTP 서버에는 VPC CIDR 범위 내에서 Resource Gateway의 IP 주소에서 오는 연결이 표시됩니
다.

예: VPC를 통한 퍼블릭 엔드포인트

이 예제에서는 VPC를 통해 퍼블릭 SFTP 서버로 연결을 라우팅하여 중앙 집중식 보안 제어를 활용하
고 자체 NAT 게이트웨이 IP 주소를 사용하는 방법을 보여줍니다.

사전 조건

1. VPC에서 Resource Gateway를 생성합니다(프라이빗 엔드포인트 예제와 동일).

2. DNS 이름을 사용하여 퍼블릭 SFTP 서버에 대한 리소스 구성을 생성합니다.

```
aws vpc-lattice create-resource-configuration \
  --name my-public-server-config \
  --resource-gateway-identifier rgw-1234567890abcdef0 \
  --resource-configuration-definition dnsResource={domainName="sftp.example.com"} \
  --port-ranges 22
```

Note

퍼블릭 엔드포인트의 경우 IP 주소가 아닌 DNS 이름을 사용해야 합니다.

커넥터 생성

- SFTP 커넥터를 생성합니다.

```
aws transfer create-connector \
  --access-role arn:aws:iam::123456789012:role/TransferConnectorRole \
  --sftp-config UserSecretId=my-public-server-credentials,TrustedHostKeys="ssh-
rsa AAAAB3NzaC..." \
  --egress-config VpcLattice={ResourceConfigurationArn=arn:aws:vpc-lattice:us-
east-1:123456789012:resourceconfiguration/rcfg-0987654321fedcba0,PortNumber=22}
```

트래픽은 커넥터에서 Resource Gateway로 흐른 다음 NAT 게이트웨이를 통해 퍼블릭 SFTP 서버에 도달합니다. 원격 서버는 NAT 게이트웨이의 탄력적 IP 주소를 소스로 볼 수 있습니다.

예: 교차 계정 프라이빗 엔드포인트

이 예제에서는 리소스 공유를 사용하여 다른 AWS 계정의 프라이빗 SFTP 서버에 연결하는 방법을 보여줍니다.

Note

와 같은 다른 메커니즘을 통해 VPC 간 리소스 공유가 이미 활성화된 경우 여기에 설명된 리소스 공유를 구성할 필요가 AWS Transit Gateway 없습니다. Transit Gateway 라우팅 테이블과 같은 기존 라우팅 메커니즘은 SFTP 커넥터에서 자동으로 사용됩니다. SFTP 커넥터를 생성하는 계정과 동일한 계정에서만 리소스 구성을 생성하면 됩니다.

계정 A(리소스 공급자) - 리소스 구성 공유

1. 계정 A에서 리소스 게이트웨이 및 리소스 구성을 생성합니다(이전 예제와 동일).
2. Resource Access Manager를 사용하여 계정 B와 AWS 리소스 구성을 공유합니다.

```
aws ram create-resource-share \
  --name cross-account-sftp-share \
  --resource-arns arn:aws:vpc-lattice:us-east-1:111111111111:resourceconfiguration/rcfg-1234567890abcdef0 \
  --principals 222222222222
```

계정 B(리소스 소비자) - 공유 수락 및 사용

1. 리소스 공유 초대를 수락합니다.

```
aws ram accept-resource-share-invitation \
  --resource-share-invitation-arn arn:aws:ram:us-east-1:111111111111:resource-share-invitation/invitation-id
```

2. 계정 B에서 SFTP 커넥터를 생성합니다.

```
aws transfer create-connector \
  --access-role arn:aws:iam::222222222222:role/TransferConnectorRole \
  --sftp-config UserSecretId=cross-account-server-credentials,TrustedHostKeys="ssh-rsa AAAAB3NzaC..." \
  --egress-config VpcLattice={ResourceConfigurationArn=arn:aws:vpc-lattice:us-east-1:111111111111:resourceconfiguration/rcfg-1234567890abcdef0,PortNumber=22}
```

이제 계정 B의 커넥터가 공유 리소스 구성을 통해 계정 A의 프라이빗 SFTP 서버에 액세스할 수 있습니다.

일반적인 문제 해결 시나리오

다음은 VPC_LATTICE 지원 커넥터를 생성할 때 발생하는 일반적인 문제에 대한 해결 방법입니다.

- 커넥터가 보류 중 상태로 멈춤: Resource Gateway가 활성 상태이고 지원되는 가용 영역에 서브넷이 있는지 확인합니다. 커넥터가 여전히 PENDING 상태인 경우 처음에 사용한 것과 동일한 구성 파라미터를 UpdateConnector 사용하여 호출합니다. 그러면 문제를 해결할 수 있는 새 상태 이벤트가 트리거됩니다.

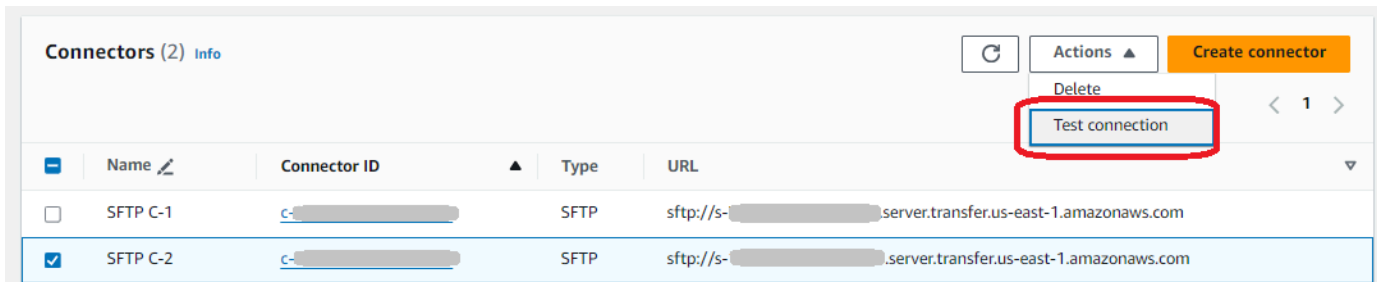
- 연결 제한 시간: 보안 그룹 규칙이 포트 22에서 트래픽을 허용하고 VPC 라우팅이 올바른지 확인합니다.
- DNS 해결 문제: 퍼블릭 엔드포인트의 경우 VPC가 NAT 게이트웨이 또는 인터넷 게이트웨이를 통해 인터넷에 연결되어 있는지 확인합니다.
- 교차 계정 액세스 거부됨: 리소스 공유가 수락되고 리소스 구성 ARN이 올바른지 확인합니다. 오리진 계정이 리소스 공유를 생성할 때 리소스 구성에 적절한 권한 정책이 연결된 경우, `vpc-lattice:AssociateViaAWSService`, `vpc-lattice:AssociateViaAWSService-EventsAndStates`, `vpc-lattice>CreateServiceNetworkResourceAssociation`, 권한이 필요합니다 `vpc-lattice:GetResourceConfiguration`.

SFTP 커넥터를 테스트합니다.

SFTP 커넥터를 만든 후에는 새 커넥터를 사용하여 파일을 전송하기 전에 테스트해 보는 것이 좋습니다.

SFTP 커넥터를 테스트하려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 SFTP 커넥터를 선택하고 커넥터를 선택합니다.
3. 작업 메뉴에서 테스트 연계를 선택합니다.



시스템은 테스트의 통과 또는 실패 여부를 나타내는 메시지를 반환합니다. 테스트가 실패하면 시스템은 테스트 실패 이유를 기반으로 오류 메시지를 제공합니다.

Test connection**✔ Connection succeeded**

Connection with connector ID c-[REDACTED] to the remote SFTP server was successful

Test connection**✖ Unable to assume the configured access role**

Connection with connector ID c-[REDACTED] to the remote SFTP server was unsuccessful

i Note

API를 사용하여 커넥터를 테스트하려면 [TestConnection](#) API 설명서를 참조하세요.

SFTP 커넥터에 대한 VPC 연결

AWS Transfer Family SFTP 커넥터는 Amazon VPC Lattice를 사용하여 VPC 환경을 통해 원격 SFTP 서버에 대한 연결을 지원합니다. 이를 통해 프라이빗 호스팅 SFTP 서버에 연결하거나 VPC의 보안 제어를 통해 인터넷 트래픽을 라우팅하고 자체 NAT 게이트웨이와 탄력적 IP 주소를 사용할 수 있습니다.

송신 유형

SFTP 커넥터는 다음 두 가지 송신 유형 중 하나를 사용할 수 있습니다.

- 서비스 관리형(기본값): 커넥터는 NAT 게이트웨이와가 소유한 IP 주소를 사용하여 퍼블릭 인터넷 AWS Transfer Family 을 통해 연결을 라우팅합니다.
- VPC_LATTICE: 커넥터는 교차 VPC 리소스 액세스를 사용하여 VPC 환경을 통해 트래픽을 라우팅합니다.

VPC 연결을 사용해야 하는 경우

다음 시나리오에서는 SFTP 커넥터에 VPC 연결을 사용합니다.

- 프라이빗 SFTP 서버: VPC에서만 액세스할 수 있는 SFTP 서버에 연결합니다.
- 온프레미스 연결: AWS Direct Connect 또는 AWS Site-to-Site VPN 연결을 통해 온프레미스 SFTP 서버에 연결합니다.
- 사용자 지정 IP 주소: BYOIP 시나리오를 포함하여 자체 NAT 게이트웨이와 탄력적 IP 주소를 사용합니다.
- 중앙 집중식 보안 제어: 조직의 중앙 수신/송신 제어를 통해 파일 전송을 라우팅합니다.

요구 사항

VPC_LATTICE 지원 SFTP 커넥터를 생성하기 전에 다음이 필요합니다.

- VPC 및 관련 인프라(서브넷, 라우팅 테이블, 보안 그룹)
- VPC의 Resource Gateway(최소 2개의 가용 영역)
- 대상 SFTP 서버를 지정하는 리소스 구성

자세한 설정 지침은 섹션을 참조하세요 [VPC_LATTICE 지원 SFTP 커넥터 생성](#). 그리고 예제는 단원을 참조하십시오 [SFTP 커넥터의 VPC 연결 예제](#).

SFTP 커넥터 사용

이 주제에서는 SFTP 커넥터를 사용하여 지원되는 파일 작업을 수행하는 방법을 설명합니다. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔에서 커넥터의 세부 정보를 선택하여 이러한 작업을 수행하는 예제 명령을 찾을 수도 있습니다.

SFTP 커넥터를 생성한 후이 커넥터를 사용하여 연결된 원격 SFTP 서버에서 다음 파일 작업을 수행할 수 있습니다.

- Amazon S3에서 원격 SFTP 서버로 파일을 전송합니다.

- 원격 SFTP 서버에서 Amazon S3로 파일을 검색합니다.
- 원격 SFTP 서버의 디렉터리에서 파일 및 하위 폴더를 나열합니다.
- 원격 SFTP 서버에서 파일 및 디렉터를 삭제, 이름 바꾸기 또는 이동합니다.

커넥터 생성에 대한 자세한 설명은 [SFTP 커넥터 생성](#) 섹션을 참조하세요.

주제

- [파일 전송](#)
- [원격 디렉터리의 콘텐츠 나열](#)
- [원격 서버에서 파일 또는 디렉터리 이동, 이름 바꾸기 또는 삭제](#)

파일 전송

주제

- [SFTP 커넥터를 사용하여 파일을 보내고 검색합니다.](#)

SFTP 커넥터를 사용하여 파일을 보내고 검색합니다.

SFTP 커넥터를 사용하여 파일을 전송하고 검색하려면 [StartFileTransfer](#) API 작업을 사용하고 파일 전송(아웃바운드 전송) 또는 파일 수신(인바운드 전송) 여부에 따라 다음 파라미터를 지정합니다. 각 StartFileTransfer 요청에는 10개의 개별 경로가 포함될 수 있습니다.

Note

기본적으로 SFTP 커넥터는 한 번에 하나의 파일을 처리하여 파일을 순차적으로 전송합니다. 커넥터가 동일한 사용자의 동시 세션을 지원하는 원격 서버와 동시 세션을 생성하고 최대 5개의 파일을 병렬로 처리하도록 하여 전송 성능을 가속화할 수 있습니다. 커넥터에 대한 동시 연결을 활성화하려면 커넥터를 생성하거나 업데이트할 때 최대 동시 연결 설정을 편집할 수 있습니다. 자세한 내용은 [서비스 관리형 송신을 사용하여 SFTP 커넥터 생성](#)을 참조하세요.

- 아웃바운드 전송
 - send-file-paths에는 파트너의 SFTP 서버로 전송할 파일을 위한 1~10개의 소스 파일 경로가 포함됩니다.

- `remote-directory-path`는 고객의 SFTP 서버에서 파일을 전송할 원격 경로입니다.
- 인바운드 전송
 - `retrieve-file-paths`에는 1~10개의 원격 경로가 포함됩니다. 각 경로는 파트너의 SFTP 서버에서 Transfer Family 서버로 파일을 전송할 위치를 지정합니다.
 - `local-directory-path`는 파일이 저장되는 Amazon S3 위치 (버킷 및 선택적 접두사) 입니다.

파일을 보내려면 `send-file-paths` 및 `remote-directory-path` 파라미터를 지정합니다. 최대 10개의 파일을 `send-file-paths` 파라미터에 지정할 수 있습니다. 다음 예 명령은 Amazon S3 스토리지에 있는 `/amzn-s3-demo-source-bucket/file1.txt` 및 `/amzn-s3-demo-source-bucket/file2.txt`라는 파일을 파트너의 SFTP 서버의 `/tmp` 디렉터리로 보냅니다. 이 예 명령을 사용하려면 `amzn-s3-demo-source-bucket`를 실제 버킷으로 대체하세요.

```
aws transfer start-file-transfer --send-file-paths /amzn-s3-demo-source-bucket/
file1.txt /amzn-s3-demo-source-bucket/file2.txt \
  --remote-directory-path /tmp --connector-id c-1111AAAA2222BBBB3 --region us-east-2
```

파일을 검색하려면 `retrieve-file-paths` 및 `local-directory-path` 파라미터를 지정합니다. 다음 예제에서는 `/my/remote/file2.txt` 파트너의 SFTP 서버에서 `/my/remote/file1.txt` 및 파일을 검색하여 Amazon S3 위치 `/amzn-s3-demo-bucket/prefix`에 배치합니다. 이 예 명령을 사용하려면 `user input placeholders`를 실제 정보로 대체하세요.

```
aws transfer start-file-transfer --retrieve-file-paths /my/remote/file1.txt /my/
remote/file2.txt \
  --local-directory-path /amzn-s3-demo-bucket/prefix --connector-id
c-2222BBBB3333CCCC4 --region us-east-2
```

이전 예는 SFTP 서버의 절대 경로를 지정합니다. 상대 경로, 즉 SFTP 사용자의 홈 디렉터리를 기준으로 하는 경로를 사용할 수도 있습니다. 예컨대, SFTP 사용자가 `marymajor`이고 SFTP 서버의 홈 디렉터리가 `/users/marymajor/`인 경우 다음 명령은 `/amzn-s3-demo-source-bucket/file1.txt`을 `/users/marymajor/test-connectors/file1.txt`으로 전송합니다.

```
aws transfer start-file-transfer --send-file-paths /amzn-s3-demo-source-bucket/
file1.txt \
  --remote-directory-path test-connectors --connector-id c-2222BBBB3333CCCC4 --
region us-east-2
```

원격 디렉터리의 콘텐츠 나열

원격 SFTP 서버에서 파일을 검색하기 전에 원격 SFTP 서버의 디렉터리 콘텐츠를 검색할 수 있습니다. 이렇게 하려면 [StartDirectoryListing](#) API 작업을 사용합니다.

다음 예시에서는 커넥터 구성에 지정된 원격 SFTP 서버의 home 폴더 콘텐츠를 나열합니다. 결과는 Amazon S3 위치 /amzn-s3-demo-bucket/connector-files, 라는 파일에 배치됩니다 c-AAAA1111BBBB2222C-6666abcd-11aa-22bb-cc33-0000aaaa3333.json.

```
aws transfer start-directory-listing \
  --connector-id c-AAAA1111BBBB2222C \
  --output-directory-path /amzn-s3-demo-bucket/example/connector-files \
  --remote-directory-path /home
```

이 AWS CLI 명령은 결과가 포함된 파일 이름과 목록 ID를 반환합니다.

```
{
  "ListingId": "6666abcd-11aa-22bb-cc33-0000aaaa3333",
  "OutputFileName": "c-AAAA1111BBBB2222C-6666abcd-11aa-22bb-cc33-0000aaaa3333.json"
}
```

Note

출력 파일의 이름 지정 규칙은 **connector-ID-listing-ID.json**.

JSON 파일에는 다음 정보가 포함되어 있습니다.

- **filePath**: 원격 서버의 SFTP 커넥터에 대한 목록 요청 디렉터리를 기준으로 한 원격 파일의 전체 경로입니다.
- **modifiedTimestamp**: 파일이 마지막으로 수정된 시간, 초 단위, 협정 세계시(UTC) 형식. 이 필드는 선택 사항입니다. 원격 파일 속성에 타임스탬프가 포함되어 있지 않으면 파일 목록에서 생략됩니다.
- **size**: 바이트 단위의 파일 크기입니다. 이 필드는 선택 사항입니다. 원격 파일 속성에 파일 크기가 포함되어 있지 않으면 파일 목록에서 생략됩니다.
- **path**: 원격 서버의 SFTP 커넥터에 대한 목록 요청의 디렉터리를 기준으로 한 원격 디렉터리의 전체 경로입니다.

- **truncated**: 목록 출력에 원격 디렉터리에 포함된 모든 항목이 포함되어 있는지 여부를 나타내는 플래그입니다. **truncated** 출력 값이 true인 경우 선택적 **max-items** 입력 속성에 제공된 값을 늘려 더 많은 항목을 나열할 수 있습니다(최대 허용 목록 크기인 항목 200,000개까지).

다음은 원격 디렉터리에 두 개의 파일과 두 개의 하위 디렉터리(경로 `c-AAAA1111BBBB2222C-6666abcd-11aa-22bb-cc33-0000aaaa3333.json`)가 포함된 출력 파일()의 내용의 예입니다.

```
{
  "files": [
    {
      "filePath": "/home/what.txt",
      "modifiedTimestamp": "2024-01-30T20:34:54Z",
      "size" : 2323
    },
    {
      "filePath": "/home/how.pgp",
      "modifiedTimestamp": "2024-01-30T20:34:54Z",
      "size" : 4691
    }
  ],
  "paths": [
    {
      "path": "/home/magic"
    },
    {
      "path": "/home/aws"
    }
  ],
  "truncated": "false"
}
```

원격 서버에서 파일 또는 디렉터리 이동, 이름 바꾸기 또는 삭제

주제

- [원격 SFTP 서버에서 파일 또는 디렉터리 이동 또는 이름 바꾸기](#)
- [원격 SFTP 서버에서 파일 또는 디렉터리 삭제](#)

원격 SFTP 서버에서 파일 또는 디렉터리 이동 또는 이름 바꾸기

SFTP 커넥터를 사용하여 원격 SFTP 서버에서 파일 및 디렉터리를 이동하거나 이름을 바꿀 수 있습니다. 커넥터를 사용하여 성공적으로 처리하려면 원격 서버가 이러한 작업을 지원해야 합니다.

몇 가지 일반적인 사용 사례는 다음과 같습니다.

- 원격 서버는 매시간 동일한 파일 이름이지만 타임스탬프가 다른 새 파일을 생성하거나 수신합니다. 기본 폴더를 최신 상태로 유지하려면(최신 파일만 포함) 커넥터를 사용하여 이전 파일을 아카이브된 폴더로 이동할 수 있습니다.
- 커넥터를 사용하여 원격 디렉터리의 모든 파일을 나열한 다음 모든 파일을 로컬 스토리지로 전송합니다. 그런 다음 커넥터를 사용하여 파일을 원격 서버의 아카이브된 폴더로 이동할 수 있습니다.

명령은 단일 소스 및 대상 파일 또는 디렉터리를 인수로 취하므로 처리하려는 각 파일 또는 디렉터리에 대해 StartRemoteMove 호출을 사용해야 합니다. 그러나 커넥터가 동일한 사용자의 동시 세션을 지원하는 원격 서버와 동시 세션을 생성하고 최대 5개의 파일을 병렬로 이동/이름 변경하도록 하여 성능을 가속화할 수 있습니다.

다음 예제에서는 원격 SFTP 서버의 파일에서 /source/folder/sourceFile 로 이동/destination/targetFile하고 작업에 대한 고유 식별자를 반환합니다.

```
aws transfer --connector-id c-AAAA1111BBBB2222C start-remote-move \
  --source-path /source/folder/sourceFile --target-path /destination/targetFile
```

Note

이동/이름 변경 작업의 경우 Transfer Family는 표준 SFTP SSH_FXP_RENAME 명령을 사용하여 이동/이름 변경 작업을 수행합니다.

원격 SFTP 서버에서 파일 또는 디렉터리 삭제

SFTP 커넥터를 사용하여 원격 SFTP 서버의 파일 또는 디렉터리를 삭제할 수 있습니다. 커넥터를 사용하여 성공적으로 처리하려면 원격 서버가 이러한 작업을 지원해야 합니다.

Note

원격 디렉터리에 대한 삭제 작업은 빈 디렉터리에만 지원됩니다.

몇 가지 일반적인 사용 사례는 다음과 같습니다.

- 커넥터를 사용하여 원격 SFTP 서버에서 파일을 검색하고 Amazon S3 버킷에 저장한 다음 암호화합니다. 마지막으로 커넥터를 사용하여 원격 서버에서 암호화되지 않은 파일을 삭제할 수 있습니다.
- 커넥터를 사용하여 원격 디렉터리의 모든 파일을 나열한 다음 모든 파일을 로컬 스토리지로 전송합니다. 그런 다음 커넥터를 사용하여 전송한 모든 파일을 삭제할 수 있습니다. 원한다면 원격 디렉터리를 삭제할 수도 있습니다.

명령은 단일 파일 또는 디렉터리를 인수로 취하므로 삭제하려는 각 파일 또는 디렉터리에 대해 `StartRemoteDelete` 호출을 사용해야 합니다. 그러나 커넥터가 동일한 사용자의 동시 세션을 지원하는 원격 서버와 동시 세션을 생성하고 최대 5개의 파일/디렉터리를 병렬로 삭제하도록 하여 성능을 가속화할 수 있습니다.

다음 예제에서는 경로의 원격 SFTP 서버에서 파일을 삭제/`delete/folder/deleteFile`하고 작업에 대한 고유 식별자를 반환합니다.

```
aws transfer start-remote-delete --connector-id c-AAAA1111BBBB2222C \
  --delete-path /delete/folder/deleteFile
```

Note

삭제 작업의 경우 Transfer Family는 표준 `SSH_FXP_REMOVE` 명령을 사용하여 파일을 삭제하고 디렉터리 `SSH_FXP_RMDIR`를 삭제합니다.

SFTP 커넥터 모니터링

다음 방법 중 하나를 사용하여 커넥터 작업의 상태를 모니터링할 수 있습니다. 필요에 맞는 접근 방식을 선택합니다.

커넥터 API를 사용하여 파일 전송 요청 상태 쿼리

파일 전송 작업의 진행 상황을 추적하려면 특정 파일 전송 작업에서 전송 중인 각 개별 파일의 상태에 대한 실시간 업데이트 및 세부 정보를 반환하는 [ListFileTransferResults](#) API 작업을 사용합니다. 커넥터 ID와 전송 ID를 제공하여 파일 전송을 지정합니다. 다음 예제에서는 커넥터 ID `a-1111222233334444` 및 `transfer-ID`에 대한 파일 목록을 반환합니다 `aa1b2c3d4-5678-90ab-cdef-EXAMPLE1111`.


```
aws transfer list-file-transfer-results --connector-id a-1111222233334444 --transfer-id a1b2c3d4-5678-90ab-cdef-EXAMPLE11111
```

Note

파일 전송 결과는 ListFileTransferResults API 작업을 호출한 후 최대 7일까지 사용할 수 있습니다.

SFTP 커넥터를 사용하는 파일 전송 요청에 대한 로그 및 이벤트를 볼 수도 있습니다. Transfer Family에 대한 Amazon EventBridge 이벤트는에 설명되어 있습니다 [SFTP 커넥터 이벤트](#). Transfer Family CloudWatch 로그 항목을 보는 방법은 섹션을 참조하세요 [Transfer Family 로그 스트림 보기](#).

Amazon EventBridge에서 SFTP 커넥터 이벤트 보기

SFTP 커넥터에서 수행하는 각 작업에 대해 Transfer Family는 자동으로 이벤트를 생성하고 Amazon EventBridge 계정의 기본 이벤트 버스로 전송합니다. 이벤트에는 작업 상태를 포함하여 작업에 대한 자세한 메타데이터가 포함됩니다. EventBridge에서 이러한 이벤트를 구독하고, 작업 상태와 같은 특정 이벤트 기준에 필터를 적용하고, 상태에 따라 다운스트림 작업을 자동으로 트리거할 수 있습니다. SFTP 커넥터 작업에서 생성된 이벤트에 대한 자세한 내용은 섹션을 참조하세요 [SFTP 커넥터 이벤트](#).

Amazon CloudWatch에서 SFTP 커넥터 로그 보기

모든 SFTP 커넥터 작업은 CloudWatch에서 세부 로그를 생성합니다. SFTP 커넥터에서 생성된 로그 항목의 예는 섹션을 참조하세요 [SFTP 커넥터의 로그 항목 예](#).

VPC 송신 유형 커넥터 모니터링

VPC 송신 유형 커넥터는 표준 서비스 관리형 커넥터 외에 추가 모니터링 기능과 고려 사항을 제공합니다.

커넥터 상태 모니터링

VPC_LATTICE 커넥터에는 프로비저닝 및 운영 상태를 모니터링하는 데 도움이 되는 추가 정보가 포함되어 있습니다.

- EgressType 필드: VPC_LATTICE 송신 유형 커넥터VPC에 대해 표시됩니다.
- EgressConfig 필드: 리소스 구성 ARN 및 포트 정보를 포함합니다.

describe-connector API를 사용하여 커넥터 상태를 모니터링합니다.

```
aws transfer describe-connector --connector-id c-1234567890abcdef0
```

VPC Lattice 비용 모니터링

VPC 송신 유형 커넥터에는 모니터링해야 하는 추가 VPC Lattice 요금이 발생합니다.

- 리소스 공급자 요금: 리소스 공급자로서 데이터 처리에 대해 0.006/GB의 요금이 청구됩니다(VPC Lattice에서 직접 청구).
- 리소스 소비자 요금: AWS Transfer Family는 0.01 USD/GB 리소스 소비자 비용(첫 1PB)을 흡수합니다.
- NAT 게이트웨이 요금: VPC를 통해 액세스하는 퍼블릭 엔드포인트의 경우 추가 NAT 게이트웨이 및 데이터 전송 요금이 적용될 수 있습니다.
- Transfer Family 요금: 표준 0.40 USD/GB 데이터 처리 요금이 계속 적용됩니다.

AWS 비용 및 결제 콘솔을 통해 VPC Lattice 사용 및 비용을 모니터링하고 VPC Lattice 서비스를 기준으로 필터링합니다.

VPC 커넥터에 대한 네트워크 모니터링

VPC 송신 유형 커넥터의 네트워크 활동 및 성능을 모니터링합니다.

- VPC 흐름 로그: VPC 흐름 로그를 활성화하여 Resource Gateway와 SFTP 서버 간의 네트워크 트래픽 패턴 모니터링
- VPC Lattice 액세스 로그: VPC Lattice는 소스/대상 IP 주소, 연결 타이밍 및 데이터 전송 볼륨을 보여주는 액세스 로그를 제공합니다.
- 보안 그룹 모니터링: 보안 그룹 규칙 및 트래픽 패턴을 모니터링하여 적절한 네트워크 액세스 제어를 보장합니다.
- DNS 확인 모니터링: 서비스 네트워크 엔드포인트의 DNS 확인 시간 및 장애 모니터링

VPC Lattice 액세스 로그 항목의 예:

```
{
  "eventTimestamp": "2025-01-16T20:59:08.531Z",
  "serviceNetworkArn": "arn:aws:vpc-lattice:us-east-1:123456789012:servicenetwork/sn-1234567890abcdef0",
}
```

```

"sourceVpcArn": "arn:aws:ec2:us-east-1:123456789012:vpc/vpc-12345678",
"resourceConfigurationArn": "arn:aws:vpc-lattice:us-east-1:123456789012:resourceconfiguration/rcfg-12345678",
"protocol": "tcp",
"sourceIpPort": "10.0.1.100:33760",
"destinationIpPort": "10.0.2.200:22",
"gatewayIpPort": "10.0.1.150:1769",
"resourceIpPort": "10.0.2.200:22"
}

```

모니터링을 통한 문제 해결

모니터링 데이터를 사용하여 일반적인 VPC 커넥터 문제를 해결합니다.

- 보류 중 상태: DNS 확인 진행 상황을 모니터링하고 전송을 시도하기 전에 ACTIVE 상태를 기다립니다.
- 연결 제한 시간: 포트 22에서 차단된 트래픽에 대한 VPC 흐름 로그 및 보안 그룹 규칙 확인
- 전송 실패: CloudWatch 로그에서 자세한 오류 메시지를 검토하고 VPC Lattice 액세스 로그에서 네트워크 수준 문제를 검토합니다.
- 성능 문제: VPC Lattice 액세스 로그에서 연결 타이밍 및 처리량 지표 모니터링

SFTP 커넥터 관리

이 주제에서는 SFTP 커넥터를 보고 업데이트하는 방법을 설명합니다.

Note

각 커넥터에는 커넥터 수명 동안 변경되지 않은 상태로 유지되는 고정 IP 주소가 자동으로 할당됩니다. 이렇게 하면 알려진 IP 주소의 인바운드 연결만 허용하는 원격 SFTP 서버에 연결할 수 있습니다. 커넥터에는에서 동일한 프로토콜(SFTP 또는 AS2)을 사용하여 모든 커넥터가 공유하는 고정 IP 주소 세트가 할당됩니다 AWS 계정. 이러한 주소는 계정 전용이 아닙니다. 예측 가능한 전용 IP 주소가 필요한 경우 VPC 기반 송신으로 커넥터를 구성합니다. 자세한 내용은 [the section called “VPC 기반 송신을 사용하여 SFTP 커넥터 생성”](#) 단원을 참조하십시오.

SFTP 커넥터 업데이트

커넥터의 기존 파라미터 값을 변경하려면 `update-connector` 명령을 실행하면 됩니다. 다음 명령은 지역의 커넥터 `connector-id`의 암호를 `region-id`에서 `secret-ARN`로 업데이트합니다. 이 예 명령을 사용하려면 `user input placeholders`를 실제 정보로 대체하세요.

```
aws transfer update-connector --sftp-config '{"UserSecretId":"secret-ARN"}' \
  --connector-id connector-id --region region-id
```

VPC 연결 설정 업데이트

서비스 관리형 및 VPC 송신 유형 간 전환 또는 리소스 구성 ARN 변경을 포함하여 기존 커넥터에 대한 VPC 연결 설정을 업데이트할 수 있습니다.

커넥터를 서비스 관리형에서 VPC 송신으로 전환하려면:

```
aws transfer update-connector \
  --connector-id connector-id \
  --egress-type VPC \
  --egress-config ResourceConfigurationArn=resource-configuration-arn
```

VPC_LATTICE 지원 커넥터의 리소스 구성 ARN을 업데이트하려면:

```
aws transfer update-connector \
  --connector-id connector-id \
  --egress-config ResourceConfigurationArn=new-resource-configuration-arn
```

Note

VPC 연결 설정을 업데이트할 때 재구성 프로세스 PENDING 중에 커넥터 상태가 `RECONFIGURING`로 변경됩니다. `describe-connector` 명령을 사용하여 커넥터 상태를 모니터링합니다.

SFTP 커넥터 세부 정보 보기

AWS Transfer Family 콘솔에서 SFTP 커넥터의 세부 정보 및 속성 목록을 찾을 수 있습니다.

커넥터 세부 정보를 보려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.

2. 왼쪽 탐색 창에서 커넥터를 선택합니다.
3. 커넥터 ID 옆에서 식별자를 선택하면 선택한 커넥터의 세부 정보 페이지가 표시됩니다.

커넥터 세부 정보 페이지에서 편집을 선택하여 SFTP 커넥터의 속성을 변경할 수 있습니다.

VPC 커넥터 상태 모니터링

VPC_LATTICE 지원 커넥터에는 프로비저닝 프로세스를 모니터링하는 데 도움이 되는 추가 상태 정보가 포함되어 있습니다.

- 상태: PENDING, ACTIVE 또는 ERRORED를 표시합니다.
- EgressType: VPC 또는 SERVICE_MANAGED를 표시합니다.
- EgressConfig: VPC 커넥터에 대한 리소스 구성 ARN 포함
- 오류: 커넥터가 ERRORED 상태인 경우 자세한 오류 정보를 제공합니다.

VPC 커넥터의 경우 트래픽이 VPC IP 주소를 대신 사용하기 때문에 ServiceManagedEgressIpAddresses 필드는 null입니다.

Note

다음 AWS Command Line Interface (AWS CLI) 명령을 실행하면 다른 형식이긴 하지만 이 정보의 대부분을 얻을 수 있습니다. 이 예 명령을 사용하려면 *user input placeholders*를 실제 정보로 대체하세요.

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

자세한 내용은 API 참조의 [DescribeConnector](#)를 참조하세요.

SFTP 커넥터의 크기 조정 및 할당량

주제

- [Transfer for SFTP 커넥터 할당량](#)
- [SFTP 커넥터 크기 조정](#)

Transfer for SFTP 커넥터 할당량

SFTP 커넥터에는 다음과 같은 할당량이 적용됩니다.

Note

SFTP 커넥터에 대한 추가 서비스 할당량은 [AWS Transfer Family의 엔드포인트 및 할당량](#)에 나열되어 있습니다. Amazon Web Services 일반 참조.

SFTP 커넥터 할당량

이름	기본값	조정 가능
초당 최대 테스트 연결 트랜잭션(TPS)	계정 한 개에 대해 초당 1개의 요청	아니요
보류 중인 파일 전송의 최대 대기열 크기	1000	아니요
최대 파일 크기	150기가바이트(GiB)	아니요
파일당 최대 전송 시간	12시간	아니요
파일당 최대 요청 대기 시간	12시간	아니요
계정당 커넥터의 최대 대역폭 (SFTP 및 AS2 커넥터 모두가 값에 기여)	50MBps	아니요
디렉터리 나열 작업을 위한 최대 항목 수	200,000	아니요

Note

기본적으로 SFTP 커넥터는 한 번에 하나의 파일을 처리하여 파일을 순차적으로 전송합니다. 커넥터가 동일한 사용자의 동시 세션을 지원하는 원격 서버와 동시 세션을 생성하고 최대 5개의 파일을 병렬로 처리하도록 하여 전송 성능을 가속화할 수 있습니다.

커넥터에 대한 동시 연결을 활성화하려면 커넥터를 생성하거나 업데이트할 때 최대 동시 연결 설정을 편집할 수 있습니다. 자세한 내용은 [서비스 관리형 송신을 사용하여 SFTP 커넥터 생성](#)을 참조하세요.

SFTP 커넥터의 자격 증명을 저장하기 위해 각 Secrets Manager 보안 암호와 연결된 할당량이 있습니다. 동일한 보안 암호를 사용하여 여러 유형의 키를 저장하는 경우 여러 용도로 이러한 할당량이 발생할 수 있습니다.

- 단일 보안 암호의 총 길이: 12,000자
- **Password** 문자열의 최대 길이: 1,024자
- **PrivateKey** 문자열의 최대 길이: 8192자
- **Username** 문자열의 최대 길이: 100자

SFTP 커넥터 크기 조정

이 섹션에서는 AWS Transfer Family SFTP 커넥터 워크로드를 확장하는 방법에 대한 고려 사항을 설명합니다. SFTP 커넥터를 사용하여 워크로드를 확장하려는 경우 적용되는 다음 세 가지 할당량을 고려해야 합니다.

- 최대 대기열 크기입니다. 이는 커넥터 대기열에서 요청된 대기 중인 작업의 최대 수를 나타냅니다. 보류 중인 작업은 아직 완료되지 않은 이전에 제출된 전송 요청을 성공 또는 실패로 나타냅니다.

보류 중인 요청의 최대 대기열 깊이는 현재 커넥터당 1,000개로 설정됩니다([AWS Transfer Family 서비스 할당량](#)에 정의됨). 짧은 기간 동안 수천 개의 전송 작업을 요청하면 워크로드가 서비스 제한을 초과할 수 있으며 대기 중인 최대 요청 초과라는 메시지가 `ThrottlingException`에 표시됩니다. 워크로드에 할당량이 적용되는 경우 AWS Support 또는 계정 팀을 통해 Transfer Family 서비스 팀에 문의하여 확장성 요구 사항에 대해 논의하세요.

다음 작업 중 하나 또는 둘 다를 수행할 수도 있습니다.

- 파일 볼륨을 여러 커넥터에 분산합니다.
- 커넥터가 원격 서버와 동시 세션을 생성하여 대기열의 여러 요청을 병렬로 처리하도록 합니다.
- 동시 세션 수입입니다. 기본적으로 SFTP 커넥터는 한 번에 하나의 파일을 전송하여 대기열에서 순차적으로 파일을 전송합니다.

커넥터가 여러 파일을 병렬로 전송하도록 하여 전송 성능을 가속화할 수 있습니다. 동일한 사용자의 동시 세션을 지원하는 원격 서버로 동시 세션을 생성하고 최대 5개의 파일을 병렬로 처리할 수 있습니다.

니다. SFTP 커넥터를 생성할 때 커넥터를 생성하거나 업데이트할 때 최대 동시 연결 설정에 대해 최대 5개의 값을 선택합니다. 자세한 내용은 [서비스 관리형 송신을 사용하여 SFTP 커넥터 생성](#)을 참조하세요.

- **StartFileTransfer** 요청 속도입니다. 각 SFTP 커넥터를 사용하여 초당 최대 100개의 파일 경로를 전송하도록 요청할 수 있습니다. 요청된 파일 경로는 처리를 위해 커넥터의 대기열에 추가됩니다. 개별 StartFileTransfer 명령에 제공된 파일 수에 관계없이 StartFileTransfer 명령을 반복적으로 사용하여 커넥터당 초당 최대 100개의 파일 경로를 요청할 수 있습니다.

SFTP 커넥터를 사용한 참조 아키텍처

이 섹션에서는 SFTP 커넥터를 사용하여 자동 파일 전송 워크플로를 구성하는 데 사용할 수 있는 참조 자료를 나열합니다. Amazon EventBridge의 SFTP 커넥터 이벤트를 사용하여에서 파일 전송 작업과 사전 및 사후 처리 작업 간에 오케스트레이션하여 자체 이벤트 기반 아키텍처를 설계할 수 있습니다 AWS.

블로그 게시물

다음 블로그 게시물은 SFTP 커넥터를 사용하여 원격 SFTP 서버로 전송하기 전에 PGP를 사용한 파일 암호화를 포함하여 [AWS Transfer Family SFTP 커넥터를 사용하여 MFT 워크플로를 빌드하기 위한 참조 아키텍처](#)를 제공합니다. SFTP 커넥터 및 PGP 암호화를 사용하여 안전하고 규정을 준수하는 관리형 파일 전송 설계.

워크숍

- 다음 워크숍에서는 SFTP 커넥터를 구성하고 커넥터를 사용하여 원격 SFTP 서버에서 파일을 보내거나 검색하기 위한 실습을 제공합니다. [Transfer Family - SFTP 워크숍](#).
- 다음 워크숍은 실습을 통해 외부 SFTP 서버와 Amazon S3 간의 파일 전송 및 해당 파일의 일반적인 사전 및 사후 처리와 관련된 완전 자동화된 이벤트 기반 워크플로인 [이벤트 기반 MFT 워크숍](#)을 구축합니다.

이 동영상에서는이 워크숍을 안내합니다.

Solutions

AWS Transfer Family 는 다음 솔루션을 제공합니다.

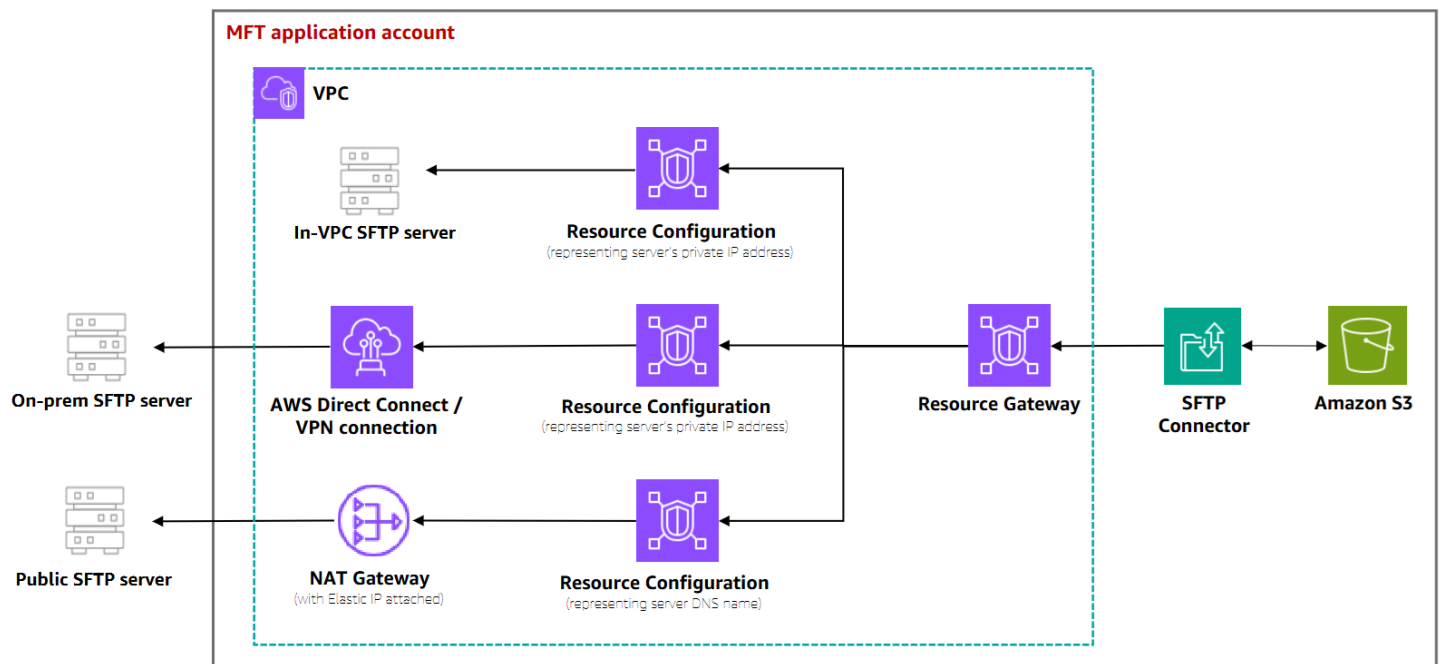
- 파일 전송 동기화 솔루션은 SFTP 커넥터를 사용하여 전체 폴더 구조를 포함한 원격 SFTP 디렉터리를 로컬 Amazon S3 버킷과 동기화하는 프로세스를 자동화하는 참조 아키텍처를 제공합니다. 원격 디렉터리를 나열하고, 변경 사항을 감지하고, 새 파일 또는 수정된 파일을 전송하는 프로세스를 오케스트레이션합니다.
- Serverlessland -를 사용한 원격 SFTP 서버와 S3; 간의 선택적 파일 전송 AWS Transfer Family은 원격 SFTP 위치에 저장된 파일을 나열하고 선택적 파일을 Amazon S3로 전송하기 위한 샘플 패턴을 제공합니다.

VPC 참조 아키텍처

다음 참조 아키텍처는 VPC_LATTICE 지원 SFTP 커넥터를 배포하기 위한 일반적인 패턴을 보여줍니다. 이 예제는 전체 AWS 아키텍처에서 VPC Lattice 리소스를 생성해야 하는 위치를 이해하는 데 도움이 됩니다.

공유 송신 인프라가 있는 단일 계정

이 아키텍처에서 송신 인프라(NAT 게이트웨이, VPN 터널 또는 Direct Connect)는 SFTP 커넥터와 동일한 계정 내의 VPC에 구성됩니다. 모든 커넥터는 동일한 리소스 게이트웨이와 NAT 게이트웨이를 공유할 수 있습니다.



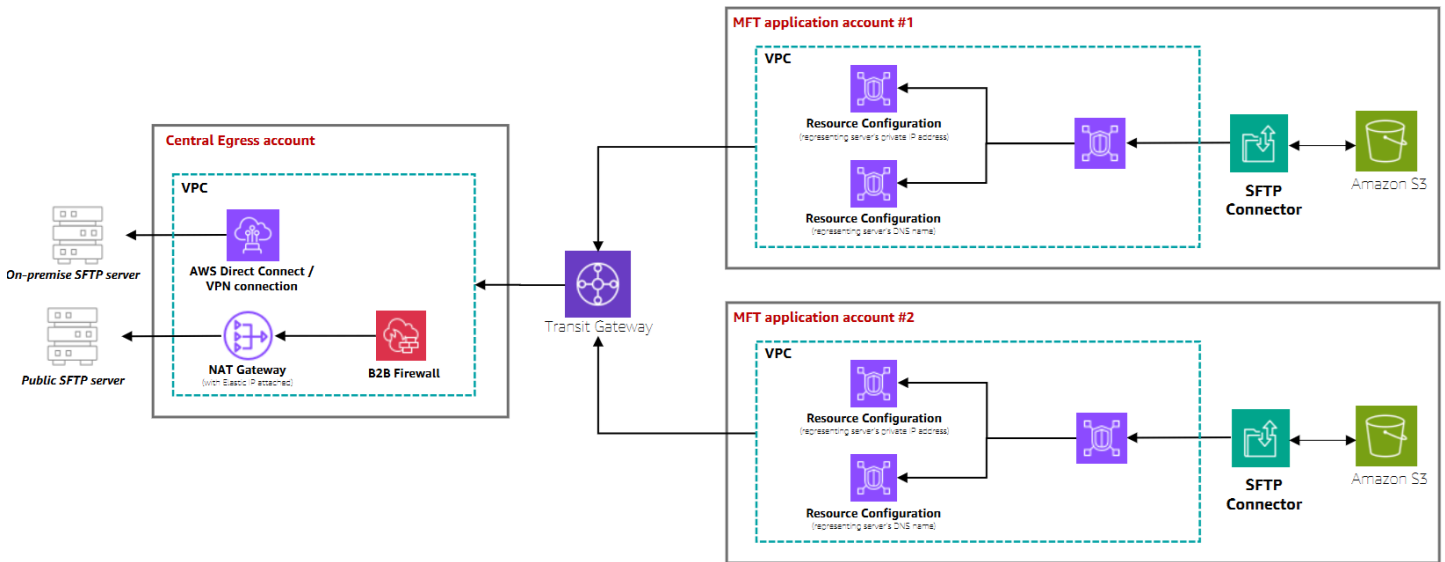
이 패턴은 다음과 같은 경우에 적합합니다.

- 모든 SFTP 커넥터는 단일 내에서 관리됩니다. AWS 계정

- 송신 인프라는 SFTP 커넥터와 동일한 계정 내의 VPC에 설정됩니다.

중앙 집중식 송신 인프라를 사용한 교차 계정

이 아키텍처에서 송신 인프라(NAT 게이트웨이, VPN 터널, Direct Connect 또는 B2B 방화벽)는 네트워킹 팀에서 관리하는 중앙 송신 계정에 구성됩니다. SFTP 커넥터는 MFT 관리자 팀이 관리하는 MFT 애플리케이션 계정에서 생성됩니다. 교차 계정 네트워킹은 기존 네트워킹 규칙을 준수하기 위해 Transit Gateway를 사용하여 설정됩니다.



이 패턴은 다음과 같은 경우에 적합합니다.

- 네트워크 인프라는 전용 계정의 별도 팀에서 관리합니다.
- SFTP 커넥터가 생성된 계정과 송신 인프라가 설정된 계정 사이에 기존 경로(예: AWS Transit Gateway)가 있습니다. SFTP 커넥터는 두 계정을 연결하는 기존 경로를 활용할 수 있습니다.
- 중앙 집중식 보안 제어 및 B2B 방화벽 필요
- 네트워킹 팀과 애플리케이션 팀 간의 업무 분리를 유지해야 합니다.

AWS Transfer Family AS2용

적용성 보고서 2(AS2)는 강력한 메시지 보호 및 검증 메커니즘을 포함하는 RFC 정의 파일 전송 사양입니다. 전송 중 AS2 페이로드를 보호하려면 암호화 및 디지털 서명과 함께 암호화 메시지 구문(CMS)을 사용하여 데이터 보호 및 피어 인증을 제공합니다. 서명된 메시지 처리 알림(MDN) 응답 페이로드는 메시지가 수신되고 성공적으로 복호화되었다는 확인(부인 방지)을 제공합니다.

AS2 프로토콜은 프로토콜에 데이터 보호 및 보안 기능을 내장하는 데 의존하는 규정 준수 요구 사항이 있는 워크플로에 매우 중요합니다. AWS Transfer Family AS2 엔드포인트는 [Drummond 인증](#)을 받았으며 소매, 생명과학, 제조, 금융 서비스 및 유틸리티와 같은 산업의 고객이 비즈니스 파트너와 안전하게 거래할 수 있도록 지원합니다.

Transfer Family와 함께 AS2를 사용하는 경우 트랜잭션된 데이터는 다음을 위해에서 AWS 기본적으로 액세스할 수 있습니다.

- 처리, 분석 및 기계 학습
- 엔터프라이즈 리소스 계획(ERP) 시스템과의 통합
- 고객 관계 관리(CRM) 시스템과의 통합

AS2-enabled 서버가 있는 파트너와 파일을 교환하려면 다음을 수행해야 합니다.

- 암호화를 위한 퍼블릭-프라이빗 키 페어 생성
- 서명을 위한 퍼블릭-프라이빗 키 페어 생성
- 파트너와 퍼블릭 키 교환

Important

HTTPS AS2 서버 엔드포인트는 현재 지원되지 않습니다. TLS 종료에 대한 책임은 사용자에게 있습니다.

Transfer Family는 AS2가 활성화된 Transfer Family 엔드포인트와 Transfer Family AS2 커넥터를 구성할 수 있는 워크숍을 제공합니다. 이 워크숍의 세부 정보는 [여기](#)에서 확인할 수 있습니다.

Transfer Family에서 AS2를 구성하는 step-by-step 지침은 다음을 참조하세요.

1. [AS2 인증서 가져오기](#)
2. [AS2 프로필 생성](#)
3. [AS2 서버 생성](#)
4. [AS2 계약 생성](#)
5. [AS2 커넥터 구성](#)

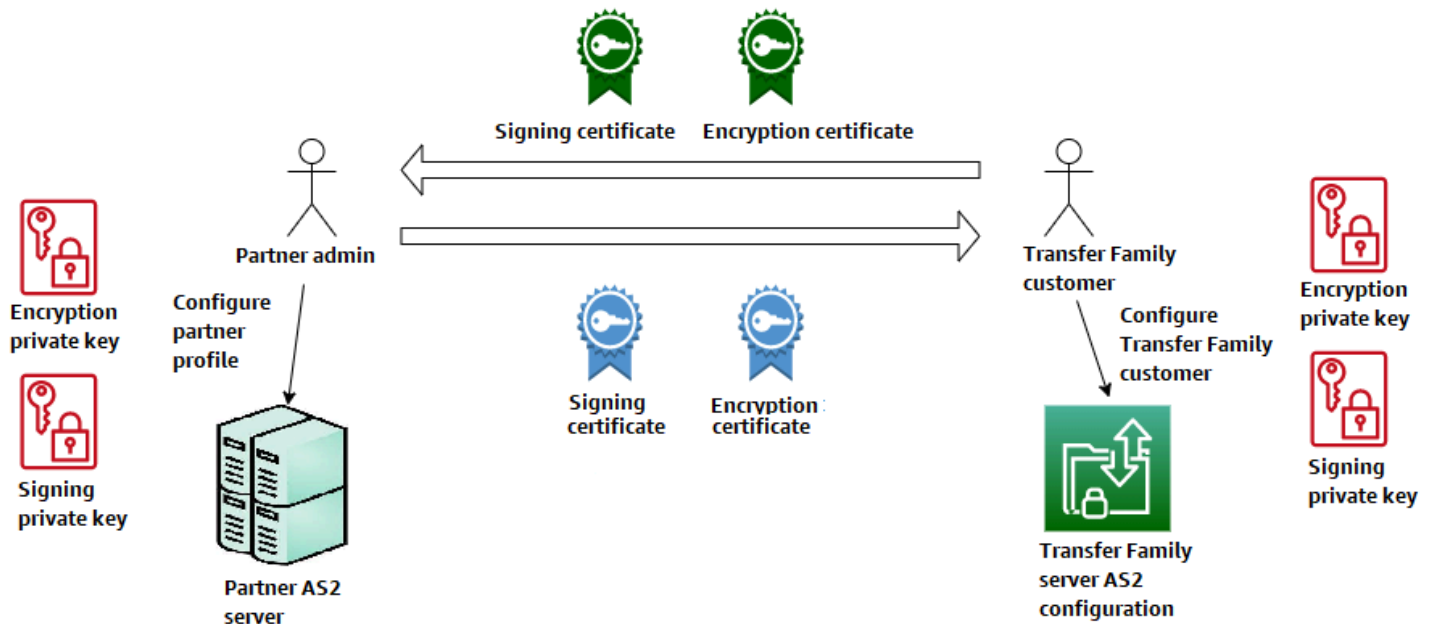
전체 예제는 섹션을 참조하세요 [AS2 구성 설정](#).

Note

AS2 Terraform 템플릿에 대한 지원을 표시하려면 [Transfer Family Terraform 템플릿 기능 요청](#)에 씬업 반응(#)을 추가합니다. 사용 사례를 설명하는 설명을 추가할 수도 있습니다.

AS2 사용 사례

AS2-enabled 서버가 있는 파트너와 파일을 교환하려는 AWS Transfer Family 고객인 경우 설정의 가장 복잡한 부분은 암호화를 위한 퍼블릭-프라이빗 키 페어 하나와 파트너와 퍼블릭 키를 서명하고 교환하기 위한 퍼블릭-프라이빗 키 페어 하나를 생성하는 것입니다.



AS2와 AWS Transfer Family 함께를 사용하려면 다음과 같은 변형을 고려하세요.

Note

거래 파트너는 해당 파트너 프로필과 연결된 파트너입니다.
다음 표의 MDN에 대한 모든 언급은 서명된 MDN을 가정합니다.

AS2 사용 사례**인바운드 전용 사용 사례**

- 거래 파트너에서 Transfer Family 서버로 암호화된 AS2 메시지를 전송합니다.

이 경우 다음과 같이 합니다:

1. 거래 파트너와 자신을 위한 프로필을 생성합니다.
2. AS2 프로토콜을 사용하는 Transfer Family 서버를 생성합니다.
3. 계약서를 작성하여 서버에 추가합니다.
4. 프라이빗 키가 있는 인증서를 가져와 프로필에 추가한 다음 암호화를 위해 파트너 프로필로 퍼블릭 키를 가져옵니다.
5. 이러한 항목을 확보한 후에는 인증서용 퍼블릭 키를 거래 파트너에게 보내세요.

이제 파트너가 암호화된 메시지를 보내고 사용자는 이를 복호화하여 Amazon S3 버킷에 저장할 수 있습니다.

- 거래 파트너에서 Transfer Family 서버로 암호화된 AS2 메시지를 전송하고 서명을 추가합니다.

이 시나리오에서는 여전히 인바운드 전송만 하고 있지만, 이제는 파트너가 보내는 메시지에 서명하도록 하는 것이 좋습니다. 이 경우 거래 파트너의 서명 퍼블릭 키(파트너의 프로필에 추가된 서명 인증서)를 가져옵니다.

- 거래 파트너에서 Transfer Family 서버로 암호화된 AS2 메시지를 전송하고 서명 및 전송 MDN 응답을 추가합니다.

이 시나리오에서는 여전히 인바운드 전송만 하고 있지만, 이제 거래 파트너는 서명된 페이로드를 받는 것 외에도 서명된 MDN 응답을 받기를 원합니다.

1. 공개 및 비공개 서명 키(프로필의 서명 인증서로 사용)를 가져오세요.
2. 거래 파트너에게 퍼블릭 서명 키를 보냅니다.

아웃바운드 전용 사용 사례

- Transfer Family 서버에서 거래 파트너로 암호화된 AS2 메시지를 전송합니다.

이 경우는 AS2 서버에 계약을 추가하는 대신 커넥터를 만든다는 점을 제외하면 인바운드 전용 전송 사용 사례와 비슷합니다. 이 경우 거래 파트너의 퍼블릭 키를 프로필로 가져옵니다.

- Transfer Family 서버에서 거래 파트너로 암호화된 AS2 메시지를 전송하고 서명을 추가합니다.

아직 아웃바운드 전송만 하고 있지만 이제 거래 파트너가 전송한 메시지에 서명하기를 원합니다.

1. 서명 프라이빗 키(프로필에 추가된 서명 인증서)를 가져오세요.
2. 거래 파트너에게 퍼블릭 키를 보냅니다.

- Transfer Family 서버에서 거래 파트너로 암호화된 AS2 메시지를 전송하고 서명을 추가하고 MDN 응답을 전송합니다.

아직 아웃바운드 전송만 수행하고 있지만 이제는 서명된 페이로드를 보내는 것 외에도 거래 파트너로부터 서명된 MDN 응답을 받으려고 합니다.

1. 거래 파트너가 퍼블릭 서명 키를 보냅니다.
2. 거래 파트너의 퍼블릭 키(파트너 프로필에 추가된 서명 인증서)를 가져옵니다.

인바운드 및 아웃바운드 사용 사례

- Transfer Family 서버와 거래 파트너 간에 암호화된 AS2 메시지를 양방향으로 전송합니다.

이 경우 다음과 같이 합니다:

1. 거래 파트너와 자신을 위한 프로필을 생성합니다.
2. AS2 프로토콜을 사용하는 Transfer Family 서버를 생성합니다.
3. 계약서를 작성하여 서버에 추가합니다.
4. 커넥터를 생성합니다.
5. 프라이빗 키가 있는 인증서를 가져와 프로필에 추가한 다음 암호화를 위해 파트너 프로필로 퍼블릭 키를 가져옵니다.
6. 거래 파트너로부터 퍼블릭 키를 받아 암호화를 위해 프로필에 추가합니다.
7. 이러한 항목을 확보한 후에는 인증서용 퍼블릭 키를 거래 파트너에게 보내세요.

이제 거래 파트너와 암호화된 메시지를 교환할 수 있으며 둘 다 암호를 복호화할 수 있습니다. 사용자는 Amazon S3 버킷에 수신한 메시지를 저장할 수 있으며, 파트너는 사용자가 전송한 메시지를 복호화하여 저장할 수 있습니다.

- Transfer Family 서버와 거래 파트너 간에 양방향으로 암호화된 AS2 메시지를 전송하고 서명을 추가합니다.

이제 당신과 당신의 파트너는 서명된 메시지를 원합니다.

1. 서명 프라이빗 키(프로필에 추가된 서명 인증서)를 가져오세요.
 2. 거래 파트너에게 퍼블릭 키를 보냅니다.
 3. 거래 파트너의 서명 퍼블릭 키를 가져와 프로필에 추가합니다.
- Transfer Family 서버와 거래 파트너 간에 양방향으로 암호화된 AS2 메시지를 전송하고 서명을 추가하고 MDN 응답을 전송합니다.

이제 서명된 페이로드를 교환하고 사용자와 거래 파트너 모두 MDN 응답을 원합니다.

1. 거래 파트너가 퍼블릭 서명 키를 보냅니다.
2. 거래 파트너의 퍼블릭 키를 가져옵니다(파트너 프로필에 대한 서명 인증서로).
3. 거래 파트너에게 퍼블릭 키를 보냅니다.

AS2 CloudFormation 템플릿

이 주제에서는 AS2 서버 및 구성을 빠르게 배포하는 데 사용할 수 있는 AWS CloudFormation 템플릿에 대한 정보를 제공합니다 AWS Transfer Family. 이러한 템플릿은 설정 프로세스를 자동화하고 AS2 파일 전송에 대한 모범 사례를 구현하는 데 도움이 됩니다.

- 기본 AS2 템플릿은에 설명되어 있습니다. [템플릿을 사용하여 데모 Transfer Family AS2 스택을 생성하세요.](#)
- HTTP 헤더를 사용자 지정하기 위한 AS2 템플릿은에 설명되어 있습니다 [AS2 메시지에 대한 HTTP 헤더 사용자 지정.](#)

AS2 템플릿 사용자 지정

제공된 템플릿을 특정 요구 사항에 맞게 사용자 지정할 수 있습니다.

1. S3 URL에서 템플릿을 다운로드합니다.
2. YAML 코드를 수정하여 다음과 같은 구성을 조정합니다.
 - 보안 설정 및 인증서 구성
 - 네트워크 아키텍처 및 VPC 설정
 - 스토리지 옵션 및 파일 처리
 - 모니터링 및 알림 기본 설정
3. 수정된 템플릿을 자체 S3 버킷에 업로드합니다.
4. CloudFormation 콘솔 또는를 사용하여 사용자 지정 템플릿을 배포합니다 AWS CLI.

Important

템플릿을 사용자 지정할 때 리소스 간의 종속성을 유지하고 보안 모범 사례를 따르십시오.

AS2 배포 테스트

템플릿을 사용하여 AS2 서버를 배포한 후 구성을 테스트할 수 있습니다.

1. CloudFormation 스택 출력에서 샘플 명령 및 엔드포인트 정보를 확인합니다.
2. AWS CLI 를 사용하여 테스트 파일을 전송합니다.


```
aws s3api put-object --bucket your-bucket-name --key test.txt --body test.txt
aws transfer start-file-transfer --connector-id your-connector-id --send-file-
paths /your-bucket-name/test.txt
```

3. 대상 S3 버킷에서 파일 전송을 확인합니다.
4. CloudWatch 로그에서 성공적인 처리 및 MDN 응답을 확인합니다.

보다 포괄적인 테스트를 위해 타사 AS2 클라이언트를 사용하여 Transfer Family AS2 서버로 파일을 전송하는 것이 좋습니다.

AS2 템플릿 배포 모범 사례

AS2 CloudFormation 템플릿을 사용할 때는 다음 모범 사례를 따르세요.

보안

강력한 인증서를 사용하고 정기적으로 교체합니다.

최소 권한 IAM 정책을 구현합니다.

보안 그룹을 사용하여 네트워크 액세스를 제한합니다.

신뢰성

여러 가용 영역에 배포합니다.

전송 실패에 대한 모니터링 및 알림을 구현합니다.

전송 실패에 대한 자동 재시도를 설정합니다.

성능

전송 볼륨에 적합한 인스턴스 유형을 선택합니다.

효율적인 파일 관리를 위해 S3 수명 주기 정책을 구현합니다.

네트워크 구성을 모니터링하고 최적화합니다.

비용 최적화

가변 워크로드에 Auto Scaling을 사용합니다.

이전 파일에 대한 S3 스토리지 클래스를 구현합니다.

실제 사용량에 따라 리소스를 모니터링하고 조정합니다.

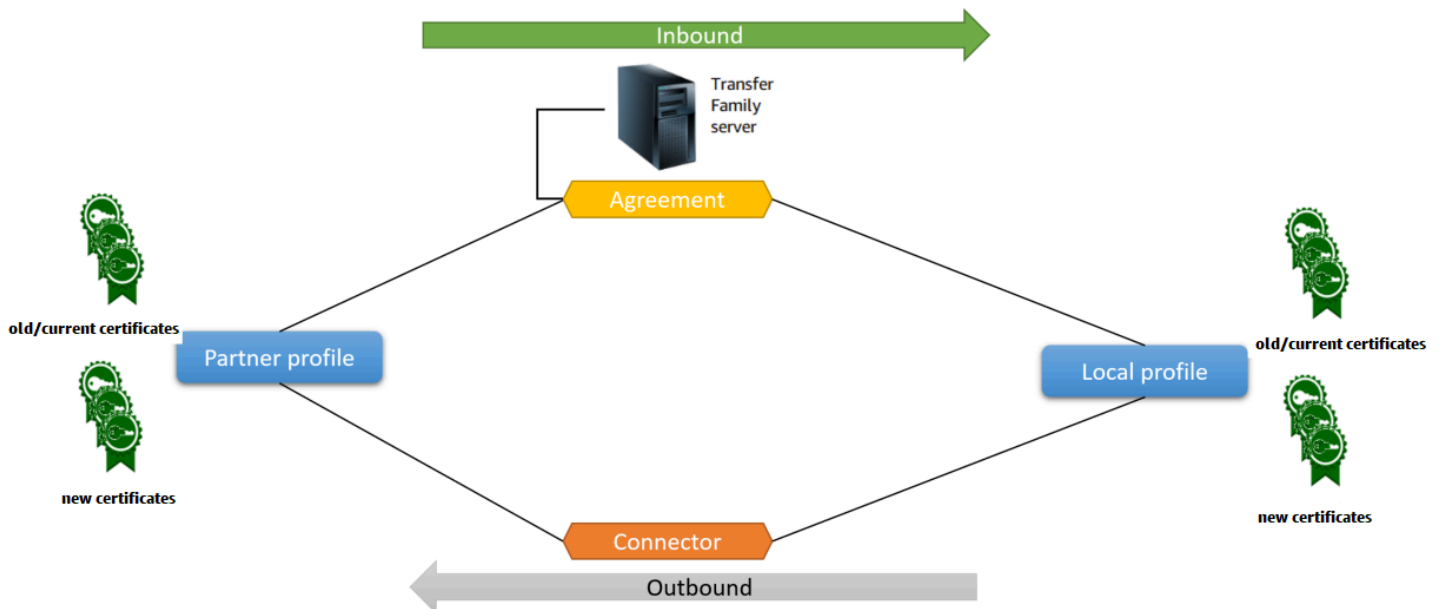
AS2 구성

AS2 지원 서버를 생성하려면 다음 구성 요소도 지정해야 합니다:

- 계약 - 양자간 거래 파트너 계약 또는 파트너십, 메시지(파일)를 교환하는 두 당사자들 사이의 관계를 정의함. 계약을 정의하기 위해 Transfer Family는 서버, 로컬 프로필, 파트너 프로필, 인증서 정보를 결합합니다. Transfer Family AS2-인바운드 프로세스는 계약을 사용합니다.
- 인증서 - 퍼블릭 키(X.509) 인증서는 AS2 통신에서 메시지 암호화 및 확인을 위해 사용됩니다. 인증서는 커넥터 엔드포인트에도 사용됩니다.
- 로컬 프로필 및 파트너 프로필 - 로컬 프로필은 로컬(AS2 지원 Transfer Family 서버) 조직 또는 “당사자”를 정의합니다. 마찬가지로 파트너 프로필은 Transfer Family 외부의 원격 파트너 조직을 정의합니다.

모든 AS2 지원 서버에 필수는 아니지만 아웃바운드 전송의 경우 커넥터가 필요합니다. 커넥터는 아웃바운드 연결에 대한 파라미터를 캡처합니다. 커넥터는 고객의 외부 비 AWS 서버로 파일을 전송하는데 필요합니다.

다음 다이어그램은 인바운드 프로세스와 아웃바운드 프로세스에 관련된 AS2 객체 간의 관계를 나타낸 것입니다.



종합 예 AS2 구성 에 대한 자세한 설명은 [AS2 구성 설정](#) 섹션을 참조하세요.

주제

- [AS2 구성](#)
- [AS2 할당량 및 제한사항](#)
- [AS2 특성 및 기능](#)

AS2 구성

이 항목에서는 허용되는 암호 및 다이제스트를 포함하여 적용성 보고서 2(AS2) 프로토콜을 사용하는 전송에 대해 지원되는 구성, 기능 및 기능에 대해 설명합니다.

서명, 암호화, 압축, MDN

인바운드 전송과 아웃바운드 전송 모두에서 다음 항목은 필수 또는 옵션입니다.

- 암호화 - 필수 (현재 지원되는 유일한 전송 방법인 HTTP 전송의 경우). 암호화되지 않은 메시지는 ALB (Application Load Balancer) 와 같은 TLS 종료 프록시에 의해 전달되고 X-Forwarded-Proto: https헤더가 있는 경우에만 수락됩니다.
- 서명 — 옵션
- 압축 - 옵션(현재 지원되는 유일한 압축 알고리즘은 ZLIB)
- 메시지 처리 알림 (MDN) — 옵션

암호(Ciphers)

인바운드 및 아웃바운드 전송 모두에 대해 다음 암호가 지원됩니다.

- AES128_CBC
- AES192_CBC
- AES256_CBC
- 3DES(이전 버전과의 호환성에만 해당)

Digests

다음 데이터 유형이 지원됩니다.

- 인바운드 서명 및 MDN — SHA1, SHA256, SHA384, SHA512

- 아웃바운드 서명 및 MDN – SHA1, SHA256, SHA384, SHA512

MDN

MDN 응답의 경우 다음과 같은 특정 타입이 지원됩니다.

- 인바운드 전송 — 동기 및 비동기
- 아웃바운드 전송 - 동기식 및 비동기식
- Simple Mail Transfer Protocol (SMTP) (이메일 MDN) — 지원되지 않음

Transports

- 인바운드 전송 - HTTP는 현재 지원되는 유일한 전송이므로 HTTP를 명시적으로 지정해야 합니다.

Note

인바운드 전송에 HTTPS를 사용해야 하는 경우 Application Load Balancer 또는 Network Load Balancer에서 TLS를 종료할 수 있습니다. 이에 대한 설명은 [HTTPS를 통해 AS2 메시지를 받습니다](#)에 나와 있습니다.

- 아웃바운드 전송 - HTTP URL을 제공하는 경우 암호화 알고리즘도 지정해야 합니다. HTTPS URL을 제공하는 경우 암호화 알고리즘에 NONE을 지정할 수 있습니다.

AS2 할당량 및 제한사항

이 섹션에서는 AS2의 할당량 및 제한 사항에 대해 설명합니다.

주제

- [AS2 할당량](#)
- [비밀 처리를 위한 할당량](#)
- [알려진 제한 사항](#)

AS2 할당량

AS2 파일 전송에는 다음과 같은 할당량이 적용됩니다. 조정 가능한 할당량 증가를 요청하려면 AWS 일반 참조의 [AWS 서비스 할당량](#)을 참조하세요.

AS2 할당량

이름	기본값	조정 가능
아웃바운드 요청당 최대 파일 수	10	아니요
초당 최대 아웃바운드 요청 수	100	아니요
초당 최대 인바운드 요청 수	100	아니요
계정당 최대 아웃바운드 대역 폭(아웃바운드 SFTP 및 AS2 요청 모두에 기여)	초당 50MB	아니요

비밀 처리를 위한 할당량

AWS Transfer Family 는 기본 인증을 사용하는 AS2 고객을 AWS Secrets Manager 대신하여를 호출합니다. 또한 Secrets Manager를 호출합니다 AWS KMS.

 Note

이러한 할당량은 Transfer Family의 보안 암호 사용에 국한되지 않습니다. 이러한 할당량은의 모든 서비스 간에 공유됩니다 AWS 계정.

[Secrets GetSecretValueManager의 경우 적용되는 할당량은 할당량에 설명된 대로 DescribeSecret 및 GetSecretValue API 요청의 합산 비율입니다.AWS Secrets Manager](#)

Secrets Manager **GetSecretValue**

이름	값	설명
DescribeSecret 및 GetSecret Value API 요청의 합산 비율	각각 지원되는 리전: 초당 10,000개	DescribeSecret 및 GetSecretValue API 작업을 합친 초당 최대 트랜잭션 수입니다.

AWS KMS의 경우 다음 할당량이에 적용됩니다Decrypt. 자세한 내용은 [각 AWS KMS API 작업에 대한 할당량 요청을 참조하세요](#).

AWS KMS Decrypt

할당량 이름	기본값(초당 요청 수)
암호화 작업(대칭) 요청 빈도	이러한 공유 할당량은 AWS 리전 및 요청에 사용되는 AWS KMS 키 유형에 따라 달라집니다. 각 할당량은 별도로 계산됩니다. • 5500(공유) • 다음 리전에서는 10000(공유): • 미국 동부(오하이오) us-east-2 • 아시아 태평양(싱가포르) ap-southeast-1 • 아시아 태평양(시드니) ap-southeast-2 • 아시아 태평양(도쿄) ap-northeast-1 • EU(프랑크푸르트) eu-central-1 • EU(런던) eu-west-2 • 다음 리전에서는 50000(공유): • 미국 동부(버지니아 북부), us-east-1 • 미국 서부(오리건), us-west-2 • 유럽(아일랜드) eu-west-1
사용자 지정 키 스토어 요청 할당량  Note 이 할당량은 외부 키 저장소를 사용하는 경우에만 적용됩니다.	사용자 지정 키 저장소 요청 할당량은 각 사용자 지정 키 저장소에 대해 별도로 계산됩니다. • 각 AWS CloudHSM 키 스토어에 대해 1,800(공유) • 각 외부 키 저장소에 대해 1800회(공유)

알려진 제한 사항

- 서버측 TCP keep-alive는 지원되지 않습니다. 클라이언트가 keep-alive 패킷을 전송하지 않는 한 350초 동안 사용하지 않으면 연결 제한 시간이 초과됩니다.

- 서비스에서 유효한 계약을 수락하고 Amazon CloudWatch Logs에 표시하려면 메시지에 유효한 AS2 헤더가 포함되어야 합니다.
- AS2 AWS Transfer Family 용에서 메시지를 수신하는 서버는 [RFC 6211](#)에 정의된 대로 메시지 서명을 검증하기 위한 암호화 메시지 구문(CMS) 알고리즘 보호 속성을 지원해야 합니다. 이 속성은 일부 구형 IBM Sterling 제품에서는 지원되지 않습니다.
- 메시지 ID가 중복되면 처리됨/경고: 문서 중복 메시지가 표시됩니다.
- AS2 인증서의 키 길이는 최소 2,048비트, 최대 4,096비트여야 합니다.
- AS2 메시지 또는 비동기 mDN을 거래 파트너의 HTTPS 엔드포인트로 보내는 경우 메시지 또는 mDN은 공개적으로 신뢰할 수 있는 인증 기관 (CA) 에서 서명한 유효한 SSL 인증서를 사용해야 합니다. 자체 서명된 인증서는 현재 아웃바운드 전송에만 지원됩니다.
- 엔드포인트는 TLS 버전 1.2 프로토콜과 보안 정책에서 허용하는 암호화 알고리즘을 ([AWS Transfer Family 서버에 대한 보안 정책](#)에서 설명한 것과 같이) 지원해야 합니다.
- AS2 버전 1.2의 다중 첨부 파일 및 CEM (인증서 교환 메시징) 은 현재 지원되지 않습니다.
- 기본 인증은 현재 아웃바운드 메시지에만 지원됩니다.
- AS2 프로토콜을 사용하는 Transfer Family 서버에 파일 처리 워크플로를 연결할 수 있습니다. 하지만 AS2 메시지는 서버에 연결된 워크플로를 실행하지 않습니다.

AS2 특성 및 기능

다음 표에는 AS2를 사용하는 Transfer Family 리소스에 사용할 수 있는 기능이 나열되어 있습니다.

AS2 특성

Transfer Family는 AS2에 다음과 같은 특성을 제공합니다.

Feature	Supported by AWS Transfer Family
Drummond 인증	Yes
AWS CloudFormation 지원	Yes
Amazon CloudWatch 지표	Yes
SHA-2 암호화 알고리즘	Yes
Support for Amazon S3	Yes

Feature	Supported by AWS Transfer Family
Support for Amazon EFS	No
Scheduled Messages	Yes ¹
AWS Transfer Family Managed Workflows	No
Certificate Exchange Messaging (CEM)	No
Mutual TLS (mTLS)	No
Support for self-signed certificates	Yes

1. [Amazon EventBridge를 사용하여 AWS Lambda 함수를 예약](#)하여 아웃바운드 예약 메시지 사용 가능

AS2 송신 및 수신 기능

다음 표에는 AWS Transfer Family AS2 송수신 기능 목록이 나와 있습니다.

Capability	Inbound: Receiving with server	Outbound: Sending with connector
TLS 암호화 전송 (HTTPS)	예 ¹	Yes
Non-TLS Transport (HTTP)	Yes	예 ²
Synchronous MDN	Yes	Yes
Message Compression	Yes	Yes
Asynchronous MDN	Yes	Yes
Static IP Address	Yes	Yes
Bring Your Own IP Address	Yes	No
Multiple File Attachments	No	No
Basic Authentication	No	Yes

Capability	Inbound: Receiving with server	Outbound: Sending with connector
AS2 Restart	Not applicable	No
AS2 Reliability	No	No
Custom Subject per Message	Not applicable	No

1. Network Load Balancer(NLB) 또는 Application Load Balancer(ALB)에서 사용할 수 있는 인바운드 TLS 암호화 전송

2. 아웃바운드 비 TLS 전송은 암호화가 활성화된 경우에만 사용할 수 있습니다.

AS2 인증서 관리

이 주제에서는 AS2 인증서를 가져오고 관리하는 방법을 설명합니다. 인증서 가져오기는 Transfer Family에 대한 AS2 프로세스의 첫 번째 단계입니다.

1. 인증서 가져오기
2. [AS2 프로파일 생성](#)
3. [AS2 서버 생성](#)
4. [AS2 계약 생성](#)
5. [AS2 커넥터 구성](#)

AS2 인증서 가져오기

Transfer Family AS2 프로세스는 전송된 정보의 암호화와 서명 모두에 인증서 키를 사용합니다. 파트너는 두 가지 용도로 동일한 키를 사용하거나 각 용도에 별도의 키를 사용할 수 있습니다. 재해 또는 보안 침해 발생 시 데이터를 복호화할 수 있도록 신뢰할 수 있는 제3자가 에스스로에 공통 암호화 키를 보관해 둔 경우 별도의 서명 키를 사용하는 것이 좋습니다. 별도의 서명 키(에스스로하지 않음)를 사용하면 디지털 서명의 부인 방지 기능이 손상되지 않습니다.

Note

AS2 인증서의 키 길이는 최소 2,048비트, 최대 4,096비트여야 합니다.

다음 사항은 프로세스 중에 AS2 인증서가 사용되는 방법을 자세히 설명합니다.

- 인바운드 AS2
 - 거래 파트너가 서명 인증서용 퍼블릭 키를 보내면 이 키를 파트너 프로필로 가져옵니다.
 - 현지 당사자가 암호화 및 서명 인증서를 위한 퍼블릭 키를 전송합니다. 그러면 파트너가 프라이빗 키를 가져옵니다. 로컬 당사자는 서명 및 암호화를 위해 별도의 인증서 키를 보내거나 두 가지 용도로 동일한 키를 사용하도록 선택할 수 있습니다.
- 아웃바운드 AS2
 - 파트너가 암호화 인증서용 퍼블릭 키를 보내고 이 키를 파트너 프로필로 가져옵니다.
 - 로컬 당사자는 서명용 인증서의 퍼블릭 키를 보내고 서명을 위해 인증서의 프라이빗 키를 가져옵니다.
 - HTTPS를 사용하는 경우 자체 서명된 전송 계층 보안(TLS) 인증서를 가져올 수 있습니다.

인증서 생성 방법에 대한 자세한 설명은 [the section called “1단계: AS2용 인증서 생성”](#) 섹션을 참조하세요.

이 절차에서는 Transfer Family 콘솔을 사용하여 인증서를 가져오는 방법을 설명합니다. AWS CLI 대신을 사용하려면 단원을 참조하십시오 [the section called “2단계: 인증서를 Transfer Family 인증서 리소스로 가져오기”](#).

AS2 지원 인증서를 지정하려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 왼쪽 탐색 창의 AS2 거래 파트너에서 인증서를 선택합니다.
3. 인증서 가져오기를 선택합니다.
4. 인증서 구성 섹션의 인증서 설명에 인증서의 쉽게 식별할 수 있는 이름을 입력합니다. 설명으로 인증서의 용도를 식별할 수 있는지 확인하세요. 또한 인증서의 역할을 선택합니다.
5. 인증서 용도 섹션에서 이 인증서의 용도를 선택합니다. 암호화, 서명 또는 둘 다에 사용할 수 있습니다.

팁: 사용량에 대해 암호화 및 서명을 선택하면 Transfer Family는 두 개의 동일한 인증서(각각 고유한 ID가 있음)를 생성합니다. 하나는 사용 값이 ENCRYPTION이고 다른 하나는 사용 값이 입니다SIGNING.

6. 인증서 내용 섹션에서 거래 파트너의 공개 인증서를 제공하거나 로컬 인증서용 공개 및 프라이빗 키를 제공하세요.

인증서 내용 섹션에 적절한 세부 정보를 입력합니다.

- 자체 서명 인증서를 선택하는 경우 인증서 체인을 제공하지 않습니다.
- 인증서 텍스트와 해당 체인을 인증서 및 인증서 체인 필드에 붙여 넣습니다.
- 이 인증서가 로컬 인증서인 경우 프라이빗 키를 붙여 넣습니다.

7. 인증서 가져오기를 선택하여 프로세스를 완료하고 가져온 인증서의 세부 정보를 저장합니다.

Note

TLS 인증서는 파트너의 퍼블릭 인증서로만 가져올 수 있습니다. 파트너의 퍼블릭 인증서를 선택한 다음 사용량에 대해 전송 계층 보안(TLS)을 선택하면 경고가 표시됩니다. 또한 TLS 인증서는 자체 서명되어야 합니다(즉, TLS 인증서를 가져오려면 자체 서명된 인증서를 선택해야 함).

AS2 인증서 교체

인증서는 6개월에서 1년 동안 유효한 경우가 많습니다. 더 오래 유지하려는 프로필을 설정했을 수 있습니다. 이를 용이하게 하기 위해 Transfer Family는 인증서 교체 서비스를 제공합니다. 프로필에 여러 인증서를 지정하여 프로필을 여러 해 동안 계속 사용할 수 있습니다. Transfer Family는 서명(옵션) 및 암호화(필수)에 인증서를 사용합니다. 원하는 경우 두 가지 용도로 사용할 단일 인증서를 지정할 수 있습니다.

인증서 교체는 만료되는 오래된 인증서를 새 인증서로 교체하는 프로세스입니다. 이 전환은 계약 파트너가 아직 아웃바운드 전송을 위한 새 인증서를 구성하지 않았거나 새 인증서를 사용할 수 있는 기간 중에 이전 인증서로 서명 또는 암호화된 페이로드를 보내는 경우 전송이 중단되지 않도록 점진적으로 진행됩니다. 기존 인증서와 새 인증서가 모두 유효한 중간 기간을 유예 기간이라고 합니다.

X.509 인증서에는 Not Before 및 Not After 날짜가 있습니다. 하지만 이러한 파라미터는 관리자에게 충분한 통제 기능을 제공하지 못할 수 있습니다. Transfer Family는 아웃바운드 페이로드에 사용되는 인증서와 인바운드 페이로드에 허용되는 인증서를 통제하는 Active Date 및 Inactive Date 설정을 제공합니다.

인증서 만료 모니터링

Transfer Family는 인증서를 가져온 DaysUntilExpiry 후 Amazon CloudWatch 지표를 게시합니다. 지표는 현재 날짜와 인증서의 로 지정된 날짜 사이의 일수InactiveDate를 내보냅니다. 지표는 CloudWatch 지표 대시보드의 Transfer AWS 네임스페이스 아래에 있습니다.

이 지표는 항상 CertificateId에 대한 지표 차원을 가지며 고객이 인증서에 제공하는 경우 선택적으로 설명 차원을 포함합니다. CloudWatch 지표 차원에 대한 자세한 내용은 CloudWatch API 참조의 [차원](#)을 참조하세요. CloudWatch

Note

Transfer Family용 인증서를 가져온 후 이 지표를 고객 계정으로 내보내는 데 최대 하루가 걸릴 수 있습니다.

이 지표를 사용하여 인증서가 만료될 때 알려주는 CloudWatch 경보를 생성할 수 있습니다.

아웃바운드 인증서 선택은 전송 날짜 이전의 최대값을 Inactive Date로 사용합니다. 인바운드 프로세스는 범위 Not Before 및 Not After 내 및 범위 Active Date 및 Inactive Date 내에서 인증서를 수락합니다.

인증서 교체 예제

다음 표에서는 단일 프로필에 대해 두 개의 인증서를 구성할 수 있는 한 가지 방법을 설명합니다.

두 개의 인증서 교대

명칭	NOT BEFORE(인증 기관에서 통제)	ACTIVE DATE(Transfer Family에서 설정)	INACTIVE DATE(Transfer Family에서 설정)	NOT AFTER(인증 기관에서 설정)
Cert1(이전 인증서)	2019-11-01	2020-01-01	2020-12-31	2024-01-01
Cert2(최신 인증서)	2020-11-01	2020-06-01	2021-06-01	2025-01-01

유념할 사항:

- 인증서에 Active Date 및 Inactive Date를 지정하는 경우 범위는 Not Before 및 Not After의 범위 내에 있어야 합니다.
- 각 프로필에 대해 여러 인증서를 구성하여 모든 인증서의 활성 날짜 범위가 프로필을 사용하려는 기간을 포함하도록 하는 것이 좋습니다.
- 이전 인증서가 비활성화되는 시점과 새 인증서가 활성화되는 시점 사이에 약간의 유예 시간을 지정하는 것이 좋습니다. 위 예에서 첫 번째 인증서는 2020년 12월 31일까지는 비활성화되지 않지만, 두 번째 인증서는 2020년 6월 1일에 활성화되어 6개월의 유예 기간이 제공됩니다. 2020년 6월 1부터 2020년 12월 31일까지의 기간 동안 두 인증서 모두 활성 상태입니다.

AS2 프로필 생성

이 주제에서는 AS2 프로세스에 사용할 프로필을 생성하는 방법을 설명합니다. 로컬 프로필은 로컬(AS2 지원 Transfer Family 서버) 조직 또는 "파트너"를 정의합니다. 마찬가지로 파트너 프로필은 Transfer Family 외부의 원격 파트너 조직을 정의합니다.

1. [AS2 인증서 가져오기](#)
2. AS2 프로필 생성
3. [AS2 서버 생성](#)
4. [AS2 계약 생성](#)
5. [AS2 커넥터 구성](#)

이 절차를 사용하여 로컬 프로필과 파트너 프로필을 모두 생성할 수 있습니다. 이 절차에서는 Transfer Family 콘솔을 사용하여 AS2 프로필을 생성하는 방법에 대해 설명합니다. 이 방법 대신 AWS CLI를 사용하려면 [the section called “3단계: 사용자와 거래 파트너를 위한 프로필 생성”](#) 섹션을 참조하세요.

AS2 프로필을 생성하려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 왼쪽 탐색 창의 AS2 거래 파트너에서 프로필을 선택한 다음 프로필 생성을 선택합니다.
3. 프로필 구성 섹션에서 프로필의 AS2 ID를 입력합니다. 이 값은 AS2 프로토콜별 HTTP 헤더 as2-from 및 as2-to에 사용되며 거래 파트너십을 식별하고 사용할 인증서를 결정하는 등의 용도로 사용됩니다.
4. 프로필 타입 섹션에서 로컬 프로필 또는 파트너 프로필을 선택합니다.
5. 인증서 섹션의 드롭다운 메뉴에서 하나 이상의 인증서를 선택합니다.

팁: 드롭다운 메뉴에 나열되지 않은 인증서를 가져오려면 새 인증서 가져오기를 선택합니다. 그러면 인증서 가져오기 화면에 새 브라우저 창이 열립니다. 인증서 가져오기 절차에 대한 자세한 설명은 [AS2 인증서 가져오기](#) 섹션을 참조하세요.

6. (옵션) 태그 섹션에서 이 프로필을 식별하는 데 도움이 되는 키-값 쌍을 하나 이상 지정합니다.
7. 프로필 생성을 선택하여 프로세스를 완료하고 새 프로필을 저장합니다.

AS2 서버 생성

이 주제에서는 콘솔 또는 CloudFormation 템플릿을 사용하여 AS2-enabled Transfer Family 서버를 생성하는 지침을 제공합니다. 종합 예 AS2 구성에 대한 자세한 설명은 [AS2 구성 설정](#) 섹션을 참조하세요. AS2 서버를 생성한 후 서버에 계약을 추가할 수 있습니다.

1. [AS2 인증서 가져오기](#)
2. [AS2 프로필 생성](#)
3. AS2 서버 생성
4. [AS2 계약 생성](#)
5. [AS2 커넥터 구성](#)

주제

- [Transfer Family 콘솔을 사용하여 AS2 서버 생성](#)
- [템플릿을 사용하여 데모 Transfer Family AS2 스택을 생성하세요.](#)
- [AS2 계약 생성](#)

Transfer Family 콘솔을 사용하여 AS2 서버 생성

이 절차에서는 Transfer Family 콘솔을 사용하여 AS2 지원 서버를 생성하는 방법을 설명합니다. AWS CLI 대신 사용하려면 단원을 참조하십시오 [the section called “4단계: AS2 프로토콜을 사용하는 Transfer Family 서버 생성”](#).

Note

AS2 프로토콜을 사용하는 Transfer Family 서버에 파일 처리 워크플로를 연결할 수 있습니다. 하지만 AS2 메시지는 서버에 연결된 워크플로를 실행하지 않습니다.

AS2 지원 서버를 만들려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 탐색 창에서 서버를 선택한 다음 서버 생성을 선택합니다.
3. 프로토콜 선택 페이지에서 AS2(적용 가능성 선언문 2)를 선택하고 다음을 선택합니다.
4. 자격 증명 제공자 선택 페이지에서 다음을 선택합니다.

Note

AS2의 경우 AS2 프로토콜에 대해 기본 인증이 지원되지 않기 때문에 ID 제공자를 선택할 수 없습니다. 대신 Virtual Private Cloud(VPC) 보안 그룹을 통해 액세스를 통제합니다.

5. 엔드포인트 선택 페이지에서 다음을 수행합니다.

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

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

☐ FIPS Enabled endpoint

- a. 엔드포인트 타입에서 서버의 엔드포인트를 호스팅할 VPC 호스팅을 선택합니다. VPC 호스팅 엔드포인트를 설정하는 자세한 설명은 [Virtual Private Cloud\(VPC\)에 서버 생성](#) 섹션을 참조하세요.

 Note

공개적으로 액세스할 수 있는 엔드포인트는 AS2 프로토콜에서 지원되지 않습니다. 인터넷을 통해 VPC 엔드포인트에 액세스할 수 있게 하려면 액세스에서 인터넷 연결을 선택한 다음 탄력적 IP 주소를 입력합니다.

- b. 액세스에서 다음 옵션 중 하나를 선택합니다.
- 내부 - 이 옵션을 선택하면 VPC 및 VPC로 연결된 환경(예: Direct Connect 또는 VPN을 통한 온프레미스 데이터 센터 등) 내에서 액세스할 수 있습니다.
 - 인터넷 연결 - 인터넷을 통해 그리고 VPC 및 Direct Connect 또는 VPN을 통한 온프레미스 데이터 센터와 같은 VPC 연결 환경 내에서 액세스를 제공하려면 이 옵션을 선택합니다.

인터넷 연결을 선택하는 경우 메시지가 표시되면 탄력적 IP 주소를 제공하세요.

- c. VPC의 경우 기존 VPC를 선택하거나 VPC 생성을 선택하여 새 VPC를 생성합니다.
- d. FIPS 지원의 경우 FIPS 지원 엔드포인트 확인란의 선택을 취소한 상태로 유지합니다.

 Note

FIPS 지원 엔드포인트는 AS2 프로토콜에서 지원되지 않습니다.

- e. 다음을 선택합니다.
6. 도메인 선택 페이지에서 선택한 프로토콜을 사용하여 파일을 객체로 저장하고 액세스하려면 Amazon S3를 선택합니다.

다음을 선택합니다.

7. 추가 세부 정보 구성 페이지에서 필요한 설정을 선택합니다.

 Note

AS2와 함께 다른 프로토콜을 구성하는 경우 모든 추가 세부 설정이 적용됩니다. 하지만 AS2 프로토콜의 경우 CloudWatch 로깅 및 태그 섹션의 설정만 적용됩니다.

CloudWatch 로깅 역할 설정은 옵션이지만 메시지 상태를 확인하고 구성 문제를 해결할 수 있도록 로깅 역할을 설정하는 것이 좋습니다.

8. 검토 및 생성 페이지에서 옵션을 검토하여 올바른지 확인하세요.

- 설정을 편집하려면 변경하려는 단계 옆의 편집을 선택합니다.

 Note

단계를 편집하는 경우 편집하기로 선택한 단계 이후의 각 단계를 검토하는 것이 좋습니다.

- 변경 사항이 없는 경우 서버 생성을 선택하여 서버를 생성하세요. 서버 페이지로 이동하고, 새 서버가 나열되는 다음 화면이 표시됩니다.

새 서버 상태가 온라인으로 변경되기까지 몇 분 정도 걸릴 수 있습니다. 이때부터 서버는 사용자의 파일 작업을 수행할 수 있습니다.

템플릿을 사용하여 데모 Transfer Family AS2 스택을 생성하세요.

AS2-enabled Transfer Family 서버를 빠르게 생성하기 위해 독립형 CloudFormation 템플릿이 제공됩니다. 템플릿은 퍼블릭 Amazon VPC 엔드포인트, 인증서, 로컬 및 파트너 프로필, 계약, 커넥터로 서버를 구성합니다.

기본 AS2 서버 템플릿은 다음 리소스를 생성합니다.

- VPC 엔드포인트가 있는 AS2-enabled Transfer Family 서버
- 인증서가 있는 로컬 및 파트너 AS2 프로필
- 프로필 간의 계약
- 파일 스토리지용 Amazon S3 버킷
- 필수 IAM 역할 및 정책
- CloudWatch 로깅 구성

이 템플릿을 사용하기 전에 다음 사항에 유의하세요.

- 이 템플릿에서 스택을 생성할 경우, 사용한 AWS 리소스에 대한 요금이 청구됩니다.

- 템플릿은 여러 인증서를 생성하고에 배치 AWS Secrets Manager 하여 안전하게 저장합니다. 이 서비스를 사용하면 요금이 부과되므로 원하는 경우 Secrets Manager에서 이러한 인증서를 삭제할 수 있습니다. Secrets Manager에서 이러한 인증서를 삭제해도 Transfer Family 서버에서는 삭제되지 않습니다. 따라서 데모 스택의 기능은 영향을 받지 않습니다. 하지만 프로덕션 AS2 서버에서 사용할 인증서의 경우 Secrets Manager를 사용하여 저장된 인증서를 관리하고 주기적으로 교체하는 것이 좋습니다.
- 템플릿은 기본 용도로만 사용하고 주로 데모 용도로만 사용하는 것이 좋습니다. 프로덕션 환경에서 이 데모 스택을 사용하려는 경우 템플릿의 YAML 코드를 수정하여 보다 강력한 스택을 만드는 것이 좋습니다. 예를 들어 프로덕션 수준의 인증서를 만들고 프로덕션에서 사용할 수 있는 AWS Lambda 함수를 만들어 보세요.

CloudFormation 템플릿에서 AS2 지원 Transfer Family 서버를 만들려면

1. <https://console.aws.amazon.com/cloudformation> CloudFormation 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 스택을 선택합니다.
3. 스택 생성을 선택한 다음 새 리소스 사용(표준)을 선택합니다.
4. 사전 조건 - 템플릿 준비 섹션에서 기존 템플릿 선택을 선택합니다.
5. [AS2 데모 템플릿](#)인 이 링크를 복사하여 Amazon S3 URL 필드에 붙여넣습니다.
6. 다음을 선택합니다.
7. 스택 세부 정보 지정 페이지에서 스택의 이름을 지정한 다음 파라미터를 지정합니다.
 - AS2에서 로컬 AS2 ID 및 파트너 AS2 ID의 값을 입력하거나 기본값인 local 및 partner를 각각 적용합니다.
 - 네트워크에서 보안 그룹 인그레스 CIDR IP에 값을 입력하거나 기본값 0.0.0.0/0을 그대로 사용합니다.

 Note

이 값은 CIDR 형식으로 AS2 서버로 들어오는 트래픽에 허용되는 IP 주소를 지정합니다. 기본값인 0.0.0.0/0은 모든 IP 주소를 허용합니다.

- 일반에서 접두사에 값을 입력하거나 기본값인 transfer-as2을 그대로 사용합니다. 이 접두사는 스택에서 생성되는 모든 리소스 이름 앞에 위치합니다. 예를 들어, 기본 접두사를 사용하는 경우 Amazon S3 버킷의 transfer-as2-*amzn-s3-demo-bucket*라는 이름이 지정됩니다.
8. 다음을 선택합니다. 스택 옵션 구성페이지에서 다음을 다시 선택합니다.

9. 생성 중인 스택의 세부 정보를 검토한 다음 스택 생성을 선택합니다.

Note

페이지 하단의 기능에서가 AWS Identity and Access Management (IAM) 리소스를 생성할 CloudFormation 수 있음을 확인해야 합니다.

스택이 생성된 후 AWS Command Line Interface ()를 사용하여 파트너 서버에서 로컬 Transfer Family 서버로 테스트 AS2 메시지를 보낼 수 있습니다. 테스트 메시지를 보내기 위한 샘플 AWS CLI 명령은 스택의 다른 모든 리소스와 함께 생성됩니다.

이 샘플 명령을 사용하려면, 스택의 출력 탭으로 이동하여 TransferExampleAs2Command 를 복사하세요. 그런 다음 AWS CLI를 사용하여 명령을 실행할 수 있습니다. 아직 설치하지 않은 경우 AWS Command Line Interface 사용 설명서의 [의 최신 버전 설치 또는 업데이트를 AWS CLI](#) AWS CLI참조하세요.

샘플 명령의 형식은 다음과 같습니다.

```
aws s3api put-object --bucket amzn-s3-demo-bucket --key test.txt && aws transfer start-file-transfer --region aws-region --connector-id TransferConnectorId --send-file-paths /amzn-s3-demo-bucket/test.txt
```

Note

이 명령의 버전에는 스택에 있는 *amzn-s3-demo-bucket* 및 *TransferConnectorId* 리소스의 실제 값이 포함되어 있습니다.

이 샘플 명령은 && 문자열을 사용하여 서로 연결된 두 개의 개별 명령으로 구성되어 있습니다.

첫 번째 명령은 버킷에 비어 있는 새 텍스트 파일을 만듭니다.

```
aws s3api put-object --bucket amzn-s3-demo-bucket --key test.txt
```

그런 다음 두 번째 명령은 커넥터를 사용하여 파트너 프로파일에서 로컬 프로파일로 파일을 보냅니다. Transfer Family 서버에는 로컬 프로파일의 파트너 프로파일의 메시지를 수락하도록 허용하는 계약이 설정되어 있습니다.

```
aws transfer start-file-transfer --region aws-region --connector-id TransferConnectorId
--send-file-paths /amzn-s3-demo-bucket/test.txt
```

명령을 실행한 후 Amazon S3 버킷(*amzn-s3-demo-bucket*)으로 이동하여 콘텐츠를 볼 수 있습니다. 명령이 제대로 실행되면 버킷에 다음과 같은 객체가 표시되어야 합니다.

- *processed/*— 이 폴더에는 전송된 파일과 MDN 응답을 설명하는 JSON 파일이 들어 있습니다.
- *processing/*— 이 폴더에는 처리 중인 파일이 일시적으로 포함되지만 전송이 완료된 후에는 이 폴더가 비어 있어야 합니다.
- *server-id/*— 이 폴더의 이름은 Transfer Family 서버 ID에 따라 붙여집니다. 여기에는 *from-partner* (이 폴더의 이름은 파트너의 AS2 ID에 따라 동적으로 붙여짐)이고, 이것은 그 자체가 *failed/processed/*, 및 *processing/* 폴더가 들어 있습니다. */server-id/from-partner/processed/* 폴더에는 전송된 텍스트 파일의 사본과 해당 JSON 및 MDN 파일이 들어 있습니다.
- *test.txt*— 이 객체는 전송된 (빈) 파일입니다.

AS2 계약 생성

계약은 Transfer Family 서버와 연결되어 있습니다. AS2 프로토콜을 사용하는 거래 파트너가 Transfer Family를 사용하여 메시지 또는 파일을 교환하는 인바운드 전송, 즉 파트너 소유의 외부 소스에서 Transfer Family 서버로 AS2 파일을 보내는 방식으로 메시지나 파일을 교환하는 거래 파트너에 대한 세부 정보를 지정합니다.

이 절차에서는 Transfer Family 콘솔을 사용하여 AS2 계약서를 생성하는 방법에 대해 설명합니다. AWS CLI 대신을 사용하려면 단원을 참조하십시오 [the section called “5단계: 사용자와 사용자의 파트너 간의 계약서 작성”](#).

Transfer Family 서버에 대한 계약을 만들려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 서버를 선택한 다음 AS2 프로토콜을 사용하는 서버를 선택합니다.

또는 AS2 프로토콜을 사용하는 Transfer Family 서버가 하나 이상 있는 경우 계약을 선택하여 AS2 AS2 거래 파트너 메뉴에서 메시지를 수신합니다. 그런 다음 계약 생성 화면에서이 계약을 연결할 AS2 서버를 선택합니다.

3. 서버 세부 정보 페이지에서 계약 섹션으로 스크롤합니다.

4. 계약 추가를 선택합니다.

5. 다음과 같이 계약 파라미터를 입력합니다:

- a. 계약 구성 섹션에 설명적인 이름을 입력합니다. 이름으로 계약의 목적을 식별할 수 있는지 확인하세요. 또한 계약 상태를 활성(기본적으로 선택됨) 또는 비활성으로 설정합니다.
- b. 통신 구성 섹션에서 로컬 프로필과 파트너 프로필을 선택합니다. 또한 메시지 서명을 적용할지 여부를 선택합니다.
 - 기본적으로 메시지 서명 적용이 활성화되어 있습니다. 즉, Transfer Family는 이 계약에 대해 거래 파트너의 서명되지 않은 메시지를 거부합니다.
 - Transfer Family가 이 계약에 대해 거래 파트너의 서명되지 않은 메시지를 수락하도록 허용하려면 이 설정을 선택 취소합니다.
- c. 받은 편지함 디렉터리 구성 섹션에서 다음 정보를 제공합니다.
 - AS2 메시지, MDN 파일 및 JSON 상태 파일을 저장할 별도의 디렉터리 지정을 선택할지 여부를 결정합니다.
 - 이 옵션을 선택하면 페이로드 파일, 실패한 파일, MDN 파일, 상태 파일 및 임시 파일에 대해 별도의 위치를 지정합니다.
 - 이 옵션을 선택 취소하면 모든 AS2 파일이 기본 디렉터리에 지정한 위치로 이동합니다.
 - S3 버킷에서 Amazon S3 버킷을 선택합니다.
 - 접두사에 버킷에 파일을 저장하는 데 사용할 접두사(폴더)를 입력할 수 있습니다.

예를 들어 버킷과 접두사 **amzn-s3-demo-bucket**에 **incoming**를 입력하면 AS2 파일이 **/amzn-s3-demo-bucket/incoming** 폴더에 저장됩니다.

 - AWS IAM 역할에서 지정한 버킷에 액세스할 수 있는 역할을 선택합니다.
 - 파일 이름 보존에서 수신 AS2 메시지 페이로드의 원래 파일 이름을 보존할지 여부를 선택합니다.
 - 이 설정을 선택하면 파일이 Amazon S3에 저장될 때 거래 당사자가 제공한 파일 이름이 보존됩니다.
 - 이 설정을 선택 취소하면 Transfer Family가 파일을 저장할 때에 설명된 대로 파일 이름이 조정됩니다 [파일 이름 및 위치](#).
- d. (옵션) 태그 섹션에서 태그를 추가합니다.
- e. 계약에 대한 모든 정보를 입력한 후 계약 생성을 선택합니다.

AS2 커넥터 구성

커넥터의 목적은 아웃바운드 전송을 위해 거래 파트너들 사이의 관계를 설정하는 것입니다. 즉, Transfer Family 서버에서 파트너 소유의 외부 대상으로 AS2 파일을 보내는 것입니다. 커넥터의 경우 로컬 및 파트너 프로필을 생성하여 로컬 당사자, 원격 파트너 및 인증서를 지정합니다.

커넥터를 설치한 후에는 거래 파트너에게 정보를 전송할 수 있습니다. 각 AS2 서버에는 세 개의 고정 IP 주소가 할당됩니다. AS2 커넥터는 이러한 IP 주소를 사용하여 AS2를 통해 거래 파트너에게 비동기 MDNs 전송합니다.

Note

거래 파트너가 수신한 메시지 크기는 Amazon S3의 객체 크기와 일치하지 않습니다. AS2 메시지가 파일을 보내기 전에 봉투에 싸서 보내기 때문에 이러한 불일치가 발생합니다. 따라서 파일을 압축하여 전송하더라도 파일 크기가 커질 수 있습니다. 따라서 거래 파트너의 최대 파일 크기가 보내는 파일 크기보다 커야 합니다.

1. [AS2 인증서 가져오기](#)
2. [AS2 프로필 생성](#)
3. [AS2 서버 생성](#)
4. [AS2 계약 생성](#)
5. AS2 커넥터 생성

AS2 커넥터 생성

이 절차에서는 AWS Transfer Family 콘솔을 사용하여 AS2 커넥터를 생성하는 방법에 대해 설명합니다. AWS CLI 대신을 사용하려면 단원을 참조하십시오 [the section called “6단계: 사용자와 사용자의 거래 파트너 간의 커넥터를 생성합니다.”](#).

AS2 커넥터를 생성하려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 왼쪽 탐색 창의 AS2 거래 파트너 메뉴에서 메시지를 전송할 커넥터를 선택한 다음 AS2 커넥터 생성을 선택합니다.
3. 커넥터 구성 섹션에서 다음 정보를 지정합니다:

- URL - 아웃바운드 연결의 URL을 입력합니다.
- IP 주소 지정 모드 - IPv4 주소를 단독으로 사용하려면 IPv4 전용(기본값)을 선택하고 IPv6 및 IPv4를 모두 지원하려면 듀얼 스택을 선택합니다. IPv4 듀얼 스택 모드에서 커넥터는 DNS 확인이 IPv6 결과를 반환할 때 IPv6를 선호하고 IPv4 DNS 결과만 반환될 때 IPv4를 사용합니다.
- 액세스 역할 - 사용할 (IAM) 역할의 Amazon 리소스 이름 AWS Identity and Access Management (ARN)을 선택합니다. 이 역할이 StartFileTransfer 요청에서 사용된 파일 위치의 상위 디렉터리에 대한 읽기 및 쓰기 액세스 권한을 제공하는지 확인하세요. 또한 그 역할이 StartFileTransfer와 함께 전송하려는 파일의 상위 디렉터리에 대한 읽기 및 쓰기 액세스 권한을 제공하는지 확인하세요.

 Note

커넥터에 기본 인증을 사용하는 경우 액세스 역할에는 암호에 대한 `secretsmanager:GetSecretValue` 권한이 필요합니다. AWS 관리형 키에서 대신 고객 관리형 키를 사용하여 보안 암호를 암호화하는 AWS Secrets Manager 경우 역할에는 해당 키에 대한 `kms:Decrypt` 권한도 필요합니다. 접두어 `aws/transfer/`를 사용하여 암호의 명칭을 지정하는 경우 [암호 생성 권한 예](#)에 나와 있는 것처럼 와일드카드 문자(*)를 사용하여 필요한 권한을 추가할 수 있습니다.

- 로깅 역할(옵션) - CloudWatch 로그에 이벤트를 푸시하는 데 사용할 커넥터의 IAM 역할을 선택합니다.
4. AS2 구성 섹션에서 로컬 및 파트너 프로필, 암호화 및 서명 알고리즘, 전송된 정보의 압축 여부를 선택합니다. 다음 사항에 유의하세요.
- S3 콘텐츠 유형 보존 파라미터는 기본적으로 활성화됩니다.
- 설정하면 Transfer Family는 파일 확장명을 기반으로 콘텐츠 유형을 매핑Content-Type하는 대신 S3의 객체와 연결된 Amazon S3를 사용합니다. 서비스가 S3 객체의 콘텐츠 유형을 사용하는 대신 파일 확장명을 기반으로 AS2 메시지의 콘텐츠 유형을 매핑하도록 하려면이 설정을 지웁니다.
- 암호화 알고리즘의 경우 취약한 암호화 알고리즘이므로 암호화 알고리즘이 필요한 레거시 클라이언트를 지원해야 하는 DES_EDE3_CBC 경우가 아니면 선택하지 마십시오.
 - 주제는 커넥터와 함께 전송되는 AS2 메시지의 subject HTTP 헤더 속성으로 사용됩니다.
 - 암호화 알고리즘 없이 커넥터를 생성하도록 선택한 경우를 프로토콜HTTPS로 지정해야 합니다.
5. 기본 인증 섹션에서 다음 정보를 지정합니다.

- 아웃바운드 메시지와 함께 로그인 자격 증명을 보내려면 기본 인증 활성화를 선택합니다. 아웃바운드 메시지와 함께 자격 증명을 보내지 않으려면 기본 인증 활성화를 선택 취소하세요.
- 인증을 사용하는 경우 암호를 선택하거나 생성하세요.
- 새 암호를 만들려면 새 암호 생성을 선택한 다음 사용자 이름과 암호를 입력합니다. 이러한 자격 증명은 파트너의 엔드포인트에 연결하는 사용자와 일치해야 합니다.

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

test-user

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.

- 암호를 사용하려면 기존 암호 선택을 선택한 후 드롭다운 메뉴에서 암호를 선택합니다. Secrets Manager에서 올바른 형식의 암호를 만드는 방법에 대한 자세한 설명은 [AS2 커넥터에 대한 기본 인증을 활성화합니다](#) 섹션을 참조하세요.

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

Choose a secret

aws/transfer/...
SFTP connector...

aws/transfer/sftp-connector1
Secret key ...

aws/transfer/c...

or to provide AWS Transfer Family with permission to credentials.

6. MDN 구성 섹션에서 다음 정보를 지정합니다.

- MDN 요청 - 거래 파트너가 AS2를 통해 메시지를 성공적으로 수신한 후 MDN을 보내도록 요청할 수 있습니다.
- 서명된 MDN - MDN에 서명을 요구하는 옵션이 있습니다. 이 옵션은 MDN 요청을 선택한 경우에만 사용할 수 있습니다.

7. 모든 설정을 확인한 후 AS2 커넥터 생성을 선택하여 커넥터를 생성합니다.

새 커넥터의 ID가 목록에 추가된 커넥터 페이지가 나타납니다. 커넥터에 대한 세부 정보를 보려면 [AS2 커넥터 세부 정보 보기](#) 섹션을 참조하세요.

AS2 커넥터 알고리즘

AS2 커넥터를 생성하면 다음 보안 알고리즘이 커넥터에 연결됩니다.

유형	Algorithm
TLS 암호	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
	TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256

유형	Algorithm
	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

AS2 커넥터의 기본 인증

AS2 프로토콜을 사용하는 Transfer Family 서버를 만들거나 업데이트할 때 아웃바운드 메시지에 대한 기본 인증을 추가할 수 있습니다. 커넥터에 인증 정보를 추가하여 이 작업을 수행합니다.

Note

기본 인증은 HTTPS를 사용하는 경우에만 사용할 수 있습니다.

커넥터에 인증을 사용하려면 기본 인증 섹션에서 기본 인증 활성화를 선택합니다. 기본 인증을 활성화한 후 새 비밀을 생성하거나 기존 비밀을 사용할 수 있습니다. 어느 경우든 비밀의 자격 증명은 이 커넥터를 사용하는 아웃바운드 메시지와 함께 전송됩니다. 자격 증명은 거래 파트너의 원격 엔드포인트에 연결하려는 사용자와 일치해야 합니다.

다음 스크린샷은 기본 인증 활성화를 선택하고 새 비밀 만들기를 선택한 것을 보여줍니다. 이러한 선택을 한 후 비밀에 대한 사용자 이름과 암호를 입력할 수 있습니다.

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
test-user

Password

[i](#) Update the [access role associated with your connector](#) to provide AWS Transfer Family with permission to read the secret containing your Basic authentication credentials.

다음 스크린샷은 기본 인증 활성화를 선택하고 기존 비밀을 선택한 것을 보여줍니다. 비밀은 [AS2 커넥터에 대한 기본 인증을 활성화합니다.](#)에 설명된 대로 올바른 형식이어야 합니다.

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

Choose a secret 

Q |

- aws/transfer/ SFTP connector
- aws/transfer/sftp-connector1 Secret key
- aws/transfer/c

[i](#) Update the [access role associated with your connector](#) to provide AWS Transfer Family with permission to read the secret containing your Basic authentication credentials.

AS2 커넥터에 대한 기본 인증을 활성화합니다.

AS2 커넥터에 대한 기본 인증을 활성화하면 Transfer Family 콘솔에서 새 암호를 만들거나 AWS Secrets Manager에서 만든 암호를 사용할 수 있습니다. 어느 경우든 암호는 Secrets Manager에 저장됩니다.

주제

- [콘솔을 사용하여 새 암호 생성](#)
- [기존 암호 사용](#)
- [에서 보안 암호 생성 AWS Secrets Manager](#)

콘솔을 사용하여 새 암호 생성

콘솔에서 커넥터를 만들 때 새 암호를 생성할 수 있습니다.

새 암호를 만들려면 새 암호 생성을 선택한 다음 사용자 이름과 암호를 입력합니다. 이러한 자격 증명은 파트너의 엔드포인트에 연결하는 사용자와 일치해야 합니다.

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

test-user

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

콘솔에서 새 암호를 만들면 암호의 명칭은 `/aws/transfer/connector-id`과 같은 명명 규칙을 따릅니다. 여기서 **connector-id**는 만들려는 커넥터의 ID입니다. AWS Secrets Manager에서 암호를 찾으려고 할 때는 이 점을 고려하세요.

기존 암호 사용

콘솔에서 커넥터를 만들 때 기존 암호를 지정할 수 있습니다.

암호를 사용하려면 기존 암호 선택을 선택한 후 드롭다운 메뉴에서 암호를 선택합니다. Secrets Manager에서 올바른 형식의 암호를 만드는 방법에 대한 자세한 설명은 [에서 보안 암호 생성 AWS Secrets Manager](#) 섹션을 참조하세요.

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

Choose a secret

- aws/transfer/... SFTP connector...
- aws/transfer/sftp-connector1
Secret key ...
- aws/transfer/c...

에서 보안 암호 생성 AWS Secrets Manager

다음 절차는 AS2 커넥터에 사용할 적절한 암호를 만드는 방법을 설명합니다.

Note

기본 인증은 HTTPS를 사용하는 경우에만 사용할 수 있습니다.

AS2 Basic 인증을 위해 Secrets Manager에 사용자 자격 증명을 저장하려면

1. 에 로그인 AWS Management Console 하고 <https://console.aws.amazon.com/secretsmanager/> AWS Secrets Manager 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 암호를 선택합니다.
3. 암호 페이지에서 새 암호 저장을 선택합니다.
4. 암호 선택 페이지의 암호 타입에서 다른 타입의 암호를 선택합니다.
5. 키/값 쌍 섹션에서 키/값 탭을 선택합니다.
 - 키 - **Username**를 입력합니다.
 - 값 - 파트너 서버에 연결할 권한이 있는 사용자의 이름을 입력합니다.

6. 암호를 제공하려면 **행 추가**를 선택하고 **키/값 쌍** 섹션에서 **키/값** 탭을 선택합니다.

행 추가를 선택하고 **키/값 쌍** 섹션에서 **키/값** 탭을 선택합니다.
 - **키** - **Password**를 입력합니다.
 - **값** - 사용자의 암호를 입력합니다.
7. 프라이빗 키를 제공하려면 **행 추가**를 선택하고 **키/값 쌍** 섹션에서 **키/값** 탭을 선택합니다.
 - **키** - **PrivateKey**를 입력합니다.
 - **값** - 사용자의 프라이빗 키를 입력합니다. 이 값은 OpenSSH 형식으로 저장되어야 하며 원격 서버에 이 사용자에게 대해 저장된 퍼블릭 키와 일치해야 합니다.
8. 다음을 선택합니다.
9. 암호 구성 페이지에 암호를 위한 명칭과 설명을 입력합니다. 이름에 **aws/transfer/**이라는 접두사를 사용하는 것이 좋습니다. 예컨대, 암호 **aws/transfer/connector-1**을 지정할 수 있습니다.
10. 다음을 선택한 후 교체 구성 페이지의 기본값을 그대로 사용합니다. 그 다음 다음을 선택합니다.
11. 검토 페이지에서 저장을 선택하여 암호를 만들고 저장합니다.

암호를 만든 후 커넥터를 만들 때 암호를 선택할 수 있습니다([AS2 커넥터 구성](#) 참조). 기본 인증을 활성화하는 단계에서 사용 가능한 암호의 드롭다운 목록에서 암호를 선택합니다.

AS2 커넥터 세부 정보 보기

AWS Transfer Family 콘솔에서 AS2 AWS Transfer Family connector에 대한 세부 정보 및 속성 목록을 찾을 수 있습니다. AS2 커넥터의 속성에는 URL, 역할, 프로필, MDN, 태그 및 모니터링 지표가 포함됩니다.

이것은 커넥터 세부 정보 보기 절차입니다.

커넥터 세부 정보를 보려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 커넥터를 선택합니다.
3. 커넥터 ID 옆에서 식별자를 선택하면 선택한 커넥터의 세부 정보 페이지가 표시됩니다.

편집을 선택하여 커넥터의 세부 정보 페이지에서 AS2 커넥터의 속성을 변경할 수 있습니다.

C- [redacted]

Delete

Connector configuration

Info

Edit

Name

my-as2-connector

Access role

[redacted]

URL

<https://www.example.com>

Logging role

[AWSTransferLoggingAccess](#)

Communication settings

Info

AS2-From header

[redacted]

AS2-To header

[redacted]

AS2 configuration

Info

Edit

Local profile

[redacted]

Compression

Enabled

Encryption algorithm

AES128_CBC

Partner profile

[redacted]

Message Subject

View

Signing algorithm

SHA384

Preserve S3 Content-Type

Enabled

MDN configuration

Info

Edit

Request MDN

Enabled

Signed MDN

Default to message signing algorithm: SHA384

Synchronization

Enabled

Basic authentication [Info](#)[Edit](#)**Basic authentication**
Enabled**Secret**
aws/transfer/c- [REDACTED]**Egress IP details** [Info](#)

Service managed static IP addresses of this connector

23.23.130.9

44.196.109.11

3.219.16.7

Tags (1)[Manage tags](#)

Use tags to organize, search, and filter your Connectors. We recommend adding a tag with 'Name' as the key and a unique string as the value to easily identify your Connector.

Find resources

< 1 >

Key	Value
Name	my-as2-connector

AS2 Monitoring

3h

1d

1w

UTC timezone

UTC timezone

UTC timezone

UTC timezone

OutboundMessage

OutboundMessage

OutboundFailedMessage

OutboundFailedMessage

1.00 No data available.
Try adjusting the dashboard time range.

0.50

0

13:05 16:00

1.00 No data available.
Try adjusting the dashboard time range.

0.50

0

13:05 16:00

Note

다음 AWS Command Line Interface (AWS CLI 명령:

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

자세한 내용은 API 참조의 [DescribeConnector](#)를 참조하세요.

AS2 메시지 전송 및 수신

이 섹션에서는 AS2 메시지를 보내고 받는 프로세스를 설명합니다. 또한 AS2 메시지와 연결된 파일 이름 및 위치에 대한 세부 정보도 제공합니다.

다음 표에는 AS2 메시지에 사용할 수 있는 암호화 알고리즘과 이를 사용할 수 있는 시기가 나와 있습니다.

암호화 알고리즘	HTTP	HTTPS	참고
AES128_CBC	Yes	Yes	
AES192_CBC	Yes	Yes	
AES256_CBC	Yes	Yes	
DES_EDE3_CBC	Yes	Yes	Only use this algorithm if you must support a legacy client that requires it, as it is a weak encryption algorithm.
NONE	No	Yes	If you are sending messages to a Transfer Family server, you can only select NONE if you are using an Application Load Balancer (ALB).

주제

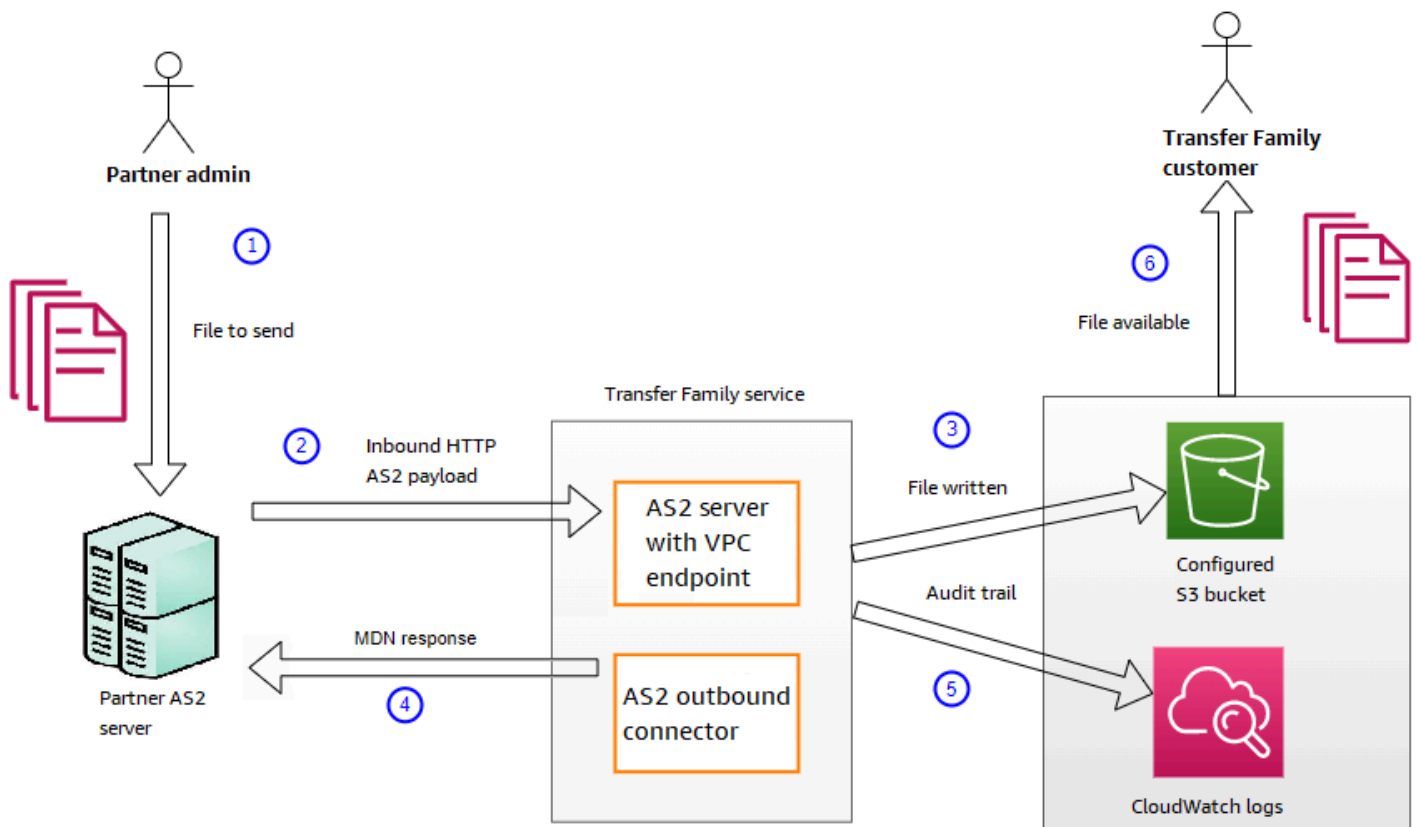
- [AS2 메시지 프로세스 수신](#)
- [HTTPS를 통한 AS2 메시지 전송 및 수신](#)
- [AS2 커넥터를 사용하여 파일 전송](#)
- [파일 이름 및 위치](#)
- [상태 코드](#)

• 샘플 JSON 파일

AS2 메시지 프로세스 수신

인바운드 프로세스는 AWS Transfer Family 서버로 전송되는 메시지 또는 파일로 정의됩니다. 인바운드 메시지의 순서는 다음과 같습니다.

1. 관리자 또는 자동 프로세스는 파트너의 원격 AS2 서버에서 AS2 파일 전송을 시작합니다.
2. 파트너의 원격 AS2 서버가 파일 내용에 서명하고 암호화한 다음 Transfer Family에서 호스팅되는 AS2 인바운드 엔드포인트에 HTTP POST 요청을 보냅니다.
3. Transfer Family는 서버, 파트너, 인증서 및 계약에 대해 구성된 값을 사용하여 AS2 페이로드를 복호화하고 확인합니다. 파일 콘텐츠는 구성된 Amazon S3 파일 스토어에 저장됩니다.
4. 서명된 MDN 응답은 HTTP 응답과 함께 인라인으로 반환되거나 별도의 HTTP POST 요청을 통해 비동기적으로 소스 서버로 반환됩니다.
5. 감사 추적은 거래소에 대한 세부 정보와 함께 Amazon CloudWatch에 작성됩니다.
6. 암호가 복호화된 파일은 inbox/processed라는 폴더에서 사용할 수 있습니다.



HTTPS를 통한 AS2 메시지 전송 및 수신

이 섹션에서는 AS2 프로토콜을 사용하여 HTTPS를 통해 메시지를 보내고 받는 Transfer Family 서버를 구성하는 방법에 대해 설명합니다.

주제

- [HTTPS를 통해 AS2 메시지를 보냅니다.](#)
- [HTTPS를 통해 AS2 메시지를 받습니다.](#)

HTTPS를 통해 AS2 메시지를 보냅니다.

HTTPS를 사용하여 AS2 메시지를 보내려면 다음 정보가 포함된 커넥터를 만드세요.

- URL의 경우 HTTPS URL을 지정하세요.
- 암호화 알고리즘에서 사용 가능한 알고리즘을 선택합니다.

Note

암호화를 사용하지 않고 Transfer Family 서버로 메시지를 보내려면(즉, 암호화 알고리즘 NONE으로 선택) Application Load Balancer(ALB)를 사용해야 합니다.

- [AS2 커넥터 구성](#)에 설명된 대로 커넥터의 나머지 값을 입력합니다.

HTTPS를 통해 AS2 메시지를 받습니다.

AWS Transfer Family AS2 서버는 현재 포트 5080을 통한 HTTP 전송만 제공합니다. 그러나 선택한 포트와 인증서를 사용하여 Transfer Family 서버 VPC 엔드포인트 앞에 있는 네트워크 또는 애플리케이션 로드 밸런서에서 TLS를 종료할 수 있습니다. 이 방법을 사용하면 들어오는 AS2 메시지가 HTTPS를 사용하도록 할 수 있습니다.

사전 조건

- VPC는 Transfer Family 서버 AWS 리전 와 동일한에 있어야 합니다.
- VPC의 서브넷은 서버를 사용하려는 가용 영역 내에 있어야 합니다.

Note

각 Transfer Family 서버는 최대 3개의 가용 영역을 지원할 수 있습니다.

- 서버와 동일한 지역에 엘라스틱 IP 주소를 최대 3개까지 할당하세요. 또는 고유한 IP 주소 범위 (BYOIP) 를 가져오도록 선택할 수 있습니다.

Note

엘라스틱 IP 주소 수는 서버 엔드포인트에서 사용하는 가용 영역 수와 일치해야 합니다.

Network Load Balance(NLB) 또는 Application Load Balancer(ALB)를 구성할 수 있습니다. 다음 표에는 각 접근 방식에 대한 장단점이 나열되어 있습니다.

아래 표에는 NLB와 ALB를 사용하여 TLS를 종료할 때의 기능 차이가 나와 있습니다.

기능	Network Load Balancer(NLB)	Application Load Balancer(ALB)
Latency	네트워크 계층에서 작동할 때 지연 시간이 줄어듭니다.	애플리케이션 계층에서 작동할 때 지연 시간이 길어집니다.
고정 IP 지원	정적일 수 있는 탄력적 IP 주소를 연결할 수 있습니다.	탄력적 IP 주소를 연결할 수 없음:는 기본 IP 주소가 변경될 수 있는 도메인을 제공합니다.
고급 라우팅	고급 라우팅을 지원하지 않습니다.	고급 라우팅을 지원합니다. 암호화 없이 AS2에 필요한 X-Forwarded-Proto 헤더를 주입할 수 있습니다. 이 헤더는 developer.mozilla.org 웹 사이트의 X-Forwarded-Proto 에 설명되어 있습니다.
TLS/SSL 종료	TLS/SSL 종료 지원	TLS/SSL 종료 지원

기능	Network Load Balancer(NLB)	Application Load Balancer(ALB)
상호 TLS(mTLS)	Transfer Family는 현재 mTLS 용 NLB 사용을 지원하지 않습니다.	mTLS 지원

Configure NLB

이 절차에서는 VPC에서 인터넷 경계 Network Load Balancer(NLB)를 설정하는 방법을 설명합니다.

Network Load Balancer를 생성하고 서버의 VPC 엔드포인트를 로드 밸런서의 대상으로 정의하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon Elastic Compute Cloud 콘솔을 엽니다.
2. 탐색 창에서, 로드 밸런서를 선택한 다음, 로드 밸런서 만들기를 선택합니다.
3. Network Load Balancer에서 [생성(Create)]을 선택합니다.
4. 기본 구성 섹션에서 다음 정보를 입력합니다.
 - 이름에 로드 밸런서를 설명하는 이름을 입력합니다.
 - 체계에서 인터넷 연결을 선택합니다.
 - IP 주소 타입에서 Ipv4를 선택합니다.
5. 네트워크 매핑 섹션에서 다음 정보를 입력합니다.
 - VPC의 경우 생성한 Virtual Private Cloud(VPC)를 선택합니다.
 - 매핑아래에서 서버 엔드포인트에서 사용하는 것과 동일한 VPC에서 사용할 수 있는 퍼블릭 서브넷과 연결된 가용 영역을 선택합니다.
 - 각 서브넷의 IPv4 주소에 대해 할당한 탄력적 IP 주소 중 하나를 선택합니다.
6. 리스너 및 라우팅 섹션에서 다음 정보를 입력합니다.
 - 프로토콜에서 TCP를 선택합니다.
 - 포트에서 **5080**를 입력합니다.
 - 기본 작업, 대상 그룹 생성을 선택합니다. 새 대상 그룹을 만드는 세부 사항에 대한 자세한 내용은 [대상 그룹 생성](#)을 참조하세요.

대상 그룹을 생성한 후 기본 작업 필드에 해당 이름을 입력합니다.

7. 보안 리스너 설정 섹션에서 기본 SSL/TLS 인증서 영역에서 인증서를 선택합니다.
8. 로드밸런서 만들기를 선택해서 사용자의 NLB를 생성합니다.
9. (옵션이지만 권장됨)[Network Load Balancer의 액세스 로그](#)에 설명된 대로 Network Load Balancer의 액세스 로그를 켜서 전체 감사 추적을 유지하세요.

NLB에서 TLS 연결이 종료되므로 이 단계를 사용하는 것이 좋습니다. 따라서 Transfer Family AS2 CloudWatch 로그 그룹에 반영되는 소스 IP 주소는 거래 파트너의 외부 IP 주소가 아니라 NLB의 사설 IP 주소입니다.

Configure ALB

이 절차에서는 VPC에서 Application Load Balancer(ALB)를 설정하는 방법을 설명합니다.

Application Load Balancer를 생성하고 서버의 VPC 엔드포인트를 로드 밸런서의 대상으로 정의하려면

1. <https://console.aws.amazon.com/ec2/>에서 Amazon Elastic Compute Cloud 콘솔을 엽니다.
2. 탐색 창에서, 로드 밸런서를 선택한 다음, 로드 밸런서 만들기를 선택합니다.
3. Application Load Balancer 아래에서 생성(Create)을 선택합니다.
4. ALB 콘솔에서 포트 443(HTTPS)에 새 HTTP 리스너를 생성합니다.
5. (선택 사항). 상호 인증(mTLS)을 설정하려면 보안 설정과 트러스트 스토어를 구성합니다.
 - a. SSL/TLS 인증서를 리스너에 연결합니다.
 - b. 클라이언트 인증서 처리에서 상호 인증(mTLS)을 선택합니다.
 - c. 트러스트 스토어로 확인을 선택합니다.
 - d. 고급 mTLS 설정에서 CA 인증서를 업로드하여 트러스트 스토어를 선택하거나 생성합니다.
6. 새 대상 그룹을 생성하고 Transfer Family AS2 서버 엔드포인트의 프라이빗 IP 주소를 포트 5080의 대상으로 추가합니다. 새 대상 그룹을 만드는 세부 사항에 대한 자세한 내용은 [대상 그룹 생성](#)을 참조하세요.
7. 포트 5080에서 HTTP 프로토콜을 사용하도록 대상 그룹에 대한 상태 확인을 구성합니다.
8. 리스너에서 대상 그룹으로 HTTPS 트래픽을 전달하는 새 규칙을 생성합니다.
9. SSL/TLS 인증서를 사용하도록 리스너를 구성합니다.

로드 밸런서를 설정하고 나면 클라이언트는 사용자 지정 포트 리스너를 통해 로드 밸런서와 통신합니다. 그러면 로드 밸런서가 포트 5080을 통해 서버와 통신합니다.

대상 그룹 생성

1. 이전 절차에서 대상 그룹 만들기를 선택하면 새 대상 그룹에 대한 그룹 세부 정보 지정 페이지로 이동합니다.
2. 기본 구성 섹션에서, 다음 정보를 입력합니다.
 - 대상 타입 선택에서 IP 주소를 선택합니다.
 - 대상 그룹 이름에 대상 그룹의 이름을 입력합니다.
 - 프로토콜의 경우 선택은 ALB를 사용하는지 NLB를 사용하는지에 따라 달라집니다.
 - Network Load Balancer(NLB)의 경우 TCP를 선택합니다.
 - Application Load Balancer(ALB)의 경우 HTTP를 선택합니다.
 - 포트에서 **5080**를 입력합니다.
 - IP 주소 타입에서 Ipv4를 선택합니다.
 - VPC 에서 Transfer Family AS2 서버용으로 생성한 VPC를 선택합니다.
3. 상태 확인 섹션에서 상태 확인 프로토콜을 선택합니다.
 - ALB의 경우 HTTP를 선택합니다.
 - NLB의 경우 TCP를 선택합니다.
4. 다음을 선택합니다.
5. 대상 등록 페이지에서 다음 정보를 입력합니다.
 - 네트워크의 경우, Transfer Family AS2 서버용으로 만든 VPC가 지정되었는지 확인합니다.
 - IPv4 주소에 Transfer Family AS2 서버 엔드포인트의 프라이빗 IPv4 주소를 입력합니다.

서버에 엔드포인트가 두 개 이상 있는 경우, IPv4 주소 추가를 선택하여 다른 IPv4 주소를 입력하는 행을 하나 더 추가합니다. 모든 서버 엔드포인트의 사설 IP 주소를 입력할 때까지 이 프로세스를 반복합니다.

 - Ports가 **5080**로 설정되어 있는지 확인하세요.
 - 아래 보류 중인 항목으로 포함을 선택하여 항목을 대상 검토 섹션에 추가하세요.
6. 대상 검토 섹션에서 IP 대상을 검토하세요.
7. 대상 그룹 생성을 선택한 다음 이전 NLB 생성 절차로 돌아가 표시된 위치에 새 대상 그룹을 입력합니다.

탄력적 IP 주소에서 서버 액세스 테스트

Network Load Balancer의 엘라스틱 IP 주소 또는 DNS 이름을 사용하여 사용자 지정 포트를 통해 서버에 연결합니다.

Important

로드 밸런서에 구성된 서브넷의 [네트워크 액세스 제어 목록\(네트워크 ACL\)](#)을 사용하여 클라이언트 IP 주소를 통한 서버 액세스를 관리합니다. 네트워크 ACL 권한은 서브넷 수준에서 설정되므로 해당 서브넷을 사용하는 모든 리소스에 규칙이 적용됩니다. 로드 밸런서의 대상 타입이 인스턴스가 아닌 IP 주소로 설정되어 있기 때문에 보안 그룹을 사용하여 클라이언트 IP 주소에서의 액세스를 제어할 수 없습니다. 따라서 로드 밸런서는 원본 IP 주소를 보존하지 않습니다. [Network Load Balancer의 상태 확인](#)이 실패하면 로드 밸런서가 서버 엔드포인트에 연결할 수 없다는 의미입니다. 이 문제를 해결하려면 다음을 확인하세요.

- 서버 [엔드포인트의 관련 보안 그룹](#)이 로드 밸런서에 구성된 서브넷으로부터의 인바운드 연결을 허용하는지 확인하세요. 로드 밸런서는 포트 5080을 통해 서버 엔드포인트에 연결할 수 있어야 합니다.
- 서버 상태가 온라인인지 확인합니다.

AS2 커넥터를 사용하여 파일 전송

AS2 커넥터는 Transfer Family 서버에서 파트너 소유의 외부 대상으로 AS2 메시지를 전송하기 위해 거래 파트너들 사이의 관계를 설정합니다.

Transfer Family를 사용하여 다음 start-file-transfer AWS Command Line Interface (AWS CLI) 명령과 같이 커넥터 ID와 파일 경로를 참조하여 AS2 메시지를 보낼 수 있습니다.

```
aws transfer start-file-transfer --connector-id c-1234567890abcdef0 \
--send-file-paths "/amzn-s3-demo-source-bucket/myfile1.txt" "/amzn-s3-demo-source-bucket/myfile2.txt"
```

커넥터의 세부 정보를 가져오려면 다음 명령을 실행합니다.

```
aws transfer list-connectors
```

이 list-connectors 명령은 커넥터의 커넥터 ID, URL 및 Amazon 리소스 이름(ARN)을 반환합니다.

특정 커넥터의 속성을 반환하려면 사용하려는 ID로 다음 명령을 실행합니다.

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

이 `describe-connector` 명령은 커넥터의 URL, 역할, 프로필, mDNS (메시지 처리 알림), 태그 및 모니터링 메트릭을 포함하여 커넥터의 모든 속성을 반환합니다.

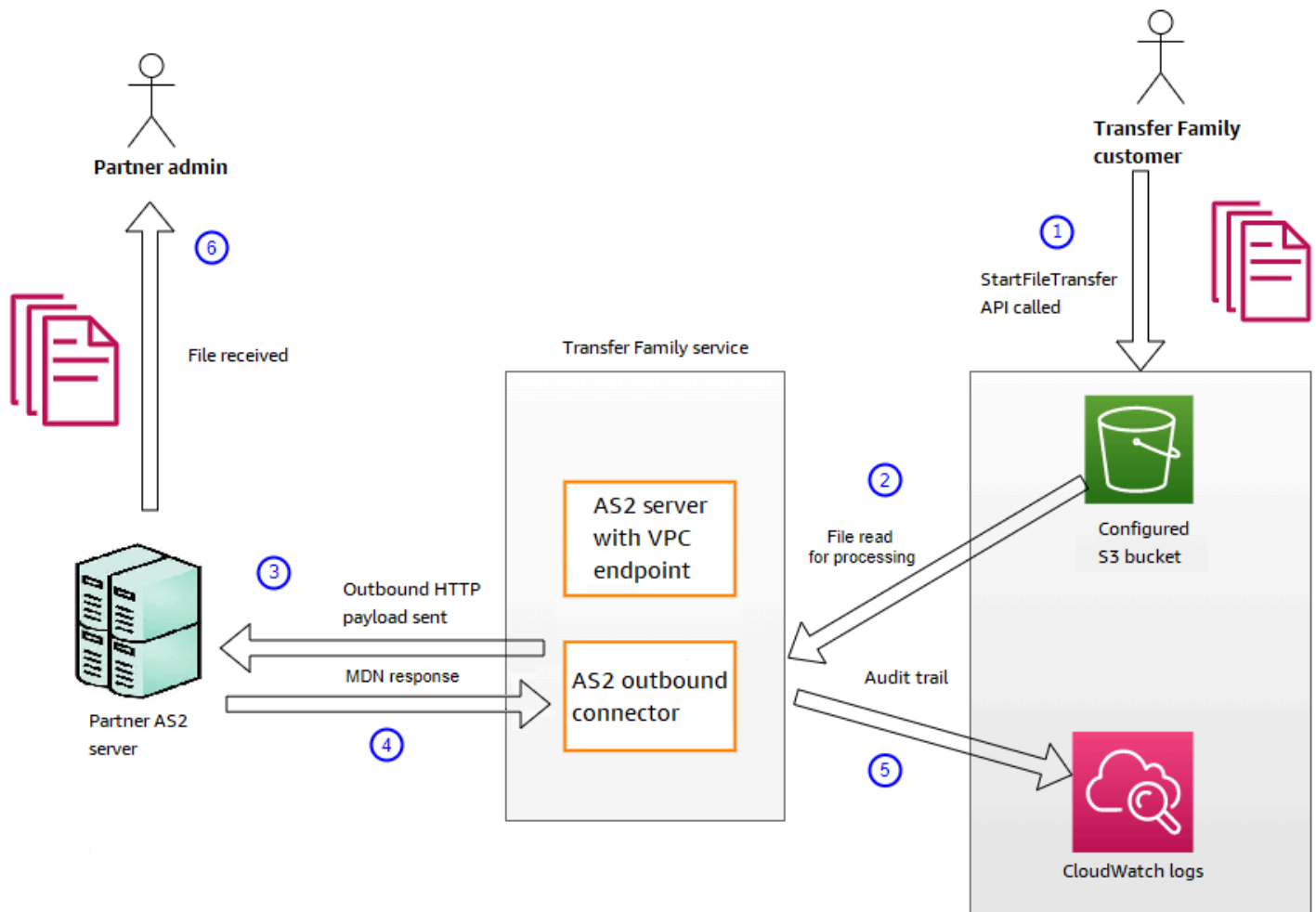
JSON 및 MDN 파일을 보면 파트너가 파일을 성공적으로 수신했는지 확인할 수 있습니다. 이러한 파일 이름은 [파일 이름 및 위치](#)에 설명된 규칙에 따라 지정됩니다. 커넥터를 생성할 때 로깅 역할을 구성한 경우, CloudWatch 로그에서 AS2 메시지 상태를 확인할 수도 있습니다.

AS2 커넥터 세부 정보를 보려면 [AS2 커넥터 세부 정보 보기](#) 섹션을 참조하세요. AS2 생성에 대한 자세한 설명은 [AS2 커넥터 구성](#) 섹션을 참조하세요.

AS2 아웃바운드 메시지를 보내려면

아웃바운드 프로세스는에서 AWS 외부 클라이언트 또는 서비스로 전송되는 메시지 또는 파일로 정의됩니다. 아웃바운드 메시지의 순서는 다음과 같습니다.

1. 관리자가 `start-file-transfer` AWS Command Line Interface (AWS CLI) 명령 또는 `StartFileTransfer` API 작업을 호출합니다. 이 작업은 `connector` 구성을 참조합니다.
2. Transfer Family는 새 파일 요청을 감지하고 파일을 찾습니다. 파일은 압축, 서명 및 암호화됩니다.
3. 전송 HTTP 클라이언트는 HTTP POST 요청을 수행하여 페이로드를 파트너의 AS2 서버로 전송합니다.
4. 프로세스는 HTTP 응답(동기식 또는 비동기식 MDN)과 인라인으로 서명된 MDN 응답을 반환합니다.
5. 파일이 여러 전송 단계 사이를 이동할 때 프로세스는 MDN 응답 수신 및 처리 세부 정보를 고객에게 전달합니다.
6. 원격 AS2 서버는 복호화되고 검증된 파일을 파트너 관리자가 사용할 수 있도록 합니다.



AS2 처리는 일반적인 사용 사례 및 기존 AS2 지원 서버 구현과의 통합에 중점을 두고 많은 RFC 4130 프로토콜을 지원합니다. 지원되는 구성에 대한 자세한 설명은 [AS2 구성](#) 섹션을 참조하세요.

파일 이름 및 위치

이 섹션에서는 AS2 전송에 대한 파일 이름 지정 규칙에 대해 설명합니다.

인바운드 파일 전송에 대해 다음 사항을 참조하세요.

- 계약서에 기본 디렉터리를 지정합니다. 기본 디렉터리는 접두사(있는 경우)와 결합된 Amazon S3 버킷 이름입니다. 예를 들어 `/amzn-s3-demo-bucket/AS2-folder`입니다.
- 수신 파일이 성공적으로 처리되면 파일(및 해당 JSON 파일)이 `/processed` 폴더에 저장됩니다. 예를 들어 `/amzn-s3-demo-bucket/AS2-folder/processed`입니다.

JSON 파일에는 다음과 같은 필드가 포함되어 있습니다.

- `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
- 들어오는 파일을 제대로 처리할 수 없는 경우 파일(및 해당 JSON 파일)이 /failed 폴더에 저장됩니다. 예를 들어 /amzn-s3-demo-bucket/AS2-folder/failed입니다.
 - 전송된 파일은 processed 폴더에 *original_filename.messageId.original_extension*로 저장됩니다. 즉, 전송 메시지 ID가 파일 이름에 원래 확장명 앞에 추가됩니다.
 - JSON 파일이 생성되고 *original_filename.messageId.original_extension.json*로 저장됩니다. 메시지 ID가 추가될 뿐 아니라 전송된 파일 이름에도 문자열 .json이 추가됩니다.
 - 메시지 처리 알림(MDN) 파일이 생성되어 *original_filename.messageId.original_extension.mdn*로 저장됩니다. 메시지 ID가 추가될 뿐 아니라 전송된 파일 이름에도 문자열 .mdn이 추가됩니다.
 - ExampleFileInS3Payload.dat라는 인바운드 파일이 있는 경우 다음 파일이 생성됩니다.
 - File –
ExampleFileInS3Payload.c4d6b6c7-23ea-4b8c-9ada-0cb811dc8b35@44313c54b0a46a36.

- JSON –

ExampleFileInS3Payload.c4d6b6c7-23ea-4b8c-9ada-0cb811dc8b35@44313c54b0a46a36.

- MDN –

ExampleFileInS3Payload.c4d6b6c7-23ea-4b8c-9ada-0cb811dc8b35@44313c54b0a46a36.

아웃바운드 전송의 경우 받는 메시지 파일이 없다는 점을 제외하면 이름이 비슷하며 전송된 메시지의 전송 ID가 파일 이름에 추가된다는 차이점이 있습니다. 전송 ID는 StartFileTransfer API 작업(또는 다른 프로세스나 스크립트가 이 작업을 호출할 때)에 의해 반환됩니다.

- transfer-id는 파일 전송과 관련된 식별자입니다. StartFileTransfer 호출에 포함된 모든 요청은 transfer-id를 공유합니다.
- 기본 디렉터리는 소스 파일에 사용하는 경로와 동일합니다. 즉, 기본 디렉터리는 StartFileTransfer API 작업 또는 start-file-transfer AWS CLI 명령에서 지정하는 경로입니다. 예제:

```
aws transfer start-file-transfer --send-file-paths /amzn-s3-demo-bucket/AS2-folder/
file-to-send.txt
```

이 명령을 실행하면 MDN 및 JSON 파일이 /amzn-s3-demo-bucket/AS2-folder/processed(전송 성공 시) 또는 /amzn-s3-demo-bucket/AS2-folder/failed(전송 실패 시)에 저장됩니다.

- JSON 파일이 생성되고 *original_filename.transferId.messageId.original_extension.json*로 저장됩니다.
- MDN 파일이 생성되고 *original_filename.transferId.messageId.original_extension.mdn*로 저장됩니다.
- ExampleFileOutTestOutboundSyncMdn.dat라는 아웃바운드 파일이 있는 경우 다음 파일이 생성됩니다:
 - JSON – ExampleFileOutTestOutboundSyncMdn.dedf4601-4e90-4043-b16b-579af35e0d83.fbe18db8-7361-42ff-8ab6-49ec1e435f34@c9c705f0baaaabaa.dat.j
 - MDN – ExampleFileOutTestOutboundSyncMdn.dedf4601-4e90-4043-b16b-579af35e0d83.fbe18db8-7361-42ff-8ab6-49ec1e435f34@c9c705f0baaaabaa.dat.m

또한 CloudWatch 로그를 확인하여 실패한 전송을 포함한 전송의 세부 정보를 볼 수 있습니다.

상태 코드

다음 표에는 사용자 또는 파트너가 AS2 메시지를 보낼 때 CloudWatch 로그에 로깅할 수 있는 모든 상태 코드가 나열되어 있습니다. 다양한 메시지 처리 단계는 다양한 메시지 유형에 적용되며 모니터링 전용입니다. COMPLETED 및 FAILED 상태는 처리의 마지막 단계를 나타내며 JSON 파일에 표시됩니다.

코드	설명	처리가 완료되었나요?
처리	메시지가 최종 형식으로 변환되는 중입니다. 예를 들어 압축 해제 및 복호화 단계 모두 이 상태입니다.	아니요
MDN_TRANSMIT	메시지 처리가 MDN 응답을 보내고 있습니다.	아니요
MDN_RECEIVE	메시지 처리가 MDN 응답을 수신하고 있습니다.	아니요
COMPLETED	메시지 처리가 성공적으로 완료되었습니다. 이 상태에는 인바운드 메시지 또는 아웃바운드 메시지의 MDN 확인을 위해 MDN이 전송되는 경우가 포함됩니다.	예
FAILED	메시지 처리에 실패했습니다. 오류 코드 목록은 섹션을 참조하세요 AS2 오류 코드 .	예

샘플 JSON 파일

이 섹션에는 성공적인 전송과 실패한 전송을 위한 샘플 파일을 포함하여 인바운드 및 아웃바운드 전송에 대한 샘플 JSON 파일이 나열되어 있습니다.

성공적으로 전송된 샘플 아웃바운드 파일:

```
{
```

```

"requester-content-type": "application/octet-stream",
"message-subject": "File xyzTest from MyCompany_OID to partner YourCompany",
"requester-file-name": "TestOutboundSyncMdn-9lmCr79hV.dat",
"as2-from": "MyCompany_OID",
"connector-id": "c-c21c63ceaaf34d99b",
"status-code": "COMPLETED",
"disposition": "automatic-action/MDN-sent-automatically; processed",
"transfer-size": 3198,
"mdn-message-id": "OPENAS2-11072022063009+0000-df865189-1450-435b-9b8d-
d8bc0cee97fd@PartnerA_OID_MyCompany_OID",
"mdn-subject": "Message be18db8-7361-42ff-8ab6-49ec1e435f34@c9c705f0baaaabaa has been
accepted",
"as2-to": "PartnerA_OID",
"transfer-id": "dedf4601-4e90-4043-b16b-579af35e0d83",
"file-path": "/amzn-s3-demo-bucket/as2testcell10000/openAs2/
TestOutboundSyncMdn-9lmCr79hV.dat",
"as2-message-id": "fbe18db8-7361-42ff-8ab6-49ec1e435f34@c9c705f0baaaabaa",
"timestamp": "2022-07-11T06:30:10.791274Z"
}

```

전송에 실패한 샘플 아웃바운드 파일:

```

{
  "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": "/amzn-s3-demo-bucket-4c1tq6ohjt9y/as2IntegCell10002/openAs2/
openAs2TestOutboundWrongAs2Ids-necco-3VYn5n8wE.dat",
  "timestamp": "2022-07-11T21:17:24.802378Z"
}

```

성공적으로 전송된 샘플 인바운드 파일:

```

{

```

```

"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_0ID",
"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_0ID_PartnerA_0ID",
"as2-to": "PartnerA_0ID",
"agreement-id": "a-f5c5cbea5f7741988",
"file-path": "processed/openAs2TestInboundAsyncMdn-
necco-5Ab6bTfC0.OPENAS2-11072022233606+0000-5dab0452-0ca1-4f9b-b622-
fba84effff3c@MyCompany_0ID_PartnerA_0ID.dat",
"server-id": "s-5f7422b04c2447ef9",
"timestamp": "2022-07-11T23:36:36.105030Z"
}

```

전송에 실패한 샘플 인바운드 파일:

```

{
  "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"
}

```

AS2 메시지에 대한 HTTP 헤더 사용자 지정

AS2 메시지를 거래 파트너에게 전송할 때 특정 요구 사항을 충족하거나 파트너의 AS2 서버 구성과의 호환성을 향상하도록 HTTP 헤더를 사용자 지정해야 할 수 있습니다. 이 CloudFormation 템플릿은 전송되는 AS2 메시지에 대해 사용자 지정 HTTP 헤더를 활성화하는 인프라를 생성합니다 AWS Transfer Family. Amazon API Gateway 및 Lambda 함수가 프록시 역할을 하도록 설정하여 거래 파트너의 AS2 서버에 필요한 헤더를 동적으로 수정할 수 있습니다.

이 템플릿을 사용하여 다음을 수행합니다.

- 아웃바운드 AS2 메시지에 사용자 지정 HTTP 헤더 추가
- 기본 헤더 값을 사용자 지정 값으로 재정의

Important

기본 헤더 값을 재정의할 때는 전송 실패를 일으킬 수 있으므로 주의해야 합니다. 일부 AS2 헤더는 필수입니다.

- 특정 헤더 요구 사항이 있는 거래 파트너와의 호환성 보장

템플릿 개요

템플릿은 다음과 같은 주요 구성 요소를 생성합니다.

- AS2 메시지를 처리하고 전달하는 Lambda 함수
- Lambda 함수를 노출하는 Amazon API Gateway
- Lambda 함수에 대한 IAM 역할 및 권한
- HTTPS 지원을 위한 조건부 리소스

템플릿 파일은 [동적 HTTP 헤더 템플릿에서 사용할 수 있습니다](#).

작동 방식

1. Amazon API Gateway는 수신 AS2 메시지를 수신합니다 AWS Transfer Family.
2. 요청은 Lambda 함수로 전달됩니다.
3. Lambda 함수는 요청을 처리하여 필요에 따라 헤더를 추가하거나 수정합니다.
4. 그런 다음 수정된 요청이 파트너의 AS2 서버로 전달됩니다.

5. 파트너 서버의 응답은 Lambda 및 Amazon API Gateway를 통해 반환됩니다 AWS Transfer Family.

주요 기능

- 동적 헤더 수정: 제목 헤더를 사용자 지정하고 기타 필수 헤더를 추가할 수 있습니다.
- 프로토콜 지원: HTTP 및 HTTPS 프로토콜 모두에서 작동합니다.
- 유연한 구성: 파트너 호스트, 포트 및 경로의 사양을 허용합니다.

구현 세부 정보

템플릿은 다음과 같은 주요 구성 요소를 구현합니다.

Lambda 함수

솔루션의 핵심은 다음과 같은 Node.js Lambda 함수입니다.

- Amazon API Gateway에서 요청을 수신합니다.
- 구성 및 수신 요청 데이터를 기반으로 헤더를 수정합니다.
- 수정된 요청을 파트너의 AS2 서버로 전달합니다.
- HTTP 및 HTTPS 프로토콜 모두 처리
- 오류 처리 및 로깅 포함

Amazon API Gateway

HTTP API는 다음과 같이 설정됩니다.

- 수신 AS2 메시지 수신
- Lambda 함수로 요청 라우팅
- 에 응답 반환 AWS Transfer Family

템플릿 파라미터

다음과 같이 템플릿 파라미터에 대한 정보를 입력합니다. 이러한 모든 파라미터는 문자열입니다.

- Environment: 이 파라미터는 템플릿이 생성하는 리소스의 이름을 지정하는 데 사용됩니다. 개발 또는 프로덕션 환경을 위한 것인지 여부입니다. 유효한 값은 dev 및 prod입니다.

- PartnerHost: AS2 파트너 서버의 IP 주소 또는 호스트 이름입니다.
- PartnerPort: AS2 파트너 서버의 포트 번호입니다. 지정하지 않으면 기본값은 HTTP의 경우 80, HTTPS의 경우 443입니다.
- PartnerPath: 파트너 서버의 AS2 엔드포인트 경로
- ProtocolType: AS2 통신에 사용할 프로토콜: 유효한 값은 HTTP 및 HTTPS입니다.

조건부 리소스

HTTPS 지원을 위해 템플릿은 조건부로 다음을 생성합니다.

- CA 인증서용 Lambda 계층
- Lambda 함수의 HTTPS별 구성

배포 및 사용

CloudFormation 템플릿을 사용하여 AS2 HTTP 헤더를 사용자 지정하려면

1. <https://console.aws.amazon.com/cloudformation> CloudFormation 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 스택을 선택합니다.
3. 스택 생성을 선택한 다음 새 리소스 사용(표준)을 선택합니다.
4. 사전 조건 - 템플릿 준비 섹션에서 기존 템플릿 선택을 선택합니다.
5. 이 링크인 [동적 HTTP 헤더 템플릿을](#) 복사하여 Amazon S3 URL 필드에 붙여 넣습니다.
6. 다음을 선택합니다.
7. 파라미터 세부 정보를 정보로 입력합니다. 이는에 자세히 설명되어 있습니다 [템플릿 파라미터](#).
8. 다음을 선택합니다. 스택 옵션 구성페이지에서 다음을 다시 선택합니다.
9. 생성 중인 스택의 세부 정보를 검토한 다음 스택 생성을 선택합니다.

Note

페이지 하단의 기능에서가 AWS Identity and Access Management (IAM) 리소스를 생성할 CloudFormation 수 있음을 확인해야 합니다.

이 CloudFormation 스택을 배포한 후:

1. 스택 출력에 제공된 Amazon API Gateway 엔드포인트 URL을 기록해 둡니다.
2. 이 새 Amazon API Gateway 엔드포인트를 사용하도록 기존 AWS Transfer Family 커넥터를 업데이트합니다.
3. 이제 솔루션은 구성된 대로 헤더를 추가하거나 수정하여 AS2 메시지를 처리합니다.

Warning

제목 헤더만 수정하거나 파트너가 명시적으로 예상하는 헤더만 추가합니다. 다른 헤더를 변경하면 전송 실패가 발생할 수 있습니다.

AS2 사용량 모니터링

Amazon CloudWatch 및를 사용하여 AS2 활동을 모니터링할 수 있습니다 AWS CloudTrail. 다른 Transfer Family 서버 지표(metrics)를 보려면 [AWS Transfer Family 서버에 대한 Amazon CloudWatch 로깅](#)을 참조하세요.

AS2 지표

지표	설명
InboundMessage	거래 파트너로부터 성공적으로 수신한 AS2 메시지의 총 개수. 단위: 개 기간: 5분
InboundFailedMessage	거래 파트너로부터 성공적으로 수신되지 못한 AS2 메시지의 총 개수. 즉, 거래 파트너가 메시지를 보냈지만 Transfer Family 서버가 메시지를 성공적으로 처리하지 못했습니다. 단위: 개 기간: 5분
OutboundMessage	Transfer Family 서버에서 거래 파트너로 성공적으로 보낸 AS2 메시지의 총 개수입니다.

지표	설명
	단위: 개 기간 = 5분
OutboundFailedMessage	거래 파트너에게 성공적으로 전송되지 못한 AS2 메시지의 총 개수. 즉, Transfer Family 서버에서 전송되었지만 거래 파트너가 성공적으로 수신하지 못했습니다. 단위: 개 기간: 5분
DaysUntilExpiry	가져올 때 인증서에 InactiveDate 설정된에 따라 인증서가 만료될 때까지의 일수입니다. 단위: 개 차원: CertificateId , Description (제공된 경우) 기간: 1일 자세한 내용은 AS2 인증서 교체 단원을 참조하십시오.

AS2 상태 코드

다음 표에는 사용자 또는 파트너가 AS2 메시지를 보낼 때 CloudWatch 로그에 로깅할 수 있는 모든 상태 코드가 나열되어 있습니다. 다양한 메시지 처리 단계는 다양한 메시지 유형에 적용되며 모니터링 전용입니다. COMPLETED 및 FAILED 상태는 처리의 마지막 단계를 나타내며 JSON 파일에 표시됩니다.

코드	설명	처리가 완료되었나요?
처리	메시지가 최종 형식으로 변환되는 중입니다. 예를 들어 압축	아니요

코드	설명	처리가 완료되었나요?
	해제 및 복호화 단계 모두가 상태입니다.	
MDN_TRANSMIT	메시지 처리가 MDN 응답을 보내고 있습니다.	아니요
MDN_RECEIVE	메시지 처리가 MDN 응답을 수신하고 있습니다.	아니요
COMPLETED	메시지 처리가 성공적으로 완료되었습니다. 이 상태에는 인바운드 메시지 또는 아웃바운드 메시지의 MDN 확인을 위해 MDN이 전송되는 경우가 포함됩니다.	예
FAILED	메시지 처리에 실패했습니다. 오류 코드 목록은 섹션을 참조하세요 AS2 오류 코드 .	예

AS2 오류 코드

다음 표에서는 AS2 파일 전송에서 수신할 수 있는 오류 코드를 나열하고 설명합니다.

AS2 오류 코드

코드	오류	설명 및 해상도
ACCESS_DENIED	- 액세스가 거부되었습니다. 액세스 역할에 필요한 권한이 있는지 확인하세요. - 파일 경로: ## 파일 경로가 잘못되었습니다. - ErrorCode로 자격 증명을 가져오지 못했습니다. ## ##	유효하지 않거나 형식이 StartFileTransfer 잘못된 요청을 처리할 때 발생합니다SendFilePaths . 즉, 경로에 Amazon S3 버킷 이름이 없거나 경로에 유효하지 않은 문자가 포함되어 있습니다. Transfer Family가 액세스 역할

코드	오류	설명 및 해상도
		또는 로깅 역할을 맡지 못하는 경우에도 발생합니다. 경로에 유효한 Amazon S3 버킷과 키 이름이 포함되어 있는지 확인하세요.
AGREEMENT_NOT_FOUND	계약을 찾을 수 없습니다.	계약을 찾을 수 없거나 계약이 비활성 프로필과 연결되어 있습니다. Transfer Family 서버 내에서 계약을 업데이트하여 활성 프로필을 포함하세요.
CONNECTOR_NOT_FOUND	커넥터 또는 관련 구성을 찾을 수 없습니다.	커넥터를 찾을 수 없거나 커넥터가 비활성 프로필과 연결되어 있습니다. 활성 프로필을 포함하도록 커넥터를 업데이트하세요.

코드	오류	설명 및 해상도
CREDENTIALS_RETRIEVAL_FAILED	1. Secrets Manager에서 암호를 찾을 수 없습니다. 2. Secrets Manager에 액세스할 수 없습니다 3. Secrets Manager에서 암호해독 실패 4. 제한으로 인해 암호 값을 가져올 수 없습니다.	AS2 Basic 인증의 경우 암호를 올바르게 형식화해야 합니다. 다음 해결 방법은 이전 열에 나열된 오류에 해당합니다. 1. 암호 ID가 정확한지 확인하세요. 2. 액세스 역할에 암호를 읽을 수 있는 적절한 권한이 있는지 확인하세요. 접근 역할은 StartFileTransfer 요청에서 사용된 파일 위치의 상위 디렉터리에 대한 읽기 및 쓰기 액세스를 제공해야 합니다. 또한 그 역할이 StartFileTransfer 와 함께 전송하려는 파일의 상위 디렉터리에 대한 읽기 및 쓰기 액세스 권한을 제공하는지 확인하세요. 3. 고객 관리형 키가 보안 암호에 사용되는 경우 액세스 역할에 AWS Key Management Service (AWS KMS) 키에 대한 권한이 있는지 확인합니다. 4. 해당 할당량은 비밀 처리를 위한 할당량을 참조하세요.

코드	오류	설명 및 해상도
DECOMPRESSION_FAILED	메시지 압축을 풀지 못했습니다.	전송된 파일이 손상되었거나 압축 알고리즘이 유효하지 않습니다. 메시지를 다시 전송하고 ZLIB 압축이 사용되는지 확인하거나 압축을 사용하지 않고 메시지를 다시 보내세요.
DECRYPT_FAILED	### ID 메시지를 해독하지 못했습니다. 파트너가 올바른 공개 암호화 키를 가지고 있는지 확인하세요.	복호화에 실패했습니다. 파트너가 유효한 인증서를 사용하여 페이로드를 전송했고 암호화는 유효한 암호화 알고리즘을 사용하여 수행되었는지 확인합니다.
DECRYPT_FAILED_INVALID_SMIME_FORMAT	엔벨로핑된 MimePart를 파싱할 수 없습니다.	MIME 페이로드가 손상되었거나 지원되지 않는 SMIME 형식입니다. 발신자는 사용 중인 형식이 지원되는지 확인한 다음 페이로드를 다시 전송해야 합니다.
DECRYPT_FAILED_NO_DECRYPTION_KEY_FOUND	일치하는 암호 해독 키를 찾을 수 없습니다.	파트너 프로필에 메시지와 일치하는 인증서가 할당되지 않았거나 메시지와 일치하는 인증서가 현재 만료되었거나 더 이상 유효하지 않습니다. 파트너 프로필을 업데이트하고 유효한 인증서가 포함되어 있는지 확인해야 합니다.

코드	오류	설명 및 해상도
DECRYPT_FAILED_UNSUPPORTED_ENCRYPTION_ALG	지원되지 않는 알고리즘(ID: <i>encryption-ID</i>)을 사용하여 SMIME 페이로드 암호 해독을 요청했습니다.	원격 발신자가 지원되지 않는 암호화 알고리즘이 포함된 AS2 페이로드를 전송했습니다. 발신자는 AWS Transfer Family에서 지원하는 암호화 알고리즘을 선택해야 합니다.
DUPLICATE_MESSAGE	중복 또는 이중 처리된 단계.	페이로드에 중복 처리 단계가 있습니다. 예를 들어 2개의 암호화 단계가 있습니다. 서명, 압축, 암호화를 한 번에 완료하여 메시지를 재전송합니다.
ENCRYPT_FAILED_NO_ENCRYPTION_KEY_FOUND	프로필에서 유효한 공개 암호화 인증서를 찾을 수 없음: <i>local-profile-ID</i>	Transfer Family가 아웃바운드 메시지를 암호화하려고 시도하지만 로컬 프로필에 대한 암호화 인증서를 찾을 수 없습니다. 해결 옵션: - 로컬 프로필에 암호화를 위한 인증서와 프라이빗 키가 첨부되어 있는지 확인하세요. - 암호화 인증서가 현재 활성 상태인지 확인하세요.

코드	오류	설명 및 해상도
ENCRYPTION_FAILED	## ## 파일을 암호화하지 못했습니다.	전송할 파일은 암호화에 사용할 수 없습니다. 파일이 예상 AS2 위치에 있고 AWS Transfer Family 가 파일을 읽을 권한이 있는지 확인하세요.
FILE_SIZE_TOO_LARGE	파일 크기가 너무 큼니다.	이는 파일 크기 제한을 초과하는 파일을 보내거나 받을 때 발생합니다.
HTTP_ERROR_RESPONSE_FROM_PARTNER	### URL ID= <i>message-ID</i> 인 메시지에 대해 상태 400을 반환했습니다.	파트너의 AS2 서버와 통신할 때 예상치 못한 HTTP 응답 코드가 반환되었습니다. 파트너는 AS2 서버 로그에서 추가 진단을 제공할 수 있습니다.
INSUFFICIENT_MESSAGE_SECURITY_UNENCRYPTED	암호화가 필요합니다.	파트너가 Transfer Family에 암호화되지 않은 메시지를 보냈지만 이 메시지는 지원되지 않습니다. 발신자는 암호화된 페이로드를 사용해야 합니다.
INVALID_ENDPOINT_PROTOCOL	HTTP와 HTTPS만 지원됩니다.	AS2 커넥터 구성에서 HTTP 또는 HTTPS를 프로토콜로 지정해야 합니다.

코드	오류	설명 및 해상도
INVALID_REQUEST	1. 메시지 헤더에 문제가 있습니다. 2. 보안 JSON을 구문 분석할 수 없습니다. 시크릿 JSON이 예상 형식과 일치하지 않았습니다. 3. 암호는 JSON 문자열이어야 합니다. 4. 사용자 이름은 콜론을 포함할 수 없습니다. 사용자 이름은 제어 문자를 포함할 수 없습니다. 사용자 이름은 ASCII 문자만 포함해야 합니다. 암호는 제어 문자를 포함할 수 없습니다. 암호는 ASCII 문자만 포함해야 합니다.	이 오류에는 여러 가지 원인이 있습니다. 다음 해결 방법은 이전 열에 나열된 오류에 해당합니다. 1. as2-from 및 as2-to 필드를 확인합니다. 원본 메시지 ID가 MDN 형식에 맞는지 확인하세요. 또한 메시지 ID 형식에 AS2 헤더가 누락되지 않았는지 확인하세요. 2. AS2 커넥터에 대한 기본 인증을 활성화합니다.에 설명된 대로 암호 값이 문서화된 형식과 일치하는지 확인하세요. 3. 암호는 바이너리가 아닌 문자열로 제공되어야 합니다. 4. 사용자 이름 또는 암호를 필요에 따라 수정하세요.
INVALID_URL_FORMAT	잘못된 URL 형식: <i>URL</i>	잘못된 URL로 구성된 커넥터를 사용하여 아웃바운드 메시지를 보낼 때 이 문제가 발생합니다. 커넥터가 유효한 HTTP 또는 HTTPS URL로 구성되어 있는지 확인하세요.

코드	오류	설명 및 해상도
MDN_RESPONSE_INDICATES_AUTHENTICATION_FAILED	해당 사항 없음	수신자는 발신자를 인증할 수 없습니다. 거래 파트너가 처리 수정자 오류: authentication-failed와 함께 Transfer Family에 MDN을 반환합니다.
MDN_RESPONSE_INDICATES_DECOMPRESSION_FAILED	해당 사항 없음	이는 수신자가 메시지 내용을 압축 해제할 수 없을 때 발생합니다. 거래 파트너가 처리 수정자 오류: decompression-failed와 함께 Transfer Family에 MDN을 반환합니다.
MDN_RESPONSE_INDICATES_DECRYPTION_FAILED	해당 사항 없음	수신자는 메시지 내용을 해독할 수 없습니다. 거래 파트너가 처리 수정자 오류: authentication-failed와 함께 Transfer Family에 MDN을 반환합니다.
MDN_RESPONSE_INDICATES_INSUFFICIENT_MESSAGE_SECURITY	해당 사항 없음	수신자는 메시지가 서명되거나 암호화되기를 기대하지만, 그렇지 않습니다. 거래 파트너가 처리 수정자 오류: insufficient-message-security와 함께 Transfer Family에 MDN을 반환합니다. 커넥터에서 서명 및/또는 암호화를 활성화하여 거래 파트너의 기대에 부합하도록 하세요.
MDN_RESPONSE_INDICATES_INTEGRITY_CHECK_FAILED	해당 사항 없음	수신자는 콘텐츠 무결성을 확인할 수 없습니다. 거래 파트너가 처리 수정자 오류: integrity-check-failed와 함께 Transfer Family에 MDN을 반환합니다.

코드	오류	설명 및 해상도
PATH_NOT_FOUND	디렉터리 <i>file-path</i> 를 생성할 수 없습니다. 상위 경로를 찾을 수 없습니다.	Transfer Family가 고객의 Amazon S3 버킷에 디렉터를 생성하려고 시도했지만 버킷을 찾을 수 없습니다. StartFileTransfer 명령에 언급된 각 경로에 기존 버킷의 이름이 포함되어 있는지 확인하세요.
SEND_FILE_NOT_FOUND	파일 경로 <i>## ##</i> 를 찾을 수 없습니다.	Transfer Family는 파일 보내기 작업에서 파일을 찾을 수 없습니다. 구성된 홈 디렉토리 및 경로가 유효하고 Transfer Family에 파일에 대한 읽기 권한이 있는지 확인합니다.
SERVER_NOT_FOUND	이벤트와 연결된 서버를 찾을 수 없습니다.	Transfer Family가 메시지를 받았을 때 서버를 찾지 못했습니다. 수신 메시지를 처리하는 중에 서버가 삭제되면 이런 일이 발생할 수 있습니다.
SERVER_NOT_ONLINE	서버 <i>server-ID</i> 가 온라인 상태가 아닙니다.	Transfer Family 서버가 오프라인 상태입니다. 메시지를 수신하고 처리할 수 있도록 서버를 시작합니다.

코드	오류	설명 및 해상도
SIGNING_FAILED	파일에 서명하지 못했습니다.	전송할 파일을 서명할 수 없거나 서명을 수행할 수 없습니다. 파일이 예상 AS2 위치에 있고 AWS Transfer Family 가 파일을 읽을 권한이 있는지 확인하세요.
SIGNING_FAILED_NO_SIGNING_KEY_FOUND	프로필: ## 프로필 ID에 대한 인증서를 찾을 수 없습니다.	아웃바운드 메시지에 서명하려고 시도했지만 로컬 프로필에 대한 서명 인증서를 찾을 수 없습니다. 해결 옵션: - 로컬 프로필에 서명을 위한 인증서와 프라이빗 키가 첨부되어 있는지 확인하세요. - 서명 인증서가 현재 활성화 상태인지 확인하세요.
UNABLE_RESOLVE_HOST_TO_IP_ADDRESS	호스트 이름을 IP 주소로 확인할 수 없습니다.	Transfer Family는 AS2 커넥터에 구성된 공용 DNS 서버에서 DNS 대 IP 주소 확인을 수행할 수 없습니다. 커넥터가 유효한 파트너 URL을 가리키도록 업데이트하세요.

코드	오류	설명 및 해상도
UNABLE_TO_CONNECT_TO_REMOTE_HOST_OR_IP	엔드포인트에 대한 연결이 시간 초과되었습니다.	Transfer Family는 구성된 파트너의 AS2 서버에 소켓 연결을 설정할 수 없습니다. 구성된 IP 주소에서 파트너의 AS2 서버를 사용할 수 있는지 확인하세요.
UNABLE_TO_RESOLVE_HOSTNAME	### ##의 호스트 이름을 확인할 수 없습니다.	Transfer Family 서버가 공용 DNS 서버를 사용하여 파트너의 호스트 이름을 확인할 수 없습니다. 구성된 호스트가 등록되어 있고 DNS 레코드를 게시할 시간이 있었는지 확인하세요.
VERIFICATION_FAILED	### ID AS2 메시지에 대한 서명 확인에 실패했거나 MIC 코드가 일치하지 않았습니다.	발신자의 서명 인증서가 원격 프로필의 서명 인증서와 일치하는지 확인하세요. 또한 MIC 알고리즘이 호환되는지 확인합니다 AWS Transfer Family.

코드	오류	설명 및 해상도
VERIFICATION_FAILED_NO_MATCHING_KEY_FOUND	- 프로필: ### ### ID 메시지 서명과 일치하는 공개 인증서를 찾을 수 없습니다. - 존재하지 않는 프로필 (partner-profile-ID)에 대한 인증서를 얻을 수 없습니다. - 프로필(partner-profile-ID)에서 유효한 인증서를 찾을 수 없습니다.	AWS Transfer Family가 수신된 메시지의 서명을 확인하려고 하지만 파트너 프로필에 일치하는 서명 인증서를 찾을 수 없습니다. 해결 옵션: - 파트너 프로필에 서명을 위한 인증서가 첨부되어 있는지 확인하세요. - 인증서가 현재 활성 상태인지 확인하세요. - 인증서가 파트너를 위한 정확한 서명 인증서인지 확인하세요.

인증서 만료 모니터링

AWS Transfer Family는 AS2 인증서의 만료 날짜를 자동으로 모니터링하고 Amazon CloudWatch 지표를 게시하여 인증서 만료 시기를 추적할 수 있습니다. 이를 통해 인증서 갱신을 사전에 관리하고 서비스 중단을 방지할 수 있습니다.

DaysUntilExpiry 지표

AS2용 인증서를 가져오면 Transfer Family는 라는 CloudWatch 지표를 자동으로 생성합니다 DaysUntilExpiry. 이 지표는 인증서를 가져올 때 InactiveDate 지정함에 따라 인증서가 만료될 때까지 남은 일수를 추적합니다.

지표 세부 정보:

- 지표 이름: DaysUntilExpiry
- 네임스페이스: AWS/Transfer
- 차원: CertificateId (항상 존재), Description (인증서 가져오기 중에 제공된 경우)
- 단위: 개수(일)

- 빈도: 매일 게시됨

⚠ Important

Transfer Family에서 이 지표를 계정에 내보내려면 인증서를 가져온 후 최대 하루가 걸릴 수 있습니다.

인증서가 비활성 날짜에 가까워지면 지표 값이 매일 1씩 감소합니다. 예를 들어 인증서에 만료까지 30일이 있는 경우 지표에는 30일, 다음 날 29일이 표시됩니다.

인증서 모니터링 모범 사례

인증서 만료 모니터링을 설정할 때 다음 모범 사례를 따르세요.

- 여러 알림 임계값 설정: 다양한 기간(예: 만료 전 30일, 14일, 7일)에 대한 경보를 생성하여 인증서 갱신에 충분한 시간을 제공합니다.
- 적절한 통계 사용: 경보를 생성할 때 Maximum 통계를 사용하여 최신 지표 값을 캡처합니다.
- 적절한 경보 작업 구성: 인증서를 갱신할 수 있는 적절한 팀원에게 알리도록 알림을 설정합니다.
- 알림 테스트: 알림 시스템을 정기적으로 테스트하여 알림이 제대로 전달되는지 확인합니다.
- 프로세스 문서화: 인증서 갱신 프로세스와 다양한 인증서를 담당하는 사람에 대한 문서를 유지 관리합니다.

경보 구성 예

다음은 다양한 알림 시나리오에 대한 경보 구성의 예입니다.

30일 만료 경고

인증서가 만료될 때까지 30일 이하일 때 트리거되는 경보를 생성합니다.

- 지표: DaysUntilExpiry
- 통계: Maximum
- 기간: 1일
- 임계값: 30
- 비교: 임계값 이하

- 누락된 데이터 처리: 누락된 데이터를 양호한 것으로 취급(위반하지 않음)

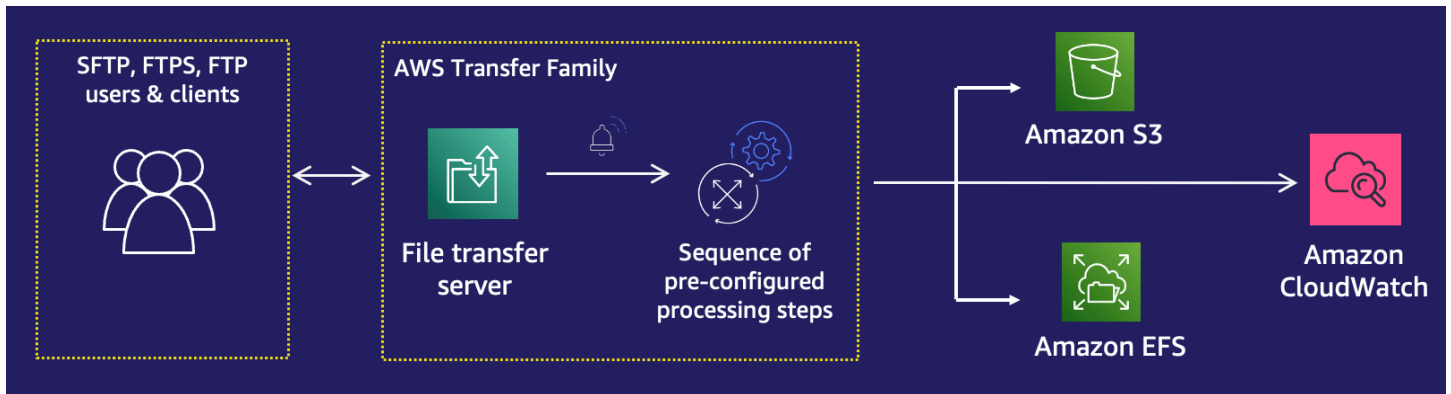
심각한 7일 만료 경고

인증서가 만료될 때까지 7일 이하일 때 트리거되는 중요한 경보를 생성합니다.

- 지표: DaysUntilExpiry
- 통계: Maximum
- 기간: 1일
- 임계값: 7
- 비교: 임계값 이하
- 누락된 데이터 처리: 누락된 데이터를 양호한 것으로 처리(위반하지 않음)

AWS Transfer Family 관리형 워크플로

AWS Transfer Family 는 파일 처리를 위한 관리형 워크플로를 지원합니다. 관리형 워크플로를 사용하면 SFTP, FTPS 또는 FTP를 통해 파일이 전송된 후 워크플로를 시작할 수 있습니다. 이 기능을 사용하면 파일 처리에 필요한 모든 단계를 조정하여 B2B(Business-to-Business) 파일 교환에 대한 규정 준수 요구 사항을 안전하고 비용 효율적으로 충족할 수 있습니다. 뿐만 아니라 엔드 투 엔드 감사 및 가시성의 이점을 활용할 수 있습니다.



관리형 워크플로는 파일 처리 작업을 오케스트레이션하여 다운스트림 애플리케이션에서 데이터를 사용하기 전에 데이터를 사전 처리하는 데 도움이 됩니다. 이러한 파일 처리 작업에는 다음이 포함될 수 있습니다.

- 파일을 사용자별 폴더로 이동.
- 워크플로의 일부로 파일 복호화.
- 파일 태그 지정.
- AWS Lambda 함수를 생성하고 워크플로에 연결하여 사용자 지정 처리를 수행합니다.
- 파일이 성공적으로 전송되면 알림을 보냅니다. (이 사용 사례를 자세히 설명하는 블로그 게시물은 [AWS Transfer Family 관리형 워크플로를 사용하여 파일 전송 알림 사용자 지정](#)을 참조하세요.)

조직 내 여러 사업부에 걸쳐 있는 일반적인 업로드 후 파일 처리 작업을 빠르게 복제하고 표준화하려면 코드형 인프라(IaC)를 사용하여 워크플로를 배포할 수 있습니다. 전체 업로드된 파일에 대해 관리형 워크플로를 시작하도록 지정할 수 있습니다. 또한 세션이 조기에 끊어져 일부만 업로드된 파일에 대해 다른 관리형 워크플로를 시작하도록 지정할 수 있습니다. 기본 제공되는 예외 처리 기능을 통해 파일 처리 결과에 신속하게 대응하는 동시에 오류 처리 방법을 제어할 수 있습니다. 또한 각 워크플로 단계는 세부 로그를 생성하며, 이를 감사하여 데이터 계보를 추적할 수 있습니다.

시작하려면 다음 단계를 수행합니다.

1. 요구 사항에 따라 복사, 태그 지정 및 기타 단계와 같은 사전 처리 작업을 포함하도록 워크플로를 설정합니다. 세부 정보는 [워크플로 만들기](#)를 참조하세요.
2. Transfer Family가 워크플로를 실행하는 데 사용하는 실행 역할을 구성합니다. 세부 정보는 [워크플로에 대한 IAM 정책](#)를 참조하세요.
3. 워크플로를 서버에 매핑하여 파일 도착 시 이 워크플로에 지정된 작업이 실시간으로 평가 및 시작 되도록 합니다. 세부 정보는 [워크플로 구성 및 실행](#)를 참조하세요.

관련 정보

- 워크플로 실행을 모니터링하려면 [Transfer Family 서버에 CloudWatch 지표 사용](#)을 참조하세요.
- 자세한 실행 로그 및 문제 해결 정보는 [Amazon CloudWatch를 사용하여 워크플로 관련 오류를 해결합니다.](#)를 참조하세요.
- Transfer Family는 파일 전송 솔루션 구축을 안내하는 블로그 게시물과 워크숍을 제공합니다. 이 솔루션은 관리형 SFTP/FTPS 엔드포인트 AWS Transfer Family 에를 활용하고 사용자 관리를 위해 Amazon Cognito 및 DynamoDB를 활용합니다.

블로그 게시물은 [AWS Transfer Family 및 Amazon S3에서 자격 증명 공급자로 Amazon Cognito 사용에서 확인할 수 있습니다Amazon S3](#). 여기에서 워크숍에 대한 세부 정보를 볼 수 [있습니다](#).

- Transfer Family 워크플로에 대한 간략한 소개는 [AWS Transfer Family 관리형 워크플로](#)를 보세요.
- 다음 워크숍은 실습을 통해 외부 SFTP 서버와 Amazon S3 간의 파일 전송 및 해당 파일의 일반적인 사전 및 사후 처리와 관련된 완전 자동화된 이벤트 기반 워크플로인 [이벤트 기반 MFT 워크숍](#)을 구축합니다.

이 동영상에서는이 워크숍을 안내합니다.

주제

- [워크플로 만들기](#)
- [사전 정의된 단계 사용](#)
- [사용자 지정 파일 처리 단계 사용](#)
- [워크플로에 대한 IAM 정책](#)
- [워크플로의 예외 처리](#)
- [워크플로 실행 모니터링](#)
- [템플릿에서 워크플로 생성](#)

- [Transfer Family 서버에서 워크플로 제거](#)
- [관리형 워크플로 제한 및 제약 조건](#)

관리형 워크플로를 시작하는 데 도움이 더 필요하다면 다음 리소스를 참조하세요.

- [AWS Transfer Family 관리형 워크플로 데모 비디오](#)
- [AWS Transfer Family 워크플로를 사용하여 클라우드 네이티브 파일 전송 플랫폼 구축](#) 블로그 게시물

워크플로 만들기

이 주제에 설명된 AWS Management Console대로를 사용하여 관리형 워크플로를 생성할 수 있습니다. 워크플로를 가능한 한 쉽게 만들기 위해 콘솔의 대부분의 섹션에서 상황별 도움말 패널을 사용할 수 있습니다.

워크플로에는 다음과 같은 두 가지 단계가 있습니다.

- 공칭 단계 - 공칭 단계는 들어오는 파일에 적용할 파일 처리 단계입니다. 공칭 단계를 두 개 이상 선택한 경우 각 단계는 선행 시퀀스로 처리됩니다.
- 예외 처리 단계 - 예외 핸들러는 공칭 단계가 실패하거나 검증 오류가 발생할 경우 AWS Transfer Family 실행되는 파일 처리 단계입니다.

워크플로 만들기

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 워크플로를 선택합니다.
3. 워크플로 페이지에서 워크플로 생성을 선택합니다.
4. 워크플로 생성 페이지에서 설명을 입력합니다. 이 설명은 워크플로 페이지에 표시됩니다.
5. 공칭 단계 섹션에서 단계 추가를 선택합니다. 단계를 하나 이상 추가합니다.
 - a. 사용 가능한 옵션에서 단계 타입을 선택합니다. 다양한 단계 타입에 대한 자세한 내용은 [the section called “사전 정의된 단계 사용”](#)를 참조하세요.
 - b. 다음을 선택한 다음 해당 단계에 대한 파라미터를 구성합니다.
 - c. 다음을 선택한 다음 해당 단계에 대한 세부 정보를 검토합니다.
 - d. 단계 생성을 선택하여 단계를 추가하고 계속합니다.

- e. 필요에 따라 단계를 계속 추가합니다. 워크플로의 최대 단계 수는 8개입니다.
- f. 필요한 공칭 단계를 모두 추가한 후 예외 처리기 — 옵션 섹션으로 아래로 스크롤하여 단계 추가를 선택합니다.

Note

실패 알림을 실시간으로 받을 수 있도록 워크플로가 실패할 때 실행할 예외 처리기와 단계를 설정하는 것이 좋습니다.

- 6. 예외 처리기를 구성하려면 앞에서 설명한 것과 같은 방식으로 단계를 추가하세요. 파일로 인해 특정 단계에서 예외가 발생하는 경우 예외 처리기가 하나씩 간접적으로 호출됩니다.
- 7. (옵션) 태그 섹션으로 스크롤하여 워크플로에 사용할 태그를 추가합니다.
- 8. 플릿 구성을 살펴본 후 워크플로 생성을 선택합니다.

Important

워크플로를 만든 후에는 편집할 수 없으므로 구성을 주의 깊게 검토해야 합니다.

워크플로 구성 및 실행

워크플로를 실행하려면 먼저 Transfer Family 서버에 연결해야 합니다.

업로드된 파일에서 워크플로를 실행하도록 Transfer Family를 구성하려면

- 1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
- 2. 왼쪽 탐색 창에서 서버를 선택합니다.
 - 기존 서버에 워크플로를 추가하려면 워크플로에 사용할 서버를 선택합니다.
 - 또는 새 서버를 만들고 이 서버에 워크플로를 추가할 수도 있습니다. 자세한 내용은 [SFTP, FTPS 또는 FTP 서버 엔드포인트 구성](#) 단원을 참조하십시오.
- 3. 서버의 세부 정보 페이지에서 추가 세부 정보 섹션까지 아래로 스크롤한 다음 편집을 선택합니다.

Note

기본적으로 서버에는 연결된 워크플로가 없습니다. 추가 세부 정보 섹션을 사용하여 워크플로를 선택한 서버에 연결할 수 있습니다.

4. 추가 세부 정보 편집 페이지의 관리형 워크플로 섹션에서 모든 업로드에서 실행할 워크플로를 선택합니다.

Note

워크플로가 아직 없는 경우 새 워크플로 생성을 선택하여 워크플로를 생성합니다.

- a. 사용할 워크플로 ID를 선택합니다.
- b. 실행 역할을 선택합니다. 이 역할은 워크플로의 단계를 실행할 때 Transfer Family가 맡는 역할입니다. 자세한 내용은 [워크플로에 대한 IAM 정책](#)를 참조하세요. 저장을 선택합니다.

Note

워크플로를 더 이상 서버와 연결하지 않으려면 연결을 제거하면 됩니다. 자세한 내용은 [Transfer Family 서버에서 워크플로 제거](#)를 참조하세요.

워크플로를 실행하려면

워크플로를 실행하려면 관련 워크플로로 구성된 Transfer Family 서버에 파일을 업로드합니다.

Note

서버에서 워크플로를 제거하고 새 워크플로로 바꾸거나 워크플로의 실행 역할에 영향을 미치는 서버 구성을 업데이트할 때마다 새 워크플로를 실행하기 전에 약 10분을 기다려야 합니다. Transfer Family 서버는 워크플로 세부 정보를 캐시하며 서버가 캐시를 새로 고치는 데 10분이 걸립니다.

또한 활성 SFTP 세션에서 로그아웃한 다음 10분 대기 시간이 지난 후 다시 로그인해야 변경 사항을 확인할 수 있습니다.

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 /amzn-s3-demo-bucket/home/users/bob/doc1.pdf
doc1.pdf                                     100% 5013KB
  601.0KB/s   00:08
sftp> exit
>
```

파일이 업로드되고 나면 파일에 정의된 작업이 수행됩니다. 예를 들어 워크플로에 복사 단계가 포함된 경우 해당 단계에서 정의한 위치에 파일이 복사됩니다. Amazon CloudWatch Logs를 사용하여 실행된 단계와 실행 상태를 추적할 수 있습니다.

워크플로 세부 정보 보기

이전에 만든 워크플로 또는 워크플로 실행에 대한 세부 정보를 볼 수 있습니다. 이러한 세부 정보를 보려면 콘솔 또는 AWS Command Line Interface ()를 사용할 수 있습니다AWS CLI.

Console**워크플로 세부 정보 보기**

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 워크플로를 선택합니다.
3. 워크플로 페이지에서 워크플로를 선택합니다.

워크플로 세부 정보 페이지가 열립니다.

The screenshot shows the AWS Transfer Family console interface. On the left is a navigation sidebar with options like Servers, Connectors, AS2 Trading Partners, Certificates, Profiles, Workflows (selected), Feature Spotlight, What's New, and Documentation. The main content area displays the details for a specific workflow. At the top, there's a breadcrumb trail: Transfer Family > Workflows > w-1234567890abcdef0. A 'Delete' button is in the top right corner. The 'Description' section shows the Name as 'Test my workflow' and the Workflow description as 'Test workflow A'. Below this is the 'Nominal steps (1)' section, which contains a table with one step: a 'tag_step' of type 'TAG'. The 'Exception handlers (1)' section also contains a table with one handler: 'delete_if_exception' of type 'DELETE'. The 'In-flight executions (0)' section shows a search bar and a message 'No executions to display'. At the bottom, the 'Tags (1)' section includes a 'Manage tags' button and a note about using tags to organize workflows.

CLI

워크플로 세부 정보를 보려면 다음 예에서와 같이 `describe-workflow` CLI 명령을 사용합니다. 워크플로 ID `w-1234567890abcdef0`를 사용자의 고유한 값으로 바꿉니다. 자세한 내용은 AWS CLI 명령 참조에서 [describe-workflow](#)를 참조하세요.

```
# 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": "amzn-s3-demo-bucket",
            "Key": "home/shared_files/"
          }
        }
      }
    },
    "OnException": {}
  }
}

```

워크플로가 AWS CloudFormation 스택의 일부로 생성된 경우 CloudFormation 콘솔(<https://console.aws.amazon.com/cloudformation>)을 사용하여 워크플로를 관리할 수 있습니다.

Transfer Family > Workflows > w-
W-
Delete

W-

Info This workflow belongs to the AWS CloudFormation stack **stack-transfer-notifications-blog**. [Manage this stack](#) on the CloudFormation console.

Description

Name	Workflow description
Stack-managed	Workflow for blogpost

Nominal steps (4) [Info](#)

Number	Description	Type	Configuration
1	Lambda	CUSTOM	Details
2	CopyFile	COPY	Details
3	TagFileNotified	TAG	Details
4	DeleteOriginalSourceFile	DELETE	Details

사전 정의된 단계 사용

워크플로를 만들 때 이 항목에 설명된 다음과 같은 사전 정의된 단계 중 하나를 추가하도록 선택할 수 있습니다. 또한 사용자 지정 파일 처리 단계 추가를 선택할 수도 있습니다. 자세한 내용은 [the section called “사용자 지정 파일 처리 단계 사용”](#)를 참조하세요.

주제

- [파일 복사](#)
- [파일 복호화](#)
- [파일 태그 지정](#)
- [파일 삭제](#)
- [워크플로에 명명된 변수](#)
- [태그 및 삭제 워크플로 예시](#)

파일 복사

파일 복사 단계는 업로드된 파일의 사본을 새 Amazon S3 위치에 생성합니다. 현재는 Amazon S3에서만 파일 복사 단계를 사용할 수 있습니다.

다음 파일 복사 단계는 파일을 *amzn-s3-demo-destination-bucket*의 test 폴더에 복사합니다.

파일 복사 단계가 워크플로의 첫 단계가 아닌 경우 파일 위치를 지정할 수 있습니다. 파일 위치를 지정하면 이전 단계에서 사용한 파일 또는 업로드된 원본 파일을 복사할 수 있습니다. 이 기능을 사용하면 파일 보관 및 기록 보존을 위해 원본 파일을 그대로 유지하면서 원본 파일의 사본을 여러 개 만들 수 있습니다. 예는 [태그 및 삭제 워크플로 예시](#)를 참조하세요.

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

버킷 및 키 세부 정보 제공

파일 복사 단계의 대상 버킷 이름과 키를 제공해야 합니다. 키는 경로 이름 또는 파일 이름일 수 있습니다. 키를 경로 이름으로 취급할지 파일 이름으로 처리할지는 키 끝에 슬래시(/) 문자를 사용하는지 여부에 따라 결정됩니다.

마지막 문자가 /인 경우 파일은 폴더에 복사되며 이름은 변경되지 않습니다. 최종 문자가 영숫자인 경우 업로드한 파일의 이름이 키 값으로 바뀝니다. 이 경우 해당 이름의 파일이 이미 있는 경우 기존 항목 덮어쓰기 필드의 설정에 따라 동작이 달라집니다.

- 기존 파일 덮어쓰기를 선택하면 기존 파일이 처리 중인 파일로 대체됩니다.
- 기존 항목 덮어쓰기를 선택하지 않은 경우 아무 일도 일어나지 않고 워크플로 처리가 중지됩니다.

i Tip

동일한 파일 경로에서 동시 쓰기를 실행하면 파일을 덮어쓸 때 예상치 못한 동작이 발생할 수 있습니다.

예를 들어 키 값이 test/인 경우 업로드한 파일이 test 폴더에 복사됩니다. 키 값이 test/today인 경우(및 기존 파일 덮어쓰기가 선택된 경우) 업로드하는 모든 파일이 test 폴더에 today(으)로 이름이 지정된 파일에 복사되고 이후의 각 파일은 이전 파일을 덮어씁니다.

i Note

Amazon S3는 버킷과 객체를 지원하며 계층 구조가 없습니다. 하지만 객체 키 이름에 접두사와 구분 기호를 사용하여 계층 구조를 나타내고 폴더와 비슷한 방식으로 데이터를 구성할 수 있습니다.

파일 복사 단계에서 이름이 지정된 변수를 사용하세요.

파일 복사 단계에서 변수를 사용하여 파일을 사용자별 폴더에 동적으로 복사할 수 있습니다. 현재는 `${transfer:UserName}` 또는 `${transfer:UploadDate}`를 변수로 사용하여 파일을 업로드하는 특정 사용자의 대상 위치 또는 현재 날짜를 기준으로 파일을 복사할 수 있습니다.

다음 예에서는 사용자가 파일을 richard-roe 업로드하면 파일이 amzn-s3-demo-destination-bucket/richard-roe/processed/ 폴더에 복사됩니다. 사용자가 mary-major 파일을 업로드하면 파일이 amzn-s3-demo-destination-bucket/mary-major/processed/ 폴더에 복사됩니다.

Configure parameters

Configure copy parameters

Step name

dynamic-copy

Destination bucket name

amzn-s3-demo-destination-bucket ▼

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

마찬가지로 `${transfer:UploadDate}`를 변수로 사용하여 현재 날짜의 이름이 지정된 대상 위치에 파일을 복사할 수 있습니다. 다음 예에서 2022년 2월 1일에 대상을 `${transfer:UploadDate}/processed`으로 설정하면 업로드된 파일이 `amzn-s3-demo-destination-bucket/2022-02-01/processed/` 폴더에 복사됩니다.

Configure copy parameters

Step name

dynamic-copy-date

Destination bucket name

amzn-s3-demo-destination-bucket ▼

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

이 두 변수를 함께 사용하여 기능을 조합하여 사용할 수도 있습니다. 예를 들어 대상 키 접두사를 로 설정하면와 같은 중첩 폴더가 **folder/\${transfer:UserName}/\${transfer:UploadDate}/** 생성됩니다folder/marymajor/2023-01-05/.

복사 단계 IAM 권한

복사 단계를 성공적으로 수행하려면 워크플로의 실행 역할에 다음 권한이 포함되어 있어야 합니다.

```
{
  "Sid": "ListBucket",
  "Effect": "Allow",
  "Action": "s3:ListBucket",
  "Resource": [
    "arn:aws:s3:::amzn-s3-demo-destination-bucket"
  ]
}, {
  "Sid": "HomeDirObjectAccess",
  "Effect": "Allow",
  "Action": [
```

```

    "s3:PutObject",
    "s3:GetObject",
    "s3:DeleteObjectVersion",
    "s3:DeleteObject",
    "s3:GetObjectVersion"
  ],
  "Resource": "arn:aws:s3:::amzn-s3-demo-destination-bucket/*"
}

```

Note

이 `s3:ListBucket` 권한은 기존 항목 덮어쓰기를 선택하지 않은 경우에만 필요합니다. 이 권한은 버킷을 검사하여 같은 이름의 파일이 이미 존재하는지 확인합니다. 기존 파일 덮어쓰기를 선택한 경우 워크플로에서 파일을 확인할 필요 없이 그냥 쓰기만 하면 됩니다.

Amazon S3 파일에 태그가 있는 경우 IAM 정책에 하나 또는 두 개의 권한을 추가해야 합니다.

- 버전이 지정되지 않은 Amazon S3 파일에 `s3:GetObjectTagging`을 추가합니다.
- 버전이 지정된 Amazon S3 파일에 `s3:GetObjectVersionTagging`을 추가합니다.

파일 복호화

AWS 스토리지 블로그에는 Transfer Family 관리형 워크플로, [PGP 및를 사용한 암호화 및 암호화 해제를 사용하여 코드를 작성하지 않고 파일을 간단하게 해독하는 방법을 설명하는 게시물이 있습니다](#) [AWS Transfer Family](#).

지원되는 대칭 암호화 알고리즘

PGP 복호화의 경우 Transfer Family는 PGP 파일 내에서 실제 파일 데이터를 암호화하는 데 사용되는 대칭 암호화 알고리즘을 지원합니다.

- 지원되는 대칭 암호화 알고리즘에 대한 자세한 내용은 섹션을 참조하세요 [PGP 대칭 암호화 알고리즘](#).
- 이러한 대칭 알고리즘과 함께 사용되는 PGP 키 페어 알고리즘에 대한 자세한 내용은 섹션을 참조하세요 [PGP 키 페어 알고리즘](#).

워크플로에서 PGP 복호화 사용

Transfer Family는 Pretty Good Privacy(PGP) 복호화를 기본적으로 지원합니다. SFTP, FTPS 또는 FTP를 통해 Amazon Simple Storage Service(S3) 또는 Amazon Elastic File System(Amazon EFS)에 업로드된 파일에서 PGP 복호화를 사용할 수 있습니다.

PGP 복호화를 사용하려면 파일 복호화에 사용할 PGP 프라이빗 키를 생성하고 저장해야 합니다. 그러면 사용자가 Transfer Family 서버에 파일을 업로드하기 전에 해당 PGP 암호화 키를 사용하여 파일을 암호화할 수 있습니다. 암호화된 파일을 받은 후 워크플로에서 해당 파일을 복호화할 수 있습니다. 자세한 자습서는 [파일 복호화를 위한 관리형 워크플로 설정](#)를 참조하세요.

지원되는 PGP 알고리즘 및 권장 사항에 대한 자세한 내용은 섹션을 참조하세요 [PGP 암호화 및 복호화 알고리즘](#).

워크플로에서 PGP 복호화 사용

1. 워크플로를 호스팅할 Transfer Family 서버를 식별하거나 새 서버를 생성합니다. PGP 키를 AWS Secrets Manager에 올바른 암호 이름으로 저장하려면 먼저 서버 ID가 있어야 합니다.
2. PGP 키를 필요한 보안 암호 이름으로 AWS Secrets Manager에 저장합니다. 자세한 내용은 [PGP 키 관리](#)를 참조하세요. 워크플로는 Secrets Manager의 암호 이름을 기반으로 복호화에 사용할 올바른 PGP 키를 자동으로 찾을 수 있습니다.

Note

Secrets Manager에 암호를 저장하면 AWS 계정 에 요금이 발생합니다. 요금에 대한 자세한 내용은 [AWS Secrets Manager 요금](#)을 참조하세요.

3. PGP 키 쌍을 사용하여 파일을 암호화합니다. (지원되는 클라이언트 목록은 [지원되는 PGP 클라이언트](#)를 참조하세요.) 명령줄을 사용하는 경우 다음 명령을 실행하세요. 이 명령을 사용하려면 `username@example.com`을 PGP 키 쌍을 만들 때 사용한 이메일 주소로 바꾸세요. `testfile.txt`를 암호화할 파일의 이름으로 바꿉니다.

```
gpg -e -r username@example.com testfile.txt
```

Important

AWS Transfer Family 워크플로에 사용할 파일을 암호화할 때는 항상 `-r` 파라미터를 사용하여 비익명 수신자를 지정해야 합니다. 익명 암호화(수신자를 지정하지 않음)는 시스템이 복호화에 사용할 키를 식별할 수 없기 때문에 워크플로에서 복호화 실패를 일으킬 수 있습니다.

니다. 이 문제에 대한 디버깅 정보는 [에서 확인할 수 있습니다](#) [익명 수신자 암호화 문제 해결](#).

4. 암호화된 파일을 Transfer Family 서버에 업로드합니다.
5. 워크플로에서 복호화 단계를 구성합니다. 자세한 내용은 [복호화 단계 추가](#)를 참조하세요.

복호화 단계 추가

복호화 단계는 워크플로의 일부로 Amazon S3 또는 Amazon EFS에 업로드된 암호화된 파일을 복호화합니다. 복호화 구성에 대한 자세한 내용은 [워크플로에서 PGP 복호화 사용](#)을 참조하세요.

워크플로의 복호화 단계를 생성할 때는 복호화한 파일의 대상을 지정해야 합니다. 대상 위치에 파일이 이미 있는 경우 기존 파일을 덮어쓸지 여부도 선택해야 합니다. Amazon CloudWatch Logs를 사용하여 실시간으로 복호화 워크플로 결과를 모니터링하고 각 파일에 대한 감사 로그를 가져올 수 있습니다.

해당 단계에서 파일 복호화 타입을 선택하면 파라미터 구성 페이지가 나타납니다. PGP 복호화 파라미터 구성 섹션의 값을 입력합니다.

사용 가능한 옵션은 다음과 같습니다.

- 단계 이름 - 단계를 설명하는 이름을 입력합니다.
- 파일 위치 - 파일 위치를 지정하여 이전 단계에서 사용한 파일 또는 업로드된 원본 파일을 복호화할 수 있습니다.

Note

이 단계가 워크플로의 첫 번째 단계인 경우에는 이 파라미터를 사용할 수 없습니다.

- 복호화된 파일의 대상 — Amazon S3 버킷 또는 Amazon EFS 파일 시스템을 복호화된 파일의 대상으로 선택합니다.
 - Amazon S3를 선택하는 경우 대상 버킷 이름과 대상 키 접두사를 제공해야 합니다. 사용자 이름을 기준으로 대상 키 접두사를 파라미터화하려면 대상 키 접두사에 `${transfer:UserName}`를 입력합니다. 마찬가지로 업로드 날짜별로 대상 키 접두사를 파라미터화하려면 대상 키 접두사에 `${Transfer:UploadDate}`를 입력합니다.
 - Amazon EFS를 선택하는 경우 대상 파일 시스템과 경로를 제공해야 합니다.

Note

여기서 선택하는 스토리지 옵션은 이 워크플로가 연결된 Transfer Family 서버에서 사용하는 스토리지 시스템과 일치해야 합니다. 그렇지 않으면 이 워크플로를 실행하려고 시도하는 동안 오류가 발생합니다.

- 기존 파일 덮어쓰기 - 파일을 업로드했는데 같은 파일 이름을 가진 파일이 대상에 이미 있는 경우 동작은 이 파라미터의 설정에 따라 달라집니다.
- 기존 파일 덮어쓰기를 선택하면 기존 파일이 처리 중인 파일로 대체됩니다.
- 기존 항목 덮어쓰기를 선택하지 않은 경우 아무 일도 일어나지 않고 워크플로 처리가 중지됩니다.

Tip

동일한 파일 경로에서 동시 쓰기를 실행하면 파일을 덮어쓸 때 예상치 못한 동작이 발생할 수 있습니다.

다음 스크린샷은 파일 복호화 단계에서 선택할 수 있는 옵션의 예를 보여줍니다.

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.

복호화 단계를 위한 IAM 권한

복호화 단계를 성공적으로 수행하려면 워크플로의 실행 역할에 다음 권한이 포함되어 있어야 합니다.

```
{
  "Sid": "ListBucket",
  "Effect": "Allow",
  "Action": "s3:ListBucket",
  "Resource": [
    "arn:aws:s3:::amzn-s3-demo-destination-bucket"
  ]
}, {
  "Sid": "HomeDirObjectAccess",
  "Effect": "Allow",
  "Action": [
    "s3:PutObject",
    "s3:GetObject",
    "s3:DeleteObjectVersion",
    "s3:DeleteObject",
    "s3:GetObjectVersion"
  ],
  "Resource": "arn:aws:s3:::amzn-s3-demo-destination-bucket/*"
}, {
  "Sid": "Decrypt",
  "Effect": "Allow",
  "Action": [
    "secretsmanager:GetSecretValue",
  ],
  "Resource": "arn:aws:secretsmanager:region:account-id:secret:aws/transfer/*"
}
```

Note

이 s3:ListBucket 권한은 기존 항목 덮어쓰기를 선택하지 않은 경우에만 필요합니다. 이 권한은 버킷을 검사하여 같은 이름의 파일이 이미 존재하는지 확인합니다. 기존 파일 덮어쓰기를 선택한 경우 워크플로에서 파일을 확인할 필요 없이 그냥 쓰기만 하면 됩니다.

Amazon S3 파일에 태그가 있는 경우 IAM 정책에 하나 또는 두 개의 권한을 추가해야 합니다.

- 버전이 지정되지 않은 Amazon S3 파일에 s3:GetObjectTagging을 추가합니다.
- 버전이 지정된 Amazon S3 파일에 s3:GetObjectVersionTagging을 추가합니다.

파일 태그 지정

추가 다운스트림 처리를 위해 인입 파일에 태그를 지정하려면 태그 단계를 사용하세요. 인입 파일에 할당하려는 태그의 값을 입력합니다. 현재 태그 작업은 Transfer Family 서버 스토리지로 Amazon S3를 사용하는 경우에만 지원됩니다.

다음 태그 단계 예에서는 각각 `scan_outcome` 및 `clean`를 태그 키와 값으로 할당합니다.

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

태그 단계를 성공적으로 수행하려면 워크플로의 실행 역할에 다음 권한이 포함되어 있어야 합니다.

```
{
    "Sid": "Tag",
    "Effect": "Allow",
    "Action": [
        "s3:PutObjectTagging",
        "s3:PutObjectVersionTagging"
    ],
    "Resource": [
        "arn:aws:s3:::amzn-s3-demo-bucket/*"
    ]
}
```

Note

워크플로에 복사 또는 복호화 단계 전에 실행되는 태그 단계가 포함되어 있는 경우 IAM 정책에 하나 또는 두 개의 권한을 추가해야 합니다.

- 버전이 지정되지 않은 Amazon S3 파일에 `s3:GetObjectTagging`을 추가합니다.
- 버전이 지정된 Amazon S3 파일에 `s3:GetObjectVersionTagging`을 추가합니다.

파일 삭제

이전 워크플로 단계에서 처리된 파일을 삭제하거나 원래 업로드된 파일을 삭제하려면 파일 삭제 단계를 사용하세요.

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

삭제 단계를 성공적으로 수행하려면 워크플로의 실행 역할에 다음 권한이 포함되어 있어야 합니다.

```
{
    "Sid": "Delete",
    "Effect": "Allow",
    "Action": [
        "s3:DeleteObjectVersion",
        "s3:DeleteObject"
    ],
    "Resource": "arn:aws:secretsmanager:region:account-ID:secret:aws/transfer/
* "
}
```

워크플로에 명명된 변수

복사 및 복호화 단계의 경우 변수를 사용하여 작업을 동적으로 수행할 수 있습니다. 현재는 다음과 같은 명명된 변수를 AWS Transfer Family 지원합니다.

- 업로드하는 사용자를 기준으로 `${transfer:UserName}`을 사용하여 대상으로 파일을 복사하거나 암호를 해독합니다.
- 현재 날짜를 기준으로 `${transfer:UploadDate}`를 사용하여 대상으로 파일을 복사하거나 암호를 해독합니다.

태그 및 삭제 워크플로 예시

다음 예는 데이터 분석 플랫폼과 같은 다운스트림 애플리케이션에서 처리해야 하는 수신 파일에 태그를 지정하는 워크플로를 보여줍니다. 인입 파일에 태그를 지정한 후 워크플로는 원래 업로드된 파일을 삭제하여 스토리지 비용을 절약합니다.

Console

태그 및 이동 워크플로 예

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 워크플로를 선택합니다.
3. 워크플로 페이지에서 워크플로 생성을 선택합니다.
4. 워크플로 생성 페이지에서 설명을 입력합니다. 이 설명은 워크플로 페이지에 표시됩니다.
5. 첫 번째 단계(복사)를 추가합니다.
 - a. 공칭 단계 섹션에서 단계 추가를 선택합니다.
 - b. 파일 복사를 선택한 후 다음을 선택합니다.
 - c. 단계 이름을 입력한 다음 대상 버킷과 키 접두사를 선택합니다.

Step 1
Choose step type

Step 2
Configure parameters

Step 3
Review and create

Configure parameters

Configure copy parameters

Step name
copy-step-first-step

Destination bucket name
example-bucket ▼

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.
test/

☐ Overwrite existing

- d. 다음을 선택한 다음 해당 단계에 대한 세부 정보를 검토합니다.
 - e. 단계 생성을 선택하여 단계를 추가하고 계속합니다.
6. 두 번째 단계(태그)를 추가합니다.
- a. 공칭 단계 섹션에서 단계 추가를 선택합니다.
 - b. 파일 태그 지정을 선택한 후 다음을 선택합니다.
 - c. 단계 이름을 입력합니다.
 - d. 파일 위치의 경우 이전 단계에서 만든 파일에 태그 지정을 선택합니다.
 - e. 키와 값을 입력합니다.

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. 다음을 선택한 다음 해당 단계에 대한 세부 정보를 검토합니다.
 - g. 단계 생성을 선택하여 단계를 추가하고 계속합니다.
7. 세 번째 단계 추가(삭제).
- a. 공칭 단계 섹션에서 단계 추가를 선택합니다.
 - b. 파일 삭제를 선택하고 다음을 선택합니다.

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. 단계 이름을 입력합니다.

- d. 파일 위치에서 원본 소스 파일 삭제를 선택합니다.
 - e. 다음을 선택한 다음 해당 단계에 대한 세부 정보를 검토합니다.
 - f. 단계 생성을 선택하여 단계를 추가하고 계속합니다.
8. 워크플로 구성을 검토한 다음 워크플로 생성을 선택합니다.

CLI

태그 및 이동 워크플로 예

- 다음 코드를 파일에 저장합니다. 예: tagAndMoveWorkflow.json. 각 *user input placeholder*를 사용자의 정보로 바꿉니다.

```
[
  {
    "Type": "COPY",
    "CopyStepDetails": {
      "Name": "CopyStep",
      "DestinationFileLocation": {
        "S3FileLocation": {
          "Bucket": "amzn-s3-demo-bucket",
          "Key": "test/"
        }
      }
    }
  },
  {
    "Type": "TAG",
    "TagStepDetails": {
      "Name": "TagStep",
      "Tags": [
        {
          "Key": "name",
          "Value": "demo"
        }
      ],
      "SourceFileLocation": "${previous.file}"
    }
  },
  {
    "Type": "DELETE",
    "DeleteStepDetails": {
```

```

        "Name": "DeleteStep",
        "SourceFileLocation": "${original.file}"
    }
}
]

```

첫 번째 단계에서는 업로드된 파일을 새 Amazon S3 위치에 복사합니다. 두 번째 단계에서는 새 위치에 복사된 파일(previous.file)에 태그(카-값 쌍)를 추가합니다. 마지막으로 세 번째 단계에서는 원본 파일(original.file)을 삭제합니다.

2. 저장된 파일에서 워크플로를 생성합니다. 각 *user input placeholder*를 사용자의 정보로 바꿉니다.

```

aws transfer create-workflow --description "short-description" --steps
file://path-to-file --region region-ID

```

예제:

```

aws transfer create-workflow --description "copy-tag-delete workflow" --steps
file://tagAndMoveWorkflow.json --region us-east-1

```

Note

파일을 사용하여 파라미터를 로드하는 방법에 대한 자세한 내용은 [파일에서 파라미터를 로드하는 방법](#)을 참조하세요.

3. 기존 서버를 업데이트합니다.

Note

이 단계에서는 이미 Transfer Family 서버가 있고 이 서버에 워크플로를 연결하려는 것으로 가정합니다. 그렇지 않은 경우 [SFTP, FTPS 또는 FTP 서버 엔드포인트 구성](#)를 참조하세요. 각 *user input placeholder*를 사용자의 정보로 바꿉니다.

```

aws transfer update-server --server-id server-ID --region region-ID
--workflow-details '{"OnUpload": [{ "WorkflowId": "workflow-ID", "ExecutionRole": "execution-role-ARN" } ]}'

```

예제:

```
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"}]}'
```

사용자 지정 파일 처리 단계 사용

사용자 지정 파일 처리 단계를 사용하면 AWS Lambda를 사용하여 Bring Your Own 파일 처리 로직을 사용할 수 있습니다. 파일이 도착하면 Transfer Family 서버는 파일 암호화, 멀웨어 검사 또는 잘못된 파일 타입 검사와 같은 사용자 지정 파일 처리 로직이 포함된 Lambda 함수를 간접적으로 호출합니다. 다음 예시에서는 대상 AWS Lambda 함수를 사용하여 이전 단계의 출력 파일을 처리합니다.

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

Lambda 함수의 예는 [사용자 지정 워크플로 단계를 위한 Lambda 함수 예](#)를 참조하세요. 예를 들어 이벤트(Lambda로 전달된 파일의 위치 포함)는 [파일 업로드 AWS Lambda 시로 전송된 이벤트 예](#)를 참조하세요.

사용자 지정 워크플로 단계를 사용하면 [SendWorkflowStepState](#) API 작업을 직접적으로 호출하도록 Lambda 함수를 구성해야 합니다. SendWorkflowStepState는 단계가 성공 또는 실패 상태로 완료되었음을 워크플로 실행에 알립니다. SendWorkflowStepState API 작업의 상태는 Lambda 함수의 결과에 따라 선형 시퀀스의 예외 처리 단계 또는 공칭 단계를 간접적으로 호출합니다.

Lambda 함수가 실패하거나 제한 시간이 초과되면 단계가 실패하고 CloudWatch 로그에서 StepErrored를 확인할 수 있습니다. Lambda 함수가 공칭 단계의 일부이고 함수가 Status="FAILURE"로 SendWorkflowStepState에 응답하거나 제한 시간이 초과되면 흐름은 예외 처리 단계로 계속됩니다. 이 경우 워크플로는 나머지(있는 경우) 공칭 단계를 계속 실행하지 않습니다. 자세한 내용은 [워크플로의 예외 처리](#)를 참조하세요.

SendWorkflowStepState API 작업을 직접적으로 호출할 때는 다음 파라미터를 전송해야 합니다.

```
{
  "ExecutionId": "string",
  "Status": "string",
  "Token": "string",
  "WorkflowId": "string"
}
```

Lambda 함수가 실행될 때 전달되는 입력 이벤트에서 ExecutionId, Token, 및 WorkflowId를 추출할 수 있습니다(예는 다음 섹션에 표시됨). 이때 Status 값은 SUCCESS 또는 FAILURE가 될 수 있습니다.

Lambda 함수에서 SendWorkflowStepState API 작업을 호출하려면 [관리형 워크플로가 도입](#)된 후 게시된 AWS SDK 버전을 사용해야 합니다.

여러 Lambda 함수를 연속적으로 사용

여러 사용자 지정 단계를 차례로 사용하는 경우 파일 위치 옵션은 단일 사용자 지정 단계만 사용하는 경우와 다르게 작동합니다. Transfer Family는 Lambda 처리 파일을 다음 단계의 입력으로 사용하기 위해 다시 전달하는 것을 지원하지 않습니다. 따라서 previous.file 옵션을 사용하도록 구성된 사용자 지정 단계가 여러 개 있는 경우 모두 동일한 파일 위치(첫 번째 사용자 지정 단계의 입력 파일 위치)를 사용합니다.

Note

사용자 지정 단계 이후에 사전 정의된 단계(태그 지정, 복사, 복호화 또는 삭제)가 있는 경우에도 previous.file 설정이 다르게 작동합니다. 사전 정의된 단계가 previous.file 설정을

사용하도록 구성된 경우 사전 정의된 단계는 사용자 정의 단계에서 사용한 것과 동일한 입력 파일을 사용합니다. 사용자 정의 단계에서 처리된 파일은 사전 정의된 단계로 전달되지 않습니다.

사용자 정의 처리 후 파일에 액세스

Amazon S3를 스토리지로 사용하고 있고 워크플로에 원래 업로드된 파일에서 작업을 수행하는 사용자 지정 단계가 포함되어 있는 경우, 후속 단계에서 처리된 파일에 액세스할 수 없습니다. 즉, 사용자 지정 단계 이후의 모든 단계는 사용자 지정 단계 출력의 업데이트된 파일을 참조할 수 없습니다.

예를 들어 워크플로에 다음과 같은 세 단계가 있다고 가정하겠습니다.

- 1단계 — example-file.txt라는 이름의 파일을 업로드합니다.
- 2단계 — 어떤 식으로든 example-file.txt를 변경하는 Lambda 함수를 간접적으로 호출합니다.
- 3단계 — example-file.txt의 업데이트된 버전에서 추가 처리를 시도합니다.

3단계에 대한 sourceFileLocation을 `${original.file}`로 구성한 경우 3단계에서는 1단계에서 서버가 스토리지에 파일을 업로드한 시점의 원래 파일 위치를 사용합니다. 3단계에서 `${previous.file}`을 사용하는 경우 3단계에서는 2단계에서 입력으로 사용한 파일 위치를 다시 사용합니다.

따라서 3단계에서 오류가 발생합니다. 예를 들어 3단계에서 업데이트된 example-file.txt를 복사하려고 하면 다음과 같은 오류 메시지가 나타납니다.

```
{
  "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"
  },
}
```

이 오류는 사용자 지정 단계에서 원본 파일과 일치하지 않도록 example-file.txt의 개체 태그(ETag)를 수정하기 때문에 발생합니다.

Note

Amazon EFS는 개체 태그를 사용하여 파일을 식별하지 않기 때문에 Amazon EFS를 사용하는 경우에는 이 동작이 발생하지 않습니다.

파일 업로드 AWS Lambda 시 로 전송된 이벤트 예

다음 예제에서는 파일 업로드가 완료될 AWS Lambda 때 로 전송되는 이벤트를 보여줍니다. 한 예로, 도메인이 Amazon S3으로 구성된 Transfer Family 서버를 사용합니다. 다른 예시에서는 도메인이 Amazon EFS를 사용하는 Transfer Family 서버를 사용합니다.

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": "amzn-s3-demo-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"
}
}

```

사용자 지정 워크플로 단계를 위한 Lambda 함수 예

다음 Lambda 함수는 실행 상태와 관련된 정보를 추출한 다음 [SendWorkflowStepState](#) API 작업을 직접적으로 호출하여 해당 단계의 워크플로에 상태 SUCCESS 또는 FAILURE를 반환합니다. 함수가 SendWorkflowStepState API 작업을 직접적으로 호출하기 전에 Lambda가 워크플로 로직을 기반으로 조치를 취하도록 구성할 수 있습니다.

```

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)
}
```

사용자 지정 단계 IAM 권한

Lambda를 직접적으로 호출하는 단계가 성공하도록 허용하려면 워크플로의 실행 역할에 다음 권한이 포함되어 있어야 합니다.

```
{
  "Sid": "Custom",
  "Effect": "Allow",
  "Action": [
    "lambda:InvokeFunction"
  ],
  "Resource": [
    "arn:aws:lambda:region:account-id:function:function-name"
  ]
}
```

워크플로에 대한 IAM 정책

서버에 워크플로를 추가할 때는 실행 역할을 선택해야 합니다. 서버는 워크플로를 실행할 때 이 역할을 사용합니다. 역할에 적절한 권한이 없는 경우 워크플로를 실행할 AWS Transfer Family 수 없습니다.

이 섹션에서는 워크플로를 실행하는 데 사용할 수 있는 한 가지 AWS Identity and Access Management (IAM) 권한 세트에 대해 설명합니다. 다른 예는 이 주제 후반부에서 설명합니다.

Note

Amazon S3 파일에 태그가 있는 경우 IAM 정책에 하나 또는 두 개의 권한을 추가해야 합니다.

- 버전이 지정되지 않은 Amazon S3 파일에 s3:GetObjectTagging을 추가합니다.
- 버전이 지정된 Amazon S3 파일에 s3:GetObjectVersionTagging을 추가합니다.

워크플로에 대한 실행 역할을 만들려면

1. 새 IAM 역할을 생성하고 AWS 관리형 정책을 AWSTransferFullAccess 역할에 추가합니다. 새 IAM 역할을 생성하는 방법에 대한 자세한 내용은 [the section called “IAM 역할 및 정책 생성”](#)을 참조하세요.
2. 다음 권한이 있는 다른 정책을 생성하여 이를 역할에 연결합니다. 각 *user input placeholder*를 사용자의 정보로 바꿉니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ConsoleAccess",
      "Effect": "Allow",
      "Action": "s3:GetBucketLocation",
      "Resource": "*"
    },
    {
      "Sid": "ListObjectsInBucket",
      "Effect": "Allow",
      "Action": "s3:ListBucket",
      "Resource": [
        "arn:aws:s3:::amzn-s3-demo-bucket"
      ]
    },
    {
      "Sid": "AllObjectActions",
      "Effect": "Allow",
      "Action": "s3:*Object",
      "Resource": [
        "arn:aws:s3:::amzn-s3-demo-bucket/*"
      ]
    },
    {
      "Sid": "GetObjectVersion",
      "Effect": "Allow",
      "Action": "s3:GetObjectVersion",
      "Resource": [
        "arn:aws:s3:::amzn-s3-demo-bucket/*"
      ]
    },
    {
```

```

        "Sid": "Custom",
        "Effect": "Allow",
        "Action": [
            "lambda:InvokeFunction"
        ],
        "Resource": [
            "arn:aws:lambda:us-east-1:123456789012:function:function-name"
        ]
    },
    {
        "Sid": "Tag",
        "Effect": "Allow",
        "Action": [
            "s3:PutObjectTagging",
            "s3:PutObjectVersionTagging"
        ],
        "Resource": [
            "arn:aws:s3:::amzn-s3-demo-bucket/*"
        ]
    }
]
}

```

3. 이 역할을 저장하고 서버에 워크플로를 추가할 때 실행 역할로 지정하세요.

Note

IAM 역할을 구성할 때는 워크플로에 대해 가능한 한 리소스에 대한 액세스를 제한할 것을 AWS 권장합니다.

워크플로 신뢰 관계

워크플로 실행 역할에는 `transfer.amazonaws.com`과의 신뢰 관계도 필요합니다. AWS Transfer Family에 대한 신뢰 관계를 설정하려면 [신뢰 관계를 구축하기 위해](#)를 참조하세요.

신뢰 관계를 구축하는 동시에 혼란스러운 대리인 문제를 피하도록 조치를 취할 수도 있습니다. 이 문제에 대한 설명과 이를 방지하는 방법의 예는 [the section called “교차 서비스 혼동된 대리자 방지”](#)을 참조하세요.

실행 역할 예시: 복호화, 복사, 태그 지정

태그 지정, 복사 및 복호화 단계가 포함된 워크플로가 있는 경우 다음 IAM 정책을 사용할 수 있습니다. 각 *user input placeholder*를 사용자의 정보로 바꿉니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "CopyRead",
      "Effect": "Allow",
      "Action": [
        "s3:GetObject",
        "s3:GetObjectTagging",
        "s3:GetObjectVersionTagging"
      ],
      "Resource": "arn:aws:s3:::amzn-s3-demo-source-bucket/*"
    },
    {
      "Sid": "CopyWrite",
      "Effect": "Allow",
      "Action": [
        "s3:PutObject",
        "s3:PutObjectTagging"
      ],
      "Resource": "arn:aws:s3:::amzn-s3-demo-destination-bucket/*"
    },
    {
      "Sid": "CopyList",
      "Effect": "Allow",
      "Action": "s3:ListBucket",
      "Resource": [
        "arn:aws:s3:::amzn-s3-demo-source-bucket",
        "arn:aws:s3:::amzn-s3-demo-destination-bucket"
      ]
    },
    {
      "Sid": "Tag",
      "Effect": "Allow",
      "Action": [
        "s3:PutObjectTagging",
        "s3:PutObjectVersionTagging"
      ]
    }
  ]
}
```

```

    ],
    "Resource": "arn:aws:s3:::amzn-s3-demo-destination-bucket/*",
    "Condition": {
        "StringEquals": {
            "s3:RequestObjectTag/Archive": "yes"
        }
    }
},
{
    "Sid": "ListBucket",
    "Effect": "Allow",
    "Action": "s3:ListBucket",
    "Resource": [
        "arn:aws:s3:::amzn-s3-demo-destination-bucket"
    ]
},
{
    "Sid": "HomeDirObjectAccess",
    "Effect": "Allow",
    "Action": [
        "s3:PutObject",
        "s3:GetObject",
        "s3:DeleteObjectVersion",
        "s3:DeleteObject",
        "s3:GetObjectVersion"
    ],
    "Resource": "arn:aws:s3:::amzn-s3-demo-destination-bucket/*"
},
{
    "Sid": "Decrypt",
    "Effect": "Allow",
    "Action": [
        "secretsmanager:GetSecretValue"
    ],
    "Resource": "arn:aws:secretsmanager:us-east-1:123456789012:secret:aws/transfer/*"
}
]
}

```

실행 역할 예시: 함수 실행 및 삭제

이 예제에는 AWS Lambda 함수를 호출하는 워크플로가 있습니다. 워크플로에서 업로드된 파일을 삭제하고 이전 단계에서 실패한 워크플로 실행에 대해 조치를 취하는 예외 처리 단계가 있는 경우 다음 IAM 정책을 사용합니다. 각 *user input placeholder*를 사용자의 정보로 바꿉니다.

```
{
  "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:us-east-1:123456789012:function:function-name"
      ]
    }
  ]
}
```

워크플로의 예외 처리

워크플로 실행 중에 오류가 발생하는 경우 지정한 예외 처리 단계가 실행됩니다. 워크플로의 공칭 단계를 지정하는 것과 같은 방식으로 워크플로의 오류 처리 단계를 지정합니다. 예를 들어 들어오는 파일의 유효성을 검사하기 위해 공칭의 단계로 사용자 지정 처리를 구성했다고 가정해 보겠습니다. 파일 검증이 실패하는 경우 예외 처리 단계를 통해 관리자에게 이메일을 보낼 수 있습니다.

다음 예 워크플로에는 두 단계가 포함되어 있습니다.

- 업로드된 파일이 CSV 형식인지 여부를 확인하는 공칭 단계 1개
- 업로드된 파일이 CSV 형식이 아니고, 공칭 단계가 실패하는 경우 이메일을 보내는 예외 처리 단계 1개

예외 처리 단계를 시작하려면 공칭 단계의 AWS Lambda 함수가 로 응답해야 합니다. `Status="FAILURE"`. 워크플로에서 오류를 처리하는 방법에 대한 자세한 내용은 [the section called “사용자 지정 파일 처리 단계 사용”](#)를 참조하세요.

w-1234567890abcdef0

Delete

Description

Name

Delete after upload

Workflow description

test-my-workflow

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

워크플로 실행 모니터링

Amazon CloudWatch에서 실행하는 AWS 리소스와 애플리케이션을 AWS 클라우드 실시간으로 모니터링합니다. Amazon CloudWatch를 사용하여 워크플로에 대해 측정할 수 있는 변수인 지표를 수집하고 추적할 수 있습니다. Amazon CloudWatch를 사용하여 워크플로 지표와 통합 로그를 볼 수 있습니다.

워크플로를 위한 CloudWatch 로깅

CloudWatch는 워크플로 진행 상황 및 결과에 대한 통합 감사 및 로깅을 제공합니다.

워크플로에 대한 Amazon CloudWatch Logs 보기

1. <https://console.aws.amazon.com/cloudwatch/>에서 Amazon CloudWatch 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 로그를 선택한 다음, 로그 그룹을 선택합니다.
3. 로그 그룹 페이지의 탐색 모음에서 AWS Transfer Family 서버에 적합한 리전을 선택합니다.
4. 서버에 해당하는 로그 그룹을 선택합니다.

예를 들어 서버 ID가 s-1234567890abcdef0인 경우 로그 그룹은 /aws/transfer/s-1234567890abcdef0입니다.

5. 서버의 로그 그룹 세부 정보 페이지에는 가장 최근의 로그 스트림이 표시됩니다. 탐색 중인 사용자에 대한 로그 스트림은 두 가지입니다.

- 각 Secure Shell(SSH) File Transfer 프로토콜(SFTP) 세션별 로그 스트림
- 서버에서 실행 중인 워크플로의 로그 스트림 워크플로의 로그 스트림 형식은 *username.workflowID.uniqueStreamSuffix*입니다.

예를 들어 사용자가 mary-major이면 다음과 같은 로그 스트림이 있습니다.

```
mary-major-east.1234567890abcdef0
mary.w-abcdef01234567890.021345abcdef6789
```

 Note

이 예에 나열된 16자리 영숫자 식별자는 가상의 식별자입니다. Amazon CloudWatch에 표시되는 값은 다릅니다.

mary-major-usa-east.1234567890abcdef0의 로그 이벤트 페이지에는 각 사용자 세션의 세부 정보가 표시되고 mary.w-abcdef01234567890.021345abcdef6789 로그 스트림에는 워크플로에 대한 세부 정보가 포함됩니다.

다음은 복사 단계가 포함된 워크플로(w-abcdef01234567890)를 기반으로 하는 mary.w-abcdef01234567890.021345abcdef6789의 샘플 로그 스트림입니다.

```
{
  "type": "ExecutionStarted",
  "details": {
```

```

        "input": {
            "initialFileLocation": {
                "bucket": "amzn-s3-demo-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": "amzn-s3-demo-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"
    }
}
}

```

워크플로를 위한 CloudWatch 지표

AWS Transfer Family 는 워크플로에 대한 여러 지표를 제공합니다. 이전 1분 간 시작, 성공적으로 완료, 및 실패한 워크플로 실행 수에 대한 지표를 볼 수 있습니다. Transfer Family에 대한 모든 CloudWatch 지표는 [Transfer Family 서버에 CloudWatch 지표 사용](#)에 설명되어 있습니다.

템플릿에서 워크플로 생성

템플릿에서 워크플로와 서버를 생성하는 CloudFormation 스택을 배포할 수 있습니다. 이 절차에는 워크플로를 빠르게 배포하는 데 사용할 수 있는 예가 포함되어 있습니다.

AWS Transfer Family 워크플로 및 서버를 생성하는 CloudFormation 스택을 생성하려면

1. <https://console.aws.amazon.com/cloudformation> CloudFormation 콘솔을 엽니다.
2. 다음 코드를 파일에 저장합니다.

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. 다음 항목을 실제 값으로 대체합니다.

- *workflow-execution-role-arn*을 실제 워크플로 실행 역할의 ARN으로 바꿉니다. 예:
arn:aws:transfer:us-east-2:111122223333:workflow/w-1234567890abcdef0

- `arn:aws:lambda:region:account-id:function:function-name`을 Lambda 함수 ARN으로 바꿉니다. 예를 들어 `arn:aws:lambda:us-east-2:123456789012:function:example-lambda-idp`입니다.
4. AWS CloudFormation 사용 설명서의 CloudFormation 스택 템플릿 [선택에서 기존 템플릿의 스택](#) 배포 지침을 따릅니다.

스택을 배포한 후에는 CloudFormation 콘솔의 출력 탭에서 스택에 대한 세부 정보를 볼 수 있습니다. 템플릿은 서비스 관리 사용자를 사용하는 새 AWS Transfer Family SFTP 서버와 새 워크플로를 생성하고 워크플로를 새 서버와 연결합니다.

Transfer Family 서버에서 워크플로 제거

워크플로를 Transfer Family 서버와 연결했는데 이제 해당 연결을 제거하려는 경우 콘솔을 사용하거나 프로그래밍 방식으로 제거할 수 있습니다.

Console

Transfer Family 서버에서 워크플로 제거

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 서버를 선택합니다.
3. 서버 ID 옆에서 서버 식별자를 선택합니다.
4. 서버의 세부 정보 페이지에서 추가 세부 정보 섹션까지 아래로 스크롤한 다음 편집을 선택합니다.
5. 추가 세부 정보 편집 페이지의 관리형 워크플로 섹션에서 모든 설정에 대한 정보를 지웁니다.
 - 전체 파일 업로드를 위한 워크플로의 워크플로 목록에서 대시(-)를 선택합니다.
 - 아직 지우지 않은 경우 부분 파일 업로드 워크플로의 워크플로 목록에서 대시(-)를 선택합니다.
 - 관리형 워크플로 실행 역할의 역할 목록에서 대시(-)를 선택합니다.

대시가 보이지 않는 경우 각 메뉴의 첫 번째 값이므로 대시가 보일 때까지 위로 스크롤하세요.

결과는 다음과 같아야 합니다.

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. 변경 내용을 저장하려면 아래로 스크롤하여 저장을 선택합니다.

CLI

update-server(또는 API용 UpdateServer) 직접 호출을 사용하고 OnUpload 및 OnPartialUpload 파라미터에 빈 인수를 제공합니다.

에서 다음 명령을 AWS CLI 실행합니다.

```
aws transfer update-server --server-id your-server-id --workflow-details
'{"OnPartialUpload":[],"OnUpload":[]}'
```

*your-server-id*를 서버의 ID로 바꿉니다. 예를 들어, 서버 ID가 s-01234567890abcdef인 경우 명령은 다음과 같습니다.

```
aws transfer update-server --server-id s-01234567890abcdef --workflow-details
'{"OnPartialUpload":[],"OnUpload":[]}'
```

관리형 워크플로 제한 및 제약 조건

제한 사항

현재 다음과 같은 제한 사항이 AWS Transfer Family의 업로드 후 처리 워크플로에 적용됩니다.

- 교차 계정 및 교차 리전 AWS Lambda 함수는 지원되지 않습니다. 그러나 AWS Identity and Access Management (IAM) 정책이 올바르게 구성된 경우 계정 간에 복사할 수 있습니다.

- 모든 워크플로 단계에서 워크플로로 액세스하는 모든 Amazon S3 버킷은 워크플로와 동일한 리전에 있어야 합니다.
- 복호화 단계의 경우 복호화 대상이 리전 및 지원 스토어의 원본과 일치해야 합니다(예를 들어, 해독할 파일이 Amazon S3에 저장되어 있는 경우 지정된 대상도 Amazon S3에 있어야 함).
- 비동기 사용자 지정 단계만 지원됩니다.
- 사용자 지정 단계 제한 시간은 근사값입니다. 즉, 지정된 제한 시간보다 약간 더 오래 걸릴 수 있습니다. 또한 워크플로는 Lambda 함수에 따라 달라집니다. 따라서 실행 중에 함수가 지연되는 경우 워크플로는 지연을 인식하지 못합니다.
- 제한 한도를 초과하는 경우 Transfer Family는 대기열에 워크플로 작업을 추가하지 않습니다.
- 크기가 0인 파일에 대해서는 워크플로가 시작되지 않습니다. 크기가 0보다 큰 파일은 관련 워크플로를 시작합니다.
- AS2 프로토콜을 사용하는 Transfer Family 서버에 파일 처리 워크플로를 연결할 수 있습니다. 하지만 AS2 메시지는 서버에 연결된 워크플로를 실행하지 않습니다.

제한 사항

또한 Transfer Family의 워크플로에는 다음과 같은 기능 제한이 적용됩니다.

- 리전별, 계정별 워크플로 수는 10개로 제한됩니다.
- 사용자 지정 단계의 최대 제한 시간은 30분입니다.
- 워크플로의 최대 단계 수는 8개입니다.
- 워크플로당 최대 태그 수는 50개입니다.
- 복호화 단계가 포함된 최대 동시 실행 수는 워크플로당 250개입니다.
- 사용자당 Transfer Family 서버당 최대 3개의 PGP 프라이빗 키를 저장할 수 있습니다.
- 복호화된 파일의 최대 크기는 10GB입니다.
- 버스트 용량이 100이고 리필 비율이 1인 [토큰 버킷](#) 시스템을 사용하여 새 실행률을 제한합니다.
- 서버에서 워크플로를 제거하고 새 워크플로로 바꾸거나 워크플로의 실행 역할에 영향을 미치는 서버 구성을 업데이트할 때마다 새 워크플로를 실행하기 전에 약 10분을 기다려야 합니다. Transfer Family 서버는 워크플로 세부 정보를 캐시하며 서버가 캐시를 새로 고치는 데 10분이 걸립니다.

또한 활성 SFTP 세션에서 로그아웃한 다음 10분 대기 시간이 지난 후 다시 로그인해야 변경 사항을 확인할 수 있습니다.

서버 관리

이 섹션에서는 서버 목록을 보는 방법, 서버 세부 정보를 보는 방법, 서버 세부 정보를 편집하는 방법, SFTP 지원 서버의 호스트 키를 변경하는 방법에 대한 정보를 찾을 수 있습니다.

주제

- [서버 목록 보기](#)
- [서버 삭제](#)
- [SFTP, FTPS 및 FTP 서버 세부 정보 보기](#)
- [AS2 서버 세부 정보 보기](#)
- [Transfer Family 서버에 대한 IPv6 지원](#)
- [서버 세부 정보 편집](#)
- [자격 증명 공급자 구성 편집](#)
- [SFTP 지원 서버의 호스트 키 관리](#)
- [콘솔 내 사용량 모니터링](#)

서버 목록 보기

AWS Transfer Family 콘솔에서 선택한 AWS 리전에 있는 모든 서버의 목록을 찾을 수 있습니다.

AWS 리전에 있는 서버 목록을 찾으려면

- <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.

현재 AWS 리전에 서버가 하나 이상 있는 경우 콘솔이 열리고 서버 목록이 표시됩니다. 서버 목록이 보이지 않으면, 올바른 AWS 리전에 있는지 확인하세요. 탐색 창에서 서버를 선택할 수도 있습니다.

로그의 세부 정보 보기에 대한 자세한 내용은 [SFTP, FTPS 및 FTP 서버 세부 정보 보기](#)를 참조하세요.

서버 삭제

이 절차에서는 AWS Transfer Family 콘솔 또는를 사용하여 Transfer Family 서버를 삭제하는 방법을 설명합니다 AWS CLI.

⚠ Important

엔드포인트에 액세스할 수 있도록 설정된 각 프로토콜에 대해 서버를 삭제할 때까지 요금이 청구됩니다.

⚠ Warning

서버를 삭제하면 해당 사용자가 모두 삭제됩니다. 서버를 사용하여 액세스한 버킷의 데이터는 삭제되지 않으며 해당 Amazon S3 버킷에 대한 권한을 가진 AWS 사용자가 계속 액세스할 수 있습니다.

Console

콘솔을 사용하여 서버를 삭제하려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 서버를 선택합니다.
3. 삭제할 서버의 확인란을 선택합니다.
4. 작업에 대해 삭제를 선택합니다.
5. 표시되는 확인 대화 상자에서 단어 **delete**를 입력한 다음 삭제를 선택하여 서버 삭제를 확인합니다.

서버 페이지에서 서버가 삭제되며 더 이상 요금이 청구되지 않습니다.

AWS CLI

CLI를 사용하여 서버를 삭제하려면

1. (선택 사항) 다음 명령을 실행하여 영구적으로 삭제하려는 서버의 세부 정보를 봅니다.

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

이 `describe-server` 명령은 서버의 모든 세부 정보를 반환합니다.

2. 다음 명령을 실행하여 서버를 삭제합니다.

```
aws transfer delete-server --server-id your-server-id
```

성공하면 명령이 서버를 삭제하고 정보를 반환하지 않습니다.

SFTP, FTPS 및 FTP 서버 세부 정보 보기

개별 AWS Transfer Family 서버의 세부 정보 및 속성 목록을 찾을 수 있습니다. 서버 속성에는 프로토콜, 자격 증명 제공자, 상태, 엔드포인트 타입, 사용자 지정 호스트 이름, 엔드포인트, 사용자, 로깅 역할, 서버 호스트 키 및 태그가 포함됩니다.

서버 세부 정보를 보려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 탐색 창에서 서버를 선택합니다.
3. 서버 ID 옆에서 식별자를 선택하여 다음과 같은 서버 세부 정보 페이지를 표시합니다.

편집을 선택하여 이 페이지에서 서버의 속성을 변경할 수 있습니다. 서버 편집에 대한 자세한 내용은 [서버 세부 정보 편집](#)을 참조하세요. AS2 서버의 세부 정보 페이지는 약간 다릅니다. AS2 서버의 경우 섹션을 참조하세요 [AS2 서버 세부 정보 보기](#).

Protocols

[Edit](#)

Protocols over which clients can connect to your server's endpoint

- SFTP

Identity provider

[Edit](#)

Identity provider type
Service managed

Note

서버 호스트 키 설명 및 가져온 날짜 값은 2022년 9월 기준으로 새로 추가되었습니다. 이러한 값은 다중 호스트 키 기능을 지원하기 위해 도입되었습니다. 이 기능을 사용하려면 여러 호스트 키가 도입되기 전에 사용 중이던 단일 호스트 키를 모두 마이그레이션해야 했습니다.

마이그레이션된 서버 호스트 키의 가져온 날짜 값은 서버의 마지막 수정 날짜로 설정됩니다. 즉, 마이그레이션된 호스트 키로 표시되는 날짜는 어떤 식으로든 서버 호스트 키 마이그레이션 전에 서버를 마지막으로 수정한 날짜와 일치합니다.

마이그레이션된 유일한 키는 가장 오래된 또는 유일한 서버 호스트 키입니다. 모든 추가 키에는 가져온 시점의 실제 날짜가 표시됩니다. 또한 마이그레이션된 키에는 마이그레이션된 키임을 쉽게 식별할 수 있는 설명이 있습니다.

마이그레이션은 9월 2일에서 9월 13일 사이에 이루어졌습니다. 이 범위 내의 실제 마이그레이션 날짜는 서버 리전에 따라 다릅니다.

Additional details

[Edit](#)

Log group

[/aws/transfer/s-](#)

Logging role

-

Security Policy

[Info](#)
 TransferSecurityPolicy-PQ-SSH-Experimental-2023-04

Domain

Amazon S3

Login display banner

[View the display message](#)

Optimized directories

[Info](#)
 ENABLED

SetStat option

Ignore

TLS session resumption

-

Passive IP

-

AS2 서버 세부 정보 보기

개별 AWS Transfer Family 서버의 세부 정보 및 속성 목록을 찾을 수 있습니다. 서버 속성에는 프로토콜, 상태 등이 포함됩니다. AS2 서버의 경우 AS2 비동기 MDN 송신 IP 주소를 볼 수도 있습니다.

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.

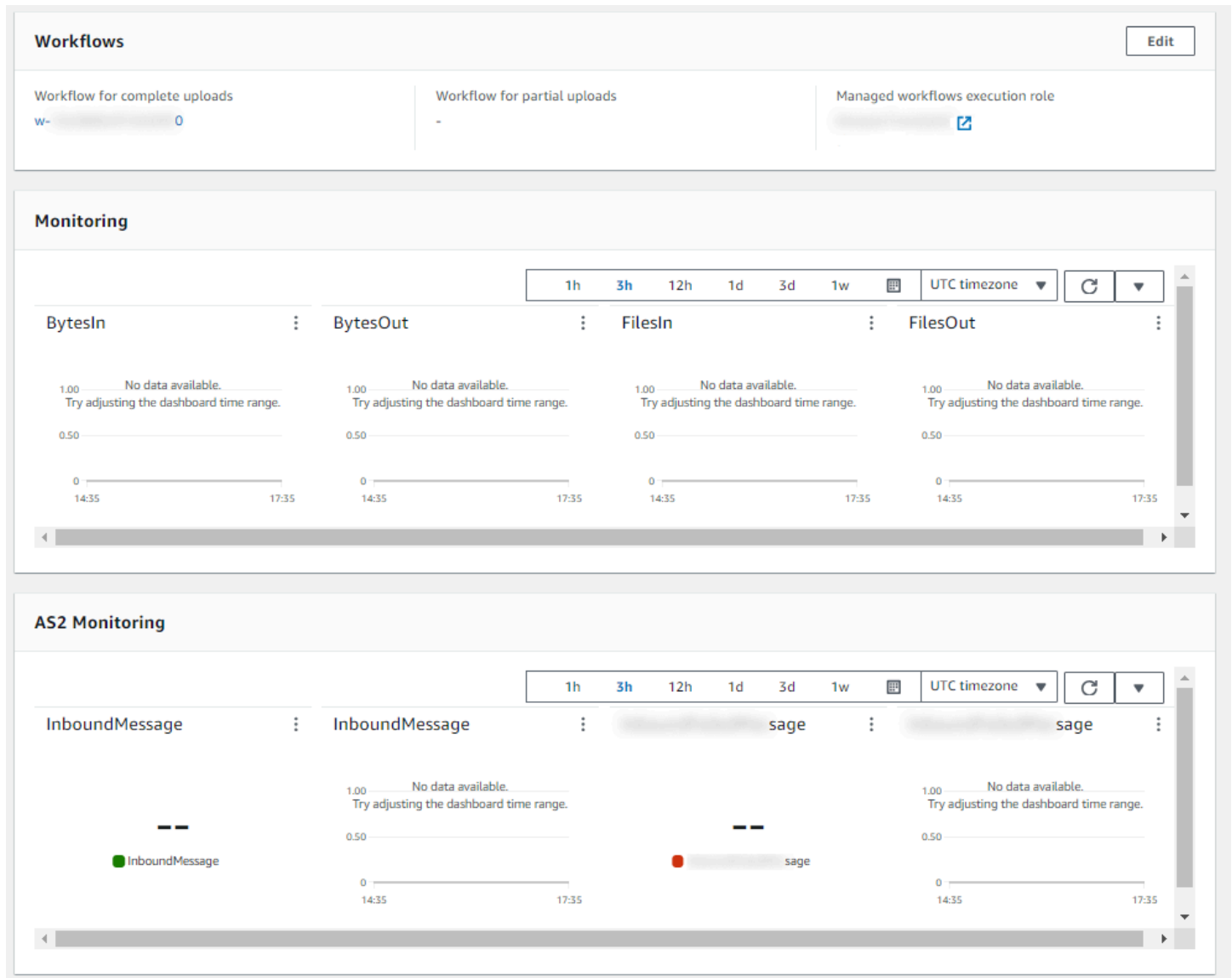
각 AS2 서버에는 세 개의 고정 IP 주소가 할당됩니다. 이러한 IP 주소를 사용하여 AS2를 통해 거래 파트너에게 비동기 MDNs 보냅니다.

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



AS2 서버 세부 정보 페이지의 하단에는 연결된 워크플로에 대한 세부 정보와 모니터링 및 태그 지정 정보가 포함되어 있습니다.



Transfer Family 서버에 대한 IPv6 지원

AWS Transfer Family 는 다음 리소스에 대해 듀얼 스택(IPv4 및 IPv6) 엔드포인트를 지원합니다.

- SFTP 퍼블릭 엔드포인트
- 모든 프로토콜에 대한 VPC 내부 엔드포인트(SFTP/FTPS/FTP 및 AS2)
- 에 제공된 단계를 사용하여 AS2-enabled Transfer Family 서버의 퍼블릭 엔드포인트 [듀얼 스택 AS2 서버 연결에 Application Load Balancer 사용](#)
- API 엔드포인트

듀얼 스택 지원을 통해 Transfer Family 엔드포인트는 IPv4 및 IPv6 지원 클라이언트와 모두 통신할 수 있습니다. 따라서 한 번에 모두 전환할 필요 없이 IPv4에서 IPv6 기반 시스템으로 점진적으로 전환할 수 있으며, IPv6 규정 준수 요구 사항을 충족하고, IPv4와 IPv6 간의 주소 변환을 처리하기 위해 비용이 많이 드는 네트워킹 장비가 필요하지 않습니다. 자세한 내용은 AWS Transfer Family API 참조의 [DNS 및 엔드포인트를 참조하세요](#). 사용 가능한 엔드포인트의 전체 목록은의 [AWS Transfer Family 엔드포인트 및 할당량을 참조하세요](#)AWS 일반 참조.

IPv6 제한 사항

다음 Transfer Family 리소스는 현재 IPv6를 지원하지 않습니다.

- VPC-인터넷 엔드포인트
- VPC_ENDPOINT 엔드포인트 유형(사용되지 않음)

FTPS 프로토콜은 파일 나열, 가져오기 및 넣기 작업을 위해 열린 데이터 포트를 요청하기 위한 PASV 및 EPSV 명령을 지원합니다. 그러나 PASV는 IPv4별 응답이 필요하므로 IPv6에서는 작동하지 않습니다. IPv4-specific EPSV는 포트 정보만 반환하므로 계속 작동합니다.

FTPS를 사용하려면 다음 중 하나를 사용하는 것이 좋습니다.

- EPSV를 사용하도록 FTPS 클라이언트 구성
- IPv6 대신 IPv4 사용 IPv6

SFTP는 IPv4와 IPv6를 모두 지원합니다. 듀얼 스택 엔드포인트로 작업할 때는 FTPS 대신 SFTP를 사용하는 것이 좋습니다.

서버용 IPv6 구성

새 서버를 생성하거나 기존 서버를 업데이트할 때 IP 주소 유형을 선택할 수 있습니다.

- IPv4(기본값): 이전 버전과의 호환성을 위해 서버는 IPv4 연결만 허용합니다.
- 듀얼 스택: 서버는 IPv4 및 IPv6 연결을 모두 허용합니다.

기존 서버의 IP 주소 유형을 업데이트하려면:

1. 서버를 중지합니다.
2. 엔드포인트 세부 정보를 편집합니다.
3. IP 주소 유형을 듀얼 스택으로 변경합니다.
4. 서버를 시작합니다.

Note

VPC-Internet 엔드포인트의 경우 듀얼 스택 모드는 현재 지원되지 않습니다.

듀얼 스택 AS2 서버 연결에 Application Load Balancer 사용

퍼블릭 엔드포인트가 있는 Application Load Balancer를 사용하여 AS2 서버에 대한 듀얼 스택(IPv4 및 IPv6) 연결을 활성화할 수 있습니다. 이를 통해 거래 파트너는 IPv4 또는 IPv6를 사용하여 AS2 서버에 연결할 수 있습니다.

AS2 서버용 듀얼 스택 Application Load Balancer를 설정하려면

1. 다음 설정을 사용하여 VPC를 생성합니다.
 - VPC 전용
 - 수동 IPv4 CIDR 입력
 - Amazon 제공 IPv6 CIDR 블록
2. 서로 다른 가용 영역에 서브넷을 두 개 이상 생성합니다.
 - 서브넷에 IPv6 CIDRs 추가
 - 서브넷을 생성할 때 VPC IPv4/IPv6 주소의 하위 집합만 할당하여 추가 서브넷에 사용할 수 있는 주소를 그대로 둡니다.

3. VPC용 인터넷 게이트웨이를 생성합니다.
4. 라우팅 테이블을 편집하고 두 개의 경로를 추가합니다.
 - 대상이 있는 경로 1개 0.0.0.0/0
 - 대상이 있는 경로 1개 ::/0
 - 두 라우팅 대상을 모두 생성한 인터넷 게이트웨이로 설정
5. 1단계에서 생성한 VPC에서 AS2-enabled 서버를 생성합니다. 를 IpAddressType로 지정해야 합니다DUALSTACK.

AS2 프로토콜을 사용하는 Transfer Family 서버를 생성하는 방법에 대한 자세한 내용은 섹션을 참조하세요 [AS2 서버 생성](#).

6. 대상 그룹 생성:
 - 그룹 세부 정보 지정에서 다음을 구성합니다.
 - 대상 유형: IP 주소
 - 이름: 이름 입력
 - Protocol: HTTP
 - 포트: 5080
 - VPC: 생성한 VPC 선택
 - 프로토콜 버전: HTTP1
 - 상태 확인: 기본값 사용
 - 대상 등록의 경우:
 - AS2 서버의 프라이빗 IPv4 주소 입력
 - 아래에서 보류 중으로 포함을 선택합니다.
7. Application Load Balancer 생성:
 - 이름 입력
 - 스키마에서 인터넷 연결을 선택합니다.
 - IP 주소 유형에서 듀얼 스택을 선택합니다.
 - 네트워크 매핑의 경우:
 - 생성한 VPC 선택
 - 서브넷을 생성한 가용 영역 선택

- 보안 그룹에서 포트 80의 모든 IP 주소에서 인바운드 IPv4 및 IPv6 트래픽을 허용하는 보안 그룹을 선택합니다.
- 리스너 및 라우팅의 경우:
 - Protocol: HTTP
 - Port: 80
 - 기본 작업: 생성한 대상 그룹에 전달
- 로드 밸런서 생성을 선택합니다.

Application Load Balancer를 생성한 후 거래 파트너는 DNS 이름을 사용하여 AS2 서버로 트래픽을 전송할 수 있습니다. 이 구성을 사용하면 AS2 서버가 듀얼 스택 Application Load Balancer를 통해 IPv4 및 IPv6 클라이언트 모두에서 연결을 수락할 수 있습니다.

서버 세부 정보 편집

AWS Transfer Family 서버를 생성한 후 서버 구성을 편집할 수 있습니다.

주제

- [File Transfer 프로토콜 편집](#)
- [서버 엔드포인트 편집](#)
- [로깅 구성 편집](#)
- [보안 정책 편집](#)
- [서버의 관리형 워크플로 변경](#)
- [서버의 디스플레이 배너 변경](#)
- [서버를 온라인이나 오프라인으로 전환](#)

서버의 구성을 편집하는 방법

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 서버를 선택합니다.
3. 서버 ID 옆에서 식별자를 선택하여 다음과 같은 서버 세부 정보 페이지를 표시합니다.

이 페이지에서 편집을 선택하여 서버의 속성을 변경할 수 있습니다.

- 프로토콜을 변경하려면 [File Transfer 프로토콜 편집](#)을 참조하세요.

- 자격 증명 공급자의 경우 이제 모든 자격 증명 공급자 유형(서비스 관리형, AWS 디렉터리 서비스 또는 사용자 지정 자격 증명 공급자) 간에 변경할 수 있습니다. 자격 증명 공급자 유형 변경 및 각 전환에 필요한 정보에 대한 자세한 내용은 [섹션을 참조하세요](#) [자격 증명 공급자 구성 편집](#).
- 엔드포인트 타입 또는 사용자 지정 호스트 이름을 변경하려면 [서버 엔드포인트 편집](#)를 참조하세요.
- 계약을 추가하려면 먼저 AS2를 프로토콜로 서버에 추가해야 합니다. 자세한 내용은 [File Transfer 프로토콜 편집](#)를 참조하세요.
- 서버의 호스트 키를 관리하려면 [SFTP 지원 서버의 호스트 키 관리](#)를 참조하세요.
- 추가 세부 정보에서 다음 정보를 편집할 수 있습니다.
 - 로깅 역할을 변경하려면 [로깅 구성 편집](#)를 참조하세요.
 - 보안 정책을 변경하려면 [보안 정책 편집](#)를 참조하세요.
 - 서버 호스트 키를 변경하려면 [SFTP 지원 서버의 호스트 키 관리](#)를 참조하세요.
 - 서버의 관리 워크플로를 변경하려면 [서버의 관리형 워크플로 변경](#)를 참조하세요.
 - 서버의 디스플레이 배너를 편집하려면 [서버의 디스플레이 배너 변경](#)를 참조하세요.
- 추가 구성 에서 다음 정보를 편집할 수 있습니다.
 - SetStat 옵션: 이 옵션을 사용하여 Amazon S3 버킷에 업로드하는 파일에 대해 클라이언트가 SETSTAT를 사용하려고 할 때 생성되는 오류를 무시합니다. 자세한 설명은 [ProtocolDetails](#) 주제의 SetStatOption 설명서를 참조하세요.
 - TLS 세션 재개: FTPS 세션에 대한 통제 및 데이터 연결 간에 협상된 암호 키를 재개하거나 공유하는 메커니즘을 제공합니다. 자세한 설명은 [ProtocolDetails](#) 주제의 TlsSessionResumptionMode 설명서를 참조하세요.
 - 수동 IP: FTP 및 FTPS 프로토콜에 대한 수동 모드를 나타냅니다. 방화벽, 라우터 또는 로드 밸런서의 퍼블릭 IP 주소와 같은 단일 IPv4 주소를 입력합니다. 자세한 설명은 [ProtocolDetails](#) 주제의 PassiveIp 설명서를 참조하세요.

Note

Network Load Balancer(NLBs) 또는 NAT 게이트웨이를 AWS Transfer Family 서버 앞에 배치하지 마십시오. 이 구성은 비용을 늘리고 성능 문제를 일으킬 수 있습니다. 자세한 내용은 [NLBs 및 NATs 앞에 배치하지 마세요. AWS Transfer Family](#) 섹션을 참조하세요.

- 서버를 시작 또는 중지하려면 [서버를 온라인이나 오프라인으로 전환](#)를 참조하세요.
- 서버를 삭제하려면 [서버 삭제](#)를 참조하세요.

- 사용자 속성을 편집하려면 [액세스 통제 관리](#)를 참조하세요.

Protocols

[Edit](#)

Protocols over which clients can connect to your server's endpoint

- SFTP

Identity provider

[Edit](#)

Identity provider type
Service managed

Note

서버 호스트 키 설명 및 가져온 날짜 값은 2022년 9월 기준으로 새로 추가되었습니다. 이러한 값은 다중 호스트 키 기능을 지원하기 위해 도입되었습니다. 이 기능을 사용하려면 여러 호스트 키가 도입되기 전에 사용 중이던 단일 호스트 키를 모두 마이그레이션해야 합니다.

마이그레이션된 서버 호스트 키의 가져온 날짜 값은 서버의 마지막 수정 날짜로 설정됩니다. 즉, 마이그레이션된 호스트 키로 표시되는 날짜는 어떤 식으로든 서버 호스트 키 마이그레이션 전에 서버를 마지막으로 수정한 날짜와 일치합니다.

마이그레이션된 유일한 키는 가장 오래된 또는 유일한 서버 호스트 키입니다. 모든 추가 키에는 가져온 시점의 실제 날짜가 표시됩니다. 또한 마이그레이션된 키에는 마이그레이션된 키임을 쉽게 식별할 수 있는 설명이 있습니다.

마이그레이션은 9월 2일에서 9월 13일 사이에 이루어졌습니다. 이 범위 내의 실제 마이그레이션 날짜는 서버 리전에 따라 다릅니다.

Additional details

[Edit](#)

Log group

/aws/transfer/s-

Logging role

-

Security Policy

TransferSecurityPolicy-PQ-SSH-Experimental-2023-04

Domain

Amazon S3

Login display banner

[View the display message](#)

Optimized directories

ENABLED

SetStat option

Ignore

TLS session resumption

-

Passive IP

-

File Transfer 프로토콜 편집

AWS Transfer Family 콘솔에서 파일 전송 프로토콜을 편집할 수 있습니다. File Transfer 프로토콜은 클라이언트를 서버의 엔드포인트에 연결합니다.

프로토콜을 편집하려면

1. 서버 세부 정보 페이지에서 프로토콜 옆의 편집을 선택합니다.
2. 프로토콜 편집 페이지에서 프로토콜 확인란 또는 확인란을 선택하거나 선택 취소하여 다음 파일 전송 프로토콜을 추가하거나 제거합니다.

- Secure Shell(SSH) File Transfer 프로토콜(SFTP): SSH를 통한 파일 전송

SFTP에 대한 자세한 내용은 [SFTP 지원 서버 생성](#)를 참조하세요.

- FTPS(File Transfer 프로토콜 보안) - TLS 암호화를 사용한 파일 전송

FTP에 대한 자세한 내용은 [FTPS 지원 서버 생성](#)를 참조하세요.

- FTP(File Transfer 프로토콜) - 암호화되지 않은 파일 전송

FTPS에 대한 자세한 내용은 [FTP 지원 서버 생성](#)를 참조하세요.

Note

기존 서버에서 SFTP만 사용하도록 설정한 상태에서 FTPS 및 FTP를 추가하려는 경우 FTPS 및 FTP와 호환되는 올바른 ID 제공자 및 엔드포인트 타입 설정이 있는지 확인해야 합니다.

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](#)

FTPS를 선택하는 경우 클라이언트가 FTPS를 통해 연결할 때 서버를 식별하는 데 사용할 AWS Certificate Manager (ACM)에 저장된 인증서를 선택해야 합니다.

새 퍼블릭 인증서를 요청하려면 AWS Certificate Manager 사용 설명서의 [퍼블릭 인증서 요청](#)을 참조하세요.

기존 인증서를 ACM으로 가져오려면 AWS Certificate Manager 사용 설명서의 [ACM으로 인증서 가져오기](#)를 참조하세요.

프라이빗 IP 주소를 통해 FTPS를 사용하도록 프라이빗 인증서를 요청하려면 AWS Certificate Manager 사용 설명서의 [프라이빗 인증서 요청](#)을 참조하세요.

다음 암호화 알고리즘 및 키 크기를 사용하는 인증서가 지원됩니다:

- 2048비트 RSA(RSA_2048)
- 4096비트 RSA(RSA_4096)
- 타원 프라임 곡선 256비트(EC_prime256v1)
- 타원 프라임 곡선 384 비트(EC_secp384r1)
- 타원 프라임 곡선 521비트(EC_secp521r1)

Note

인증서는 FQDN 또는 IP 주소가 지정된 유효한 SSL/TLS X.509 버전 3 인증서여야 하며 발급자에 대한 정보가 포함되어야 합니다.

3. 저장을 선택합니다. 서버 세부 정보 페이지로 돌아갑니다.

서버 엔드포인트 편집

AWS Transfer Family 콘솔에서 서버 엔드포인트 유형과 사용자 지정 호스트 이름을 수정할 수 있습니다. 또한 VPC 엔드포인트의 경우 가용 영역 정보를 편집할 수 있습니다.

서버 엔드포인트 세부 정보를 편집하려면

1. 서버 세부 정보 페이지에서 엔드포인트 세부 정보 옆의 편집을 선택합니다.
2. 엔드포인트 유형을 편집하려면 먼저 서버를 중지해야 합니다. 그런 다음 엔드포인트 구성 편집 페이지의 엔드포인트 유형에서 다음 값 중 하나를 선택할 수 있습니다.
 - 퍼블릭 - 이 옵션을 사용하면 인터넷을 통해 서버에 액세스할 수 있습니다.
 - VPC - 이 옵션을 사용하면 VPC(VPC)의 서버에 액세스할 수 있습니다. VPC에 대한 자세한 내용은 [Virtual Private Cloud\(VPC\)에 서버 생성](#)을 참조하세요.
3. 사용자 지정 호스트 이름의 경우 다음 중 하나를 선택합니다.
 - 없음 - 사용자 지정 도메인을 사용하지 않으려면 없음을 선택합니다.

에서 제공하는 서버 호스트 이름을 가져옵니다 AWS Transfer Family. 서버 호스트 이름은 `serverId.server.transfer.regionId.amazonaws.com` 형식을 취합니다.

- Amazon Route 53 DNS 별칭 - Route 53에서 자동으로 생성된 DNS 별칭을 사용하려면 이 옵션을 선택합니다.
- 기타 DNS - 외부 DNS 서비스에서 이미 소유하고 있는 호스트 이름을 사용하려면 기타 DNS를 선택합니다.

Amazon Route 53 DNS 별칭 또는 기타 DNS를 선택하면 서버의 엔드포인트와 연결할 이름 확인 방법이 지정됩니다.

예를 들어 사용자 지정 도메인이 `sftp.inbox.example.com`일 수 있습니다. 사용자 지정 호스트 이름은 사용자가 입력했고 DNS 서비스가 해석할 수 있는 DNS 이름을 사용합니다. Route 53을 DNS 해석기로 사용하거나, 자체 DNS 서비스 공급자를 사용할 수 있습니다. AWS Transfer Family가 Route 53을 이용해 트래픽을 사용자 지정 도메인에서 SFTP 엔드포인트로 라우팅하는 방법은 [사용자 지정 호스트 이름으로 작업](#)을 참조하세요.

Transfer Family > Servers > s- > Edit endpoint configuration

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

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

☐ FIPS Enabled endpoint

[Cancel](#) [Save](#)

4. VPC 엔드포인트의 경우 가용 영역 창에서 정보를 변경할 수 있습니다.
5. 저장을 선택합니다. 서버 세부 정보 페이지로 돌아갑니다.

로깅 구성 편집

AWS Transfer Family 콘솔에서 로깅 구성을 변경할 수 있습니다.

Note

서버를 생성할 때 Transfer Family가 CloudWatch 로깅 IAM 역할을 생성한 경우 IAM 역할을 `AWSTransferLoggingAccess(이)`라고 합니다. 모든 Transfer Family 서버에 사용할 수 있습니다.

로깅 구성 편집

1. 서버 세부 정보 페이지에서 추가 세부 정보 옆의 편집을 선택합니다.
2. 구성에 따라 로깅 역할, 정형 JSON 로깅 또는 둘 다 선택할 수 있습니다. 자세한 내용은 [서버의 로깅 업데이트](#)를 참조하세요.

보안 정책 편집

이 절차에서는 AWS Transfer Family 콘솔 또는를 사용하여 Transfer Family 서버의 보안 정책을 변경하는 방법을 설명합니다 AWS CLI.

Note

엔드포인트가 FIPS를 사용하는 경우 FIPS 보안 정책을 비 FIPS 보안 정책으로 변경할 수 없습니다.

Console

콘솔을 사용하여 보안 정책을 편집하려면

1. 서버 세부 정보 페이지에서 추가 세부 정보 옆의 편집을 선택합니다.
2. 암호화 알고리즘 옵션의 경우 서버에서 사용하도록 설정된 암호화 알고리즘이 포함된 보안 정책을 선택합니다.

보안 정책에 대한 자세한 설명은 섹션을 참조하세요 [AWS Transfer Family 서버에 대한 보안 정책](#).

3. 저장을 선택합니다.

업데이트된 보안 정책을 볼 수 있는 서버 세부 정보 페이지로 돌아갑니다.

AWS CLI

CLI를 사용하여 보안 정책을 편집하려면

1. 다음 명령을 실행하여 서버에 연결된 현재 보안 정책을 확인합니다.

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

이 describe-server 명령은 다음 줄을 포함하여 서버의 모든 세부 정보를 반환합니다.

```
"SecurityPolicyName": "TransferSecurityPolicy-2018-11"
```

이 경우 서버의 보안 정책은 입니다 TransferSecurityPolicy-2018-11.

- 명령에 보안 정책의 정확한 이름을 제공해야 합니다. 예를 들어 다음 명령을 실행하여 서버를 업데이트합니다TransferSecurityPolicy-2023-05.

```
aws transfer update-server --server-id your-server-id --security-policy-name
"TransferSecurityPolicy-2023-05"
```

Note

사용 가능한 보안 정책의 이름은에 나열되어 있습니다[AWS Transfer Family 서버에 대한 보안 정책](#).

성공하면 명령이 다음 코드를 반환하고 서버의 보안 정책을 업데이트합니다.

```
{
  "ServerId": "your-server-id"
}
```

서버의 관리형 워크플로 변경

AWS Transfer Family 콘솔에서 서버와 연결된 관리형 워크플로를 변경할 수 있습니다.

관리형 워크플로 변경

- 서버 세부 정보 페이지에서 추가 세부 정보 옆의 편집을 선택합니다.
- 추가 세부 정보 편집 페이지의 관리형 워크플로 섹션에서 모든 업로드에서 실행할 워크플로를 선택합니다.

Note

워크플로가 아직 없는 경우 새 워크플로 생성을 선택하여 워크플로를 생성합니다.

- 사용할 워크플로 ID를 선택합니다.
- 실행 역할을 선택합니다. 이 역할은 워크플로의 단계를 실행할 때 Transfer Family가 맡는 역할입니다. 자세한 내용은 [워크플로에 대한 IAM 정책](#)를 참조하세요. 저장을 선택합니다.

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]

3. 저장을 선택합니다. 서버 세부 정보 페이지로 돌아갑니다.

서버의 디스플레이 배너 변경

AWS Transfer Family 콘솔에서 서버와 연결된 디스플레이 배너를 변경할 수 있습니다.

디스플레이 배너를 변경하려면

1. 서버 세부 정보 페이지에서 추가 세부 정보 옆의 편집을 선택합니다.
2. 추가 세부정보 편집 페이지의 디스플레이 배너 섹션에서 사용 가능한 디스플레이 배너의 텍스트를 입력합니다.
3. 저장을 선택합니다. 서버 세부 정보 페이지로 돌아갑니다.

서버를 온라인이나 오프라인으로 전환

AWS Transfer Family 콘솔에서 서버를 온라인 상태로 전환하거나 오프라인으로 전환할 수 있습니다.

서버를 온라인 상태로 전환하려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 탐색 창에서 서버를 선택합니다.
3. 오프라인 상태인 서버의 확인란을 선택합니다.
4. 작업에서 시작을 선택합니다.

서버가 오프라인에서 온라인으로 전환되는 데는 몇 분 정도 걸릴 수 있습니다.

Note

오프라인으로 전환하기 위해 서버를 중단해도, 해당 서버에 대한 서비스 요금은 계속 발생합니다. 추가 서버 관련 요금을 방지하려면, 해당 서버를 삭제해야 합니다.

서버를 오프라인 상태로 전환하려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 탐색 창에서 서버를 선택합니다.
3. 온라인 상태인 서버의 확인란을 선택합니다.
4. 작업에서 중지를 선택합니다.

서버가 시작 또는 종료 중일 때는 서버가 파일 작업을 수행할 수 없습니다. 콘솔은 시작 및 중단 상태를 표시하지 않습니다.

오류 조건 START_FAILED 또는가 발견되면 AWS Support 에 STOP_FAILED 문의하여 문제를 해결하세요.

자격 증명 공급자 구성 편집

서버의 자격 증명 공급자 유형을 모든 유형에서 다른 유형으로 변경할 수 있습니다. 사용 가능한 자격 증명 공급자 유형은 다음과 같습니다.

- 서비스 관리형 - 서비스 내에 사용자 자격 증명 저장
- AWS 디렉터리 서비스 - Entra ID 도메인 서비스에 AWS 관리형 Microsoft AD 또는 AWS 디렉터리 서비스 사용
- 사용자 지정 - Lambda 함수 또는 Amazon API Gateway를 사용하여 기존 자격 증명 공급자와 통합

자격 증명 공급자 유형을 변경할 때는 전환에 따라 특정 정보를 제공해야 합니다. 다음 섹션에서는 각 변경 유형에 필요한 정보를 설명합니다.

Important

자격 증명 공급자 변경 시 고려 사항:

- 사용자 마이그레이션 - 자격 증명 공급자 유형을 변경할 때 기존 사용자 구성은 자동으로 마이그레이션되지 않습니다. 새 자격 증명 공급자 시스템에서 사용자를 설정해야 합니다.
- 테스트 - 프로덕션 환경에서 변경하기 전에 새 자격 증명 공급자 구성을 철저히 테스트합니다.
- 권한 - 변경하기 전에 새 자격 증명 공급자에 필요한 IAM 권한 및 역할이 구성되어 있는지 확인합니다.

서비스 관리형 자격 증명 공급자로 변경

다른 자격 증명 공급자 유형에서 서비스 관리형으로 변경할 때는 다음을 수행해야 합니다.

- ID 제공업체 유형으로 관리형 서비스 선택
- 다른 자격 증명 공급자의 기존 사용자 구성은 전송되지 않으므로 변경이 완료된 AWS Transfer Family 후에서 직접 새 사용자를 생성합니다.

예: 사용자 지정 자격 증명 공급자에서 서비스 관리형으로 변경하는 경우 서비스 내에서 AWS Transfer Family 모든 사용자 계정 및 관련 권한을 다시 생성해야 합니다.

AWS 디렉터리 서비스로 변경

다른 자격 증명 공급자 유형에서 AWS 디렉터리 서비스로 변경할 때는 다음을 제공해야 합니다.

- 디렉터리 - 기존 AWS Managed Microsoft AD 또는 AWS Directory Service for Entra ID Domain Services 디렉터리 선택
- 액세스 - 특정 그룹에 대한 액세스를 제한할지 또는 디렉터리의 모든 사용자에게 대한 액세스를 허용할지 선택합니다.
- 액세스 역할 -가 디렉터리에 액세스할 수 있도록 허용하는 AWS Transfer Family IAM 역할

예: 서비스 관리형에서 AWS 디렉터리 서비스로 변경하는 경우 기존 d-1234567890 디렉터리를 선택하고 TransferUsers 그룹에 대한 액세스를 제한하도록 선택한 다음 TransferDirectoryAccessRole IAM 역할을 지정합니다.

사용자 지정 자격 증명 공급자로 변경

다른 자격 증명 공급자 유형에서 사용자 지정 자격 증명 공급자로 변경하는 경우 Lambda 함수 또는 Amazon API Gateway 중에서 선택하고 필요한 구성을 제공해야 합니다.

Lambda 함수 사용

Lambda 함수 통합의 경우 다음을 제공합니다.

- 함수 - 인증을 처리하는 기존 Lambda 함수 선택
- 인증 방법(SFTP 프로토콜의 경우) - 암호, 퍼블릭 키 또는 둘 다를 선택합니다.

예: AWS 디렉터리 서비스에서 사용자 지정 Lambda 자격 증명 공급자로 변경하는 경우 TransferCustomAuth 함수를 선택하고 인증 방법으로 암호를 선택합니다.

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

Amazon API Gateway 사용

Amazon API Gateway 통합의 경우 다음을 제공합니다.

- API Gateway URL - API Gateway 엔드포인트의 호출 URL입니다.

- 호출 역할 -가 API Gateway를 호출 AWS Transfer Family 하도록 허용하는 IAM 역할
- 인증 방법(SFTP 프로토콜의 경우) - 암호, 퍼블릭 키 또는 둘 다를 선택합니다.

예: 서비스 관리형에서 API Gateway로 변경하는 경우 URL을 제공하고https://
abcdef123.execute-api.us-east-1.amazonaws.com/prod,
TransferApiGatewayInvocationRole IAM 역할을 지정하고, 인증 방법으로 퍼블릭 키를 선택합
니다.

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

[Refresh](#)

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

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

Cancel Save

Amazon API Gateway에서 Lambda 함수로 변경

사용자 지정 ID 제공업체 통합을 위해 Amazon API Gateway에서 Lambda 함수로 일반적인 전환이 변경되고 있습니다. 이 변경 사항을 통해 동일한 인증 로직을 유지하면서 아키텍처를 간소화할 수 있습니다.

이 전환의 주요 고려 사항:

- 동일한 함수, 다른 권한 - API Gateway와 직접 Lambda 통합 모두에 동일한 Lambda 함수를 사용할 수 있지만 리소스 정책을 업데이트해야 합니다.
- 리소스 정책 요구 사항 - 직접 Lambda 통합으로 변경할 때 함수의 리소스 정책은 외에도 함수를 호출할 수 있는 `transfer.amazonaws.com` 권한을 부여해야 합니다 `apigateway.amazonaws.com`.

이 변경을 수행하려면

1. 가 함수를 `transfer.amazonaws.com` 호출할 수 있도록 Lambda 함수의 리소스 정책을 업데이트합니다.
2. AWS Transfer Family 콘솔에서 자격 증명 공급자를 API Gateway에서 Lambda 함수로 변경합니다.
3. 기존 Lambda 함수를 선택합니다.
4. 구성을 테스트하여 인증이 올바르게 작동하는지 확인합니다.

직접 Lambda 통합을 위한 리소스 정책 예제:

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Principal": {
      "Service": [
        "transfer.amazonaws.com",
        "apigateway.amazonaws.com"
      ]
    },
    "Action": "lambda:InvokeFunction",
    "Resource": "arn:aws:lambda:us-east-1:123456789012:function:function-name"
  }]
}
```

자격 증명 공급자 전환 중 사용자 보존

자격 증명 공급자 유형 간에 변경할 때 문제가 발생할 경우 효율적인 롤백을 위해 특정 시나리오에서 기존 사용자 구성이 보존됩니다.

- 서비스 관리형에서 사용자 지정 자격 증명 공급자로 및 뒤로 - 서비스 관리형에서 사용자 지정 자격 증명 공급자로 변경한 다음 다시 서비스 관리형으로 변경하면 모든 사용자가 마지막으로 알려진 구성으로 보존됩니다.
- AWS Directory Service 사용자 지정 자격 증명 공급자 및 뒤로 -에서 사용자 지정 자격 증명 공급자 AWS Directory Service 로 변경한 다음 다시 로 변경하면 위임된 액세스 그룹에 대한 AWS Directory Service 모든 정의가 마지막으로 알려진 구성에 유지됩니다.

이러한 보존 동작을 통해 사용자 지정 ID 제공업체 구성을 안전하게 테스트하고 사용자 액세스 구성을 잃지 않고 이전 설정으로 롤백할 수 있습니다.

자격 증명 공급자 변경 시 중요 고려 사항

- 사용자 마이그레이션 - 자격 증명 공급자 유형을 변경할 때 기존 사용자 구성은 자동으로 마이그레이션되지 않습니다. 새 자격 증명 공급자 시스템에서 사용자를 설정해야 합니다.
- 테스트 - 프로덕션 환경에서 변경하기 전에 새 자격 증명 공급자 구성을 철저히 테스트합니다.
- 권한 - 변경하기 전에 새 자격 증명 공급자에 필요한 IAM 권한 및 역할이 구성되어 있는지 확인합니다.

SFTP 지원 서버의 호스트 키 관리

서버 호스트 키는 Transfer Family 서버에서 호출자에게 고유한 자격 증명을 제공하고 올바른 서버임을 보장하기 위해 사용하는 프라이빗 키입니다. 이 보장은 호출자의 `known_hosts` 파일에 올바른 퍼블릭 키가 있으면 적용됩니다. (이 `known_hosts` 파일은 대부분의 SSH 클라이언트가 연결한 서버의 퍼블릭 키를 저장하는 데 사용하는 표준 기능입니다.) 서버에 `ssh-keyscan` 대해를 실행하여 서버 호스트 키에 해당하는 퍼블릭 키를 검색할 수 있습니다.

Important

실수로 서버의 호스트 키를 변경하면 작업에 영향을 줄 수 있습니다. SFTP 클라이언트의 구성 방식에 따라 신뢰할 수 있는 호스트 키가 없다는 메시지와 함께 SFTP 클라이언트가 즉시 실패

하거나 위협적인 메시지가 표시될 수 있습니다. 연결을 자동화하는 스크립트가 있는 경우 이 스크립트도 실패할 가능성이 높습니다.

기본적으로는 SFTP 지원 서버의 호스트 키를 AWS Transfer Family 생성합니다. 서버 호스트 키를 가져와 호스트 자격 증명을 유지하고 클라이언트 트러스트 스토어를 업데이트하지 않도록 할 수 있습니다. 이 작업을 수행할 수 있는 몇 가지 이유를 [호스트 키를 가져와야 하는 경우](#) 나열합니다. 호스트 키를 제공하지 않으면 새 키가 생성됩니다.

AWS Transfer Family 는 다양한 유형의 여러 호스트 키(RSA, ECDSA 및 ED25519)를 지원하여 광범위한 클라이언트 호스트 서명 알고리즘과의 호환성을 제공합니다. 다양한 키 유형이 특정 알고리즘을 활성화합니다. RSA 키는 rsa-* 알고리즘을 활성화하고, ECDSA 키는 ecdsa-* 알고리즘을 활성화하며, ED25519 키는 ed25519 알고리즘을 활성화합니다. 클라이언트가 서버와 상호 작용을 시작한 후 추가 키 유형을 도입하면 일부 클라이언트가 중단될 수 있고 기존 호스트 키를 교체하는 것만큼 문제가 될 수 있으므로 서버 생성 시 키 유형을 계획합니다.

사용자가 SFTP 지원 서버의 신뢰성 확인 메시지를 받지 않으려면 온프레미스 서버의 호스트 키를 SFTP 지원 서버로 가져옵니다. 이렇게 하면 잠재적인 중간자 공격에 대한 경고 메시지가 사용자에게 표시되지 않습니다.

추가 보안 조치로 호스트 키를 주기적으로 교체할 수도 있습니다. 자세한 내용은 [서버 호스트 키 교체](#)를 참조하세요.

Note

서버 호스트 키는 SFTP 프로토콜을 지원하는 서버에서 사용됩니다.

호스트 키를 가져와야 하는 경우

호스트 키를 자동으로 생성할 AWS Transfer Family 수 있지만 자체 호스트 키를 가져오는 것이 운영상의 이점을 제공하는 몇 가지 시나리오가 있습니다.

- 서버 마이그레이션 - 기존 서버에서 로 마이그레이션 AWS Transfer Family 중이며 기존 클라이언트에 대한 클라이언트 트러스트 스토어(known_hosts 파일)를 업데이트하지 않으려고 합니다.
- 재해 복구 및 장애 조치 - 동일한 퍼블릭 DNS 이름을 공유하는 여러 AWS Transfer Family 서버(예: 미국 동부(오하이오) 및 미국 서부(오레곤))가 있습니다. 두 서버에서 동일한 호스트 키를 사용하면 클라이언트 인증 실패 없이 원활한 장애 조치가 가능합니다.

- 운영 연속성 - 향후 다른 서버(AWS Transfer Family 또는 기타 서버)와 함께 사용할 수 있는 호스트 키 구성 요소를 사용하여 인프라 전체에서 일관된 서버 ID를 유지하려고 합니다.
- 알고리즘 제어 - 더 많은 호스트 키 알고리즘을 제공하여 클라이언트 호환성을 높이거나 클라이언트가 특정 알고리즘과 호환되는 키만 제공하여 사용할 수 있는 알고리즘을 제어하려는 경우.

다음 주제에서는 서버 호스트 키를 관리하기 위한 자세한 절차를 제공합니다.

- [추가 서버 호스트 키 추가](#) - 서버에 호스트 키 추가
- [서버 호스트 키 삭제](#) - 서버에서 호스트 키 제거
- [서버 호스트 키 교체](#) - 보안 강화를 위해 호스트 키 교체
- [추가 서버 호스트 키 정보](#) - 호스트 키 세부 정보 보기 및 관리

추가 서버 호스트 키 추가

AWS Transfer Family 콘솔에서 서버 호스트 키를 추가할 수 있습니다. 형식이 다른 호스트 키를 추가하면 클라이언트가 서버에 연결할 때 서버를 식별하고 보안 프로필을 개선하는 데 유용할 수 있습니다. 예를 들어 원래 키가 RSA 키인 경우 ECDSA 키를 추가할 수 있습니다.

Note

SFTP 클라이언트는 키의 알고리즘과 일치하는 구성에서 가장 오래된 키를 사용하여 연결합니다. 각 키 유형(RSA, ECDSA 또는 ED25519)의 가장 오래된 키는 해당 유형의 서버에 대한 활성 키입니다.

Transfer Family 서버에 여러 유형의 호스트 키가 있는 경우의 보안 참고 사항

서버에 여러 유형의 호스트 키가 있는 경우 SFTP 클라이언트는 유형별로 기본 설정을 할당할 수 있습니다. 따라서 서버에 대한 RSA, ECDSA 및 ED25519 호스트 키가 있는 경우 선택은 유형별 기본 설정에 따라 결정됩니다.

최신 SFTP 클라이언트는 존재하는 경우 ECDSA 및 ED25519 호스트 키를 선호합니다. 이는 이전에 서버에 RSA 키만 있는 경우 ECDSA 또는 ED25519 키를 추가하려는 경우에 중요합니다. 새 ECDSA 또는 ED25519 키를 추가하면 클라이언트에 대한 보안 경고로 나타날 수 있습니다.

클라이언트에게 실제로 변경되지 않은 경우 기존 RSA 키 외에 새 키가 추가된 경우 키가 변경된 것으로 표시됩니다. 새 유형의 서버 호스트 키를 추가하기로 결정한 경우이 점에 유의하세요.

추가 서버 호스트 키 추가

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 서버를 선택한 다음 SFTP 프로토콜을 사용하는 서버를 선택합니다.
3. 서버 세부 정보 페이지에서 서버 호스트 키 섹션까지 아래로 스크롤합니다.

Server host keys (2)

Find resources

☐	Name	Host key ID	Fingerprint	Description	Key type	Date imported
☐	Default host key	hostkey-	SHA256:	Default	ssh-rsa	2023-06-30
☐	ecdsa-521	hostkey-	SHA256:	ECDSA host key	ecdsa-sha2-nistp521	2024-06-13

4. 호스트 키 추가를 선택합니다.

서버 호스트 키 추가 페이지가 표시됩니다.

5. (옵션) 서버 호스트 키로는 클라이언트가 SFTP 지원 서버를 통해 서버에 연결할 때 서버를 식별하는 데 사용할 RSA, ED25519 또는 ECDSA 프라이빗 키를 입력합니다.

Note

서버 호스트 키를 생성할 때는 반드시 -N ""를 지정해야 합니다(암호 없음). 키 쌍을 생성하는 방법에 대한 자세한 내용은 [macOS, Linux 또는 Unix에서 SSH 키 생성](#)을 참조하세요.

6. (옵션) 설명을 추가하여 여러 서버 호스트 키를 구분합니다. 키에 태그를 추가할 수도 있습니다.
7. 키 추가를 선택합니다. 서버 세부 정보 페이지로 돌아갑니다.

AWS Command Line Interface (AWS CLI)를 사용하여 호스트 키를 추가하려면 [ImportHostKey](#) API 작업을 사용하고 새 호스트 키를 제공합니다. 새 SFTP 지원 서버를 생성하는 경우 [CreateServer](#) API 작업에서 호스트 키를 파라미터로 제공합니다. AWS CLI 를 사용하여 기존 호스트 키에 대한 설명을 업데이트할 수도 있습니다.

다음 예제 import-host-key AWS CLI 명령은 지정된 SFTP 지원 서버의 호스트 키를 가져옵니다.

```
aws transfer import-host-key --description key-description --server-id your-server-id
--host-key-body file://my-host-key
```

서버 호스트 키 삭제

AWS Transfer Family 콘솔에서 서버 호스트 키를 삭제할 수 있습니다.

서버 호스트 키를 삭제하려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 서버를 선택한 다음 SFTP 프로토콜을 사용하는 서버를 선택합니다.
3. 서버 세부 정보 페이지에서 서버 호스트 키 섹션까지 아래로 스크롤합니다.

Server host keys (2)							Actions ▾	Add host key
Find resources							< 1 >	
☐	Name ↗	Host key ID ▾	Fingerprint Info ▾	Description ▾	Key type ▾	Date imported ▾		
☐	Default host key	hostkey-	SHA256:/	Default	ssh-rsa	2023-06-30		
☐	ecdsa-521	hostkey-	SHA256:	ECDSA host key	ecdsa-sha2-nistp521	2024-06-13		

4. 서버 호스트 키 섹션에서 키를 선택한 다음 작업에서 삭제를 선택합니다.
5. 표시되는 확인 대화 상자에 단어 **delete**을 입력한 다음 삭제를 선택한 다음 삭제를 선택합니다.

서버 페이지에서 호스트 키가 삭제됩니다.

를 사용하여 호스트 키를 삭제하려면 [DeleteHostKey](#) API 작업을 AWS CLI사용하고 서버 ID와 호스트 키 ID를 제공합니다.

다음 예제 delete-host-key AWS CLI 명령은 지정된 SFTP 지원 서버의 호스트 키를 삭제합니다.

```
aws transfer delete-host-key --server-id your-server-id --host-key-id your-host-key-id
```

서버 호스트 키 교체

서버 호스트 키를 주기적으로 교체할 수 있습니다. 이 주제에서는 서버에서 적용할 키를 선택하는 방법과 이러한 키를 교체하는 절차를 설명합니다.

클라이언트가 서버 호스트 키를 선택하는 방법

Transfer Family가 적용할 서버 키를 선택하는 방법은 여기에 설명된 대로 SFTP 클라이언트의 조건에 따라 달라집니다. 하나의 이전 키와 하나의 최신 키가 있다고 가정합니다.

- SFTP 클라이언트에는 서버에 대한 이전 퍼블릭 호스트 키가 없습니다. 클라이언트가 서버에 처음 연결할 때 다음 중 하나가 발생합니다.

- 클라이언트가 연결에 실패하도록 구성된 경우 연결이 실패합니다.
- 또는 클라이언트는 사용 가능한 알고리즘과 일치하는 첫 번째 키를 선택하고 사용자에게 해당 키를 신뢰할 수 있는지 묻습니다. 이 경우 클라이언트는 `known_hosts` 파일(또는 클라이언트가 신뢰 결정을 기록하는 데 사용하는 로컬 구성 파일 또는 리소스)을 자동으로 업데이트하고 해당 키를 입력합니다.
- SFTP 클라이언트의 `known_hosts` 파일에 이전 키가 있습니다. 이 키의 알고리즘 또는 다른 알고리즘에 최신 키가 있더라도 클라이언트는 이 키를 사용하는 것을 선호합니다. 이는 클라이언트가 `known_hosts` 파일에 있는 키에 대해 더 높은 수준의 신뢰를 갖기 때문입니다.
- SFTP 클라이언트의 키 파일에 새 `known_hosts` 키(사용 가능한 모든 알고리즘)가 있습니다. 클라이언트는 이전 키는 신뢰할 수 없으므로 무시하고 새 키를 사용합니다.
- SFTP 클라이언트에는 `known_hosts` 파일에 두 키가 모두 있습니다. 클라이언트는 서버에서 제공하는 사용 가능한 키 목록과 일치하는 인덱스별 첫 번째 키를 선택합니다.

Transfer Family는 Transfer Family 서버에 연결할 때 유연성을 극대화할 수 있으므로 SFTP 클라이언트의 `known_hosts` 파일에 모든 키가 있는 것을 선호합니다. 키 교체는 동일한 Transfer Family 서버의 `known_hosts` 파일에 여러 항목이 존재할 수 있다는 사실을 기반으로 합니다.

서버 호스트 키 절차 교체

예를 들어 Transfer Family 서버에 다음 서버 호스트 키 세트를 추가했다고 가정합니다.

서버 호스트 키

호스트 키 타입	서버에 추가된 날짜
RSA	2020년 4월 1일
ECDSA	2020년 2월 1일
ED25519	2019년 12월 1일
RSA	2019년 10월 1일
ECDSA	2019년 6월 1일
ED25519	2019년 3월 1일

서버 호스트 키를 교체하려면

1. 새 서버 호스트 키를 추가합니다. 이 절차는 [추가 서버 호스트 키 추가](#)에 설명되어 있습니다.
2. 이전에 추가한 것과 같은 타입의 호스트 키를 하나 이상 삭제합니다. 이 절차는 [서버 호스트 키 삭제](#)에 설명되어 있습니다.
3. 에서 앞서 설명한 동작에 따라 모든 키가 표시되며 활성화될 수 있습니다. [클라이언트가 서버 호스트 키를 선택하는 방법](#).

추가 서버 호스트 키 정보

호스트 키를 선택하여 해당 키에 대한 세부 정보를 표시할 수 있습니다.

Transfer Family > Servers > s-3fe215d89f074ed2a > Hostkey: hostkey-a6fed60bcb704ea9b

hostkey-a6fed60bcb704ea9b Delete

Host key configuration Edit

Name ecdsa-521	Key type ecdsa-sha2-nistp521
Fingerprint SHA256: [redacted]	Date imported Thu, 13 Jun 2024 17:08:59 GMT
Description ECDSA host key	Amazon Resource Name (ARN) arn:aws:transfer:us-east-1:[redacted]:host-key/s-[redacted]/hostkey-a-[redacted]

서버 세부 정보 화면의 작업 메뉴에서 호스트 키를 삭제하거나 설명을 편집할 수 있습니다. 호스트 키를 선택한 다음 메뉴에서 적절한 작업을 선택합니다.

Server host keys (2)

Actions Add host key < 1 >

Edit Delete

	Name	Host key ID	Fingerprint	Description	Key type	Date imported
☐	Default host key	hostkey-[redacted]	SHA256:[redacted]	Default	ssh-rsa	2023-06-30
☒	ecdsa-521	hostkey-[redacted]	SHA256:[redacted]	ECDSA host key	ecdsa-sha2-nistp521	2024-06-13

콘솔 내 사용량 모니터링

서버 세부 정보 페이지에서 서버 메트릭에 대한 정보를 얻을 수 있습니다. 이를 통해 파일 전송 워크로드를 한 곳에서 모니터링할 수 있습니다. 중앙 집중식 대시보드를 사용하여 파트너와 교환한 파일 수를

추적하고 파트너의 사용량을 면밀히 추적할 수 있습니다. 자세한 내용은 [SFTP, FTPS 및 FTP 서버 세부 정보 보기](#)를 참조하세요. 다음 표에서는 Transfer Family에 사용할 수 있는 측정치를 설명합니다.

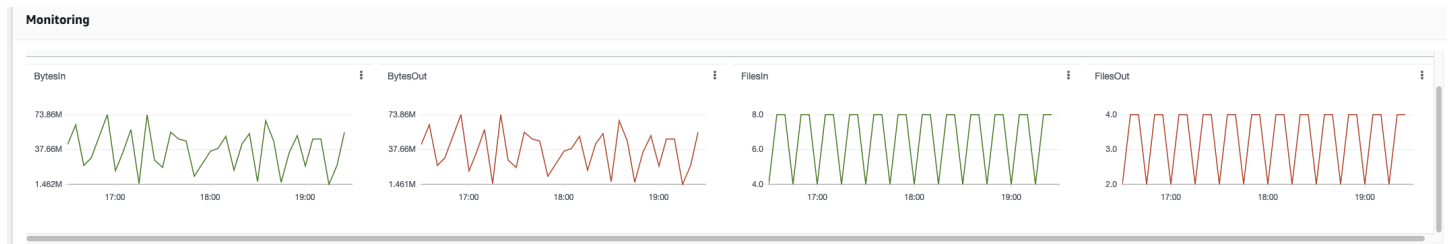
네임스페이스	지표	설명
AWS/Transfer	BytesIn	서버로 전송된 총 바이트 수. 보고 기준: - SFTP/FTP/FTPS의 경우: Transfer Family 서버에 대한 연결이 설정되는 동안 5분마다 내보내집니다. 해당 기간 동안 파일 또는 바이트가 전송되지 않으면 "0"이 발생합니다. - AS2의 경우: 고객이 AS2 서버에서 메시지를 수신하고 인바운드 메시지 처리가 완료되는 즉시 내보낸 경우 단위: 개 기간: 5분
	BytesOut	서버에서 전송된 총 바이트 수입니다. 보고 기준: - SFTP/FTP/FTPS의 경우: Transfer Family 서버에 대한 연결이 설정되는 동안 5분마다 내보내집니다. 해당 기간 동안 파일 또는 바이트가 전송되지 않으면 "0"이 발생합니다. - AS2의 경우: 고객이 AS2 커넥터StartFile Transfer 에서를 호출하고 아웃바운드 메시지 처리가 완료되는 즉시 내보낸 경우. 단위: 개 기간: 5분
	FilesIn	서버로 전송된 총 파일 수.

네임스페이스	지표	설명
		AS2 프로토콜을 사용하는 서버의 경우 이 지표는 수신된 메시지 수를 나타냅니다. 보고 기준: - SFTP/FTP/FTPS의 경우: Transfer Family 서버에 대한 연결이 설정되는 동안 5분마다 내보내집니다. 해당 기간 동안 파일 또는 바이트가 전송되지 않으면 "0"이 발생합니다. - AS2의 경우: 고객이 AS2 서버에서 메시지를 수신하고 인바운드 메시지 처리가 완료되는 즉시 내보낸 경우. 단위: 개 기간: 5분
	FilesOut	서버에서 전송된 총 파일 수. 보고 기준: - SFTP/FTP/FTPS의 경우: Transfer Family 서버에 대한 연결이 설정되는 동안 5분마다 내보내집니다. 해당 기간 동안 파일 또는 바이트가 전송되지 않으면 "0"이 발생합니다. - AS2의 경우: 고객이 AS2 커넥터StartFile Transfer 에서를 호출하고 아웃바운드 메시지 처리가 완료되는 즉시 내보낸 경우. 단위: 개 기간: 5분

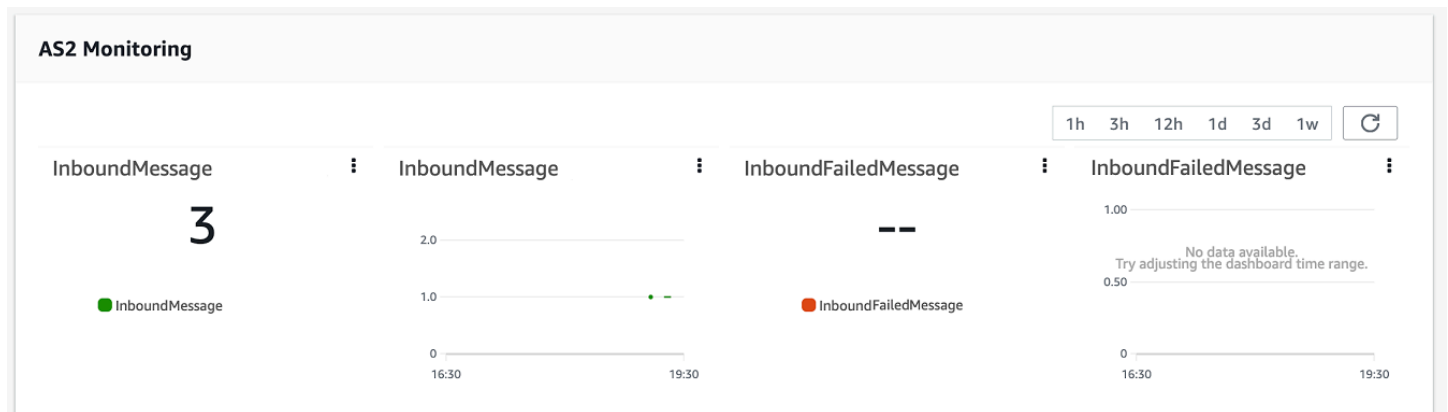
네임스페이스	지표	설명
	InboundMessage	거래 파트너로부터 성공적으로 수신한 AS2 메시지의 총 개수. 보고 기준: 고객이 AS2 서버에서 메시지를 수신하고 인바운드 메시지 처리가 성공적으로 완료되는 즉시 내보낸 경우 단위: 개 기간: 5분
	InboundFailedMessage	거래 파트너로부터 성공적으로 수신되지 못한 AS2 메시지의 총 개수. 즉, 거래 파트너가 메시지를 보냈지만 Transfer Family 서버가 메시지를 성공적으로 처리하지 못했습니다. 보고 기준: 고객이 AS2 서버에서 메시지를 수신하고 인바운드 메시지 처리가 성공적으로 완료되는 즉시 전송된 경우 단위: 개 기간: 5분
	OnUploadExecutionsStarted	서버에서 시작된 워크플로를 실행한 총 횟수. 보고 기준: 실행이 시작될 때마다 트리거됩니다. 단위: 개 기간: 1분

네임스페이스	지표	설명
	OnUploadExecutionsSuccess	서버에서 성공적인 워크플로를 실행한 총 횟수. 보고 기준: 실행이 성공적으로 완료될 때마다 트리거됩니다. 단위: 개 기간: 1분
	OnUploadExecutionsFailed	서버에서 실패한 워크플로를 실행한 총 횟수. 보고 기준: 실행이 성공적으로 완료되지 않을 때마다 트리거됩니다. 단위: 개 기간: 1분
	OutboundMessage	거래 파트너에게 성공적으로 전송된 AS2 메시지의 총 수입니다. 보고 기준: 고객이 AS2 커넥터StartFileTransfer 에서 전화를 걸어 아웃바운드 메시지 처리가 성공적으로 완료되는 즉시 내보낸 경우 단위: 개 기간: 5분
	OutboundFailedMessage	거래 파트너에게 성공적으로 전송되지 못한 AS2 메시지의 총 개수. 보고 기준: 고객이 AS2 커넥터StartFileTransfer 에서 전화를 걸어 아웃바운드 메시지 처리가 성공적으로 완료되는 즉시 내보낸 경우 단위: 개 기간: 5분

모니터링 섹션에는 4개의 개별 그래프가 있습니다. 이 그래프는 입력 바이트, 출력 바이트, 파일 입력 및 출력 파일을 보여 줍니다.



AS2 프로토콜이 활성화된 서버의 경우 모니터링 정보 아래에 AS2 모니터링 섹션이 있습니다. 이 섹션에는 성공 및 실패 인바운드 메시지 수에 대한 세부 정보가 포함되어 있습니다.



선택한 그래프를 자체 창에서 열려면 확장 아이콘



을 선택합니다. 그래프의 세로 줄임표 아이콘



을 클릭하여 다음 항목이 포함된 드롭다운 메뉴를 열 수도 있습니다.

- 확대 - 선택한 그래프를 자체 창에서 엽니다.
- 새로 고침 - 그래프를 최신 데이터로 다시 로드합니다.
- 지표에서 보기 - Amazon CloudWatch에서 해당 지표 세부 정보를 엽니다.
- 로그 보기 - CloudWatch에서 해당 로그 그룹을 엽니다.

액세스 통제 관리

AWS Identity and Access Management (IAM) 정책을 사용하여 AWS Transfer Family 리소스에 대한 사용자의 액세스를 제어할 수 있습니다. IAM 정책은 명령문으로, 대부분 리소스에 대한 특정 수준의 액세스를 허용하는 JSON 형식을 취합니다. IAM 정책을 사용하여 SFTP 사용자가 수행하거나 수행하지 않게 할 파일 작업을 정의합니다. 또한 IAM 정책을 이용해 사용자에게 액세스를 허용할 S3 버킷 또는 일반 버킷을 정의할 수도 있습니다. 사용자에게 이러한 정책을 지정하려면 IAM 정책 및 신뢰 관계가 연결된 AWS Transfer Family 에 대한 IAM 역할을 생성합니다.

각 사용자에게 IAM 역할이 할당됩니다. 에서 AWS Transfer Family 사용하는 IAM 역할 유형을 서비스 역할이라고 합니다. 사용자가 서버에 로그인하면는 사용자에게 매핑된 IAM 역할을 AWS Transfer Family 가정합니다. Amazon S3 버킷에 대한 사용자 액세스를 제공하는 IAM 역할을 생성하는 방법에 대한 자세한 내용은 IAM 사용 설명서의 [AWS 서비스에 대한 권한을 위임할 역할 생성을 참조하세요](#).

IAM 정책 내에서 특정 권한을 사용하여 Amazon S3 객체에 대한 쓰기 전용 액세스 권한을 부여할 수 있습니다. 자세한 내용은 [파일을 쓰고 나열할 수 있는 권한만 부여](#)를 참조하세요.

AWS 스토리지 블로그에는 최소 권한 액세스를 설정하는 방법을 자세히 설명하는 게시물이 포함되어 있습니다. 자세한 내용은 [AWS Transfer Family 워크플로에서 최소 권한 액세스 구현을 참조하세요](#).

Note

Amazon S3 버킷이 AWS Key Management Service (AWS KMS)를 사용하여 암호화된 경우 정책에 추가 권한을 지정해야 합니다. 자세한 내용은 [데이터 보호 및 암호화](#)를 참조하세요. 또한 IAM 사용자 가이드에서 [세션 정책에](#) 대한 자세한 정보를 확인할 수 있습니다.

주제

- [Amazon S3 버킷에 대한 읽기 및 쓰기 액세스 허용](#)
- [Amazon S3 버킷을 위한 세션 정책 생성](#)
- [동적 권한 관리 접근 방식](#)

Amazon S3 버킷에 대한 읽기 및 쓰기 액세스 허용

이 섹션에서는 특정 Amazon S3 버킷에 대한 읽기 및 쓰기 액세스를 허용하는 IAM 정책 생성 방법을 설명합니다. 이 IAM 정책이 있는 IAM 역할을 사용자에게 할당하면, 해당 사용자는 지정한 Amazon S3 버킷에 대한 읽기 및 쓰기 액세스 권한을 얻게 됩니다.

다음 정책은 Amazon S3 버킷에 대한 프로그래밍 방식의 읽기 및 쓰기 액세스 권한을 제공합니다. GetObjectACL 및 PutObjectACL 명령문은 크로스 계정 액세스를 활성화할 필요가 있는 경우에만 요구됩니다. 예컨대, Transfer Family 서버가 다른 계정의 버킷에 액세스할 필요가 있는 경우입니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ReadWriteS3",
      "Action": [
        "s3:ListBucket"
      ],
      "Effect": "Allow",
      "Resource": ["arn:aws:s3:::amzn-s3-demo-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:::amzn-s3-demo-bucket/*"]
    }
  ]
}
```

ListBucket 작업은 버킷 자체에 대한 권한을 요구합니다. PUT, GET, DELETE 작업은 객체 권한을 요구합니다. 이들은 각각 다른 리소스이며, 따라서 서로 다른 Amazon 리소스 이름(ARN)을 이용해 지정됩니다.

사용자의 액세스를 지정된 Amazon S3 버킷의 home 접두사로만 제약하는 방법은 [Amazon S3 버킷을 위한 세션 정책 생성](#) 섹션을 참조하세요.

Amazon S3 버킷을 위한 세션 정책 생성

세션 정책은 사용자를 Amazon S3 버킷의 특정 부분으로 제한하는 AWS Identity and Access Management (IAM) 정책입니다. 이 정책은 액세스를 실시간으로 평가해 이를 수행합니다.

Note

세션 정책은 Amazon S3에서만 사용됩니다. Amazon EFS의 경우 POSIX 파일 권한을 사용하여 액세스를 제한합니다.

세션 정책은 사용자 그룹 전원이 Amazon S3 버킷의 특정 부분에만 액세스하도록 제한해야 할 때 사용됩니다. 예를 들어 사용자 그룹이 home 디렉터리에만 액세스해야 할 수도 있습니다. 해당 사용자 그룹은 같은 IAM 역할을 공유합니다.

Note

세션 정책의 최대 길이는 2,048자입니다. 자세한 내용은 API 참조의 CreateUser 작업에 대한 [정책 요청 파라미터](#)를 참조하세요.

세션 정책을 생성하려면, IAM 정책에서 다음 정책 변수를 사용하세요.

- `${transfer:HomeBucket}`
- `${transfer:HomeDirectory}`
- `${transfer:HomeFolder}`
- `${transfer:UserName}`

Important

관리형 정책에서는 위의 변수를 사용할 수 없습니다. 또한 위의 변수를 IAM 역할 정의에서 정책 변수로 사용할 수도 없습니다. IAM 정책에 있는 이러한 변수는 사용자를 설정할 때 생성하고 바로 공급합니다. 그리고 이 세션 정책에서는 `${aws:Username}` 변수는 사용할 수 없습니다. 이 변수는 AWS Transfer Family이(가) 요구하는 사용자 이름이 아닌 IAM 사용자 이름을 의미합니다.

세션 정책 예제

다음 코드는 예 세션 정책을 보여줍니다.

```
{
  "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

위의 정책 예에서는 사용자의 홈 디렉터리가 디렉터리임을 나타내기 위해 후행 슬래시를 포함하도록 설정된 것으로 가정합니다. 그 반면에 후행 슬래시가 없이 사용자의 HomeDirectory를 설정하는 경우에는 이를 정책의 일부로 포함해야 합니다.

이전 예 정책에서, `transfer:HomeFolder`, `transfer:HomeBucket`, `transfer:HomeDirectory` 정책 파라미터의 사용을 기억해 두세요. 이러한 파라미터는 [HomeDirectory](#) 및 [API Gateway 메서드 구현](#)에 설명된 대로 사용자에게 맞게 구성된 HomeDirectory을 위해 설정됩니다 이러한 파라미터의 정의는 다음과 같습니다.

- `transfer:HomeBucket` 파라미터는 HomeDirectory의 첫 번째 구성 요소로 대체됩니다.
- `transfer:HomeFolder` 파라미터가 HomeDirectory 파라미터의 나머지 부분으로 대체됩니다.
- `transfer:HomeDirectory` 파라미터는 Resource 명령문에서 S3 Amazon 리소스 이름 (ARN)의 일부로 사용할 수 있도록 선행 슬래시(/)가 제거되었습니다.

Note

귀하가 논리적 디렉터를 사용하는 경우—즉 사용자의 `homeDirectoryType`가 LOGICAL인 경우—이러한 정책 파라미터(`HomeBucket`, `HomeDirectory`, 및 `HomeFolder`)는 지원되지 않습니다.

예를 들어, Transfer Family 사용자에게 대해 구성된 HomeDirectory 파라미터가 `/home/bob/amazon/stuff/(이)`라고 가정해 보겠습니다.

- `transfer:HomeBucket`를 `/home(으)`로 설정합니다.
- `transfer:HomeFolder`를 `/bob/amazon/stuff/(으)`로 설정합니다.
- `transfer:HomeDirectory`는 `home/bob/amazon/stuff/`가 됩니다.

첫 번째 "Sid"은(는) 사용자가 `/home/bob/amazon/stuff/`부터 시작하여 모든 디렉터리를 나열할 수 있게 해 줍니다.

두 번째 "Sid"은(는) 동일한 경로인 `/home/bob/amazon/stuff/`에 대한 사용자 put 및 get 액세스를 제한합니다.

앞의 정책을 활성화하면, 로그인 시 사용자는 자신의 홈 디렉터리에 있는 객체만 액세스할 수 있습니다. 연결 시는 이러한 변수를 사용자에게 적합한 값으로 바 AWS Transfer Family 끝니다. 이렇게 하면 같은 정책 설명서를 다수의 사용자에게 쉽게 적용할 수 있습니다. 이 방법은 Amazon S3 버킷에 대한 사용자의 액세스 관리에 필요한 IAM 역할과 정책 관리 오버헤드를 줄입니다.

세션 정책을 사용하면 업무 요건에 따라 각 사용자의 액세스를 조절할 수도 있습니다. 자세한 내용은 IAM 사용자 가이드의 [Permissions for AssumeRole, AssumeRoleWithSAML 및 AssumeRoleWithWebIdentity](#)를 참조하세요.

Note

AWS Transfer Family 는 정책의 Amazon 리소스 이름(ARN) 대신 정책 JSON을 저장합니다. 따라서 IAM 콘솔에서 정책을 변경할 때 AWS Transfer Family 콘솔로 돌아가서 사용자를 최신 정책 내용으로 업데이트해야 합니다. 사용자 구성 섹션의 정책 정보 탭에서 사용자를 업데이트 할 수 있습니다.

를 사용하는 경우 다음 명령을 사용하여 정책을 업데이트할 AWS CLI 수 있습니다.

```
aws transfer update-user --server-id server --user-name user --policy \
    "$(aws iam get-policy-version --policy-arn policy --version-id version --
    output json)"
```

세션 정책에 대한 중첩 대체

중첩 대체는 Transfer Family 세션 정책에서 수행되지 않습니다. 세션 정책은와 같은 중첩 변수를 사용할 수 있습니다 `${transfer:HomeDirectory}`. 정책이 처리되면 외부 변수(예: `${transfer:HomeDirectory}`)가 다른 변수(예: `{amzn-s3-demo-bucket:/${transfer:UserName}}`)가 포함된 값으로 대체될 수 있습니다. 그러나 중첩 변수는 실제 사용자 이름(예: johndoe)으로 더 이상 대체되지 않습니다.

즉, Transfer Family에 대한 세션 정책을 생성할 때이 동작을 고려하고 정책 구조와 변수 사용량이 그에 따라 설계되어야 합니다. 중첩된 변수는 예상대로 확인되지 않을 수 있으며 정책은 의도한 권한을 부여 하지 않을 수 있습니다. 세션 정책이 예상대로 작동하는지 철저히 테스트하고 검증하는 것이 중요합니다. 이 동작은 Transfer Family 환경에 대한 액세스 제어 및 권한을 구현할 때 주요 고려 사항입니다.

이 문제의 한 가지 해결 방법은 세션 정책에서 실제 Amazon S3 버킷 이름을 사용하는 것입니다. 예를 들어 세션 정책 `${transfer:HomeDirectory}`에서를 지정하는 대신 `amzn-s3-demo-bucket`이 실제 버킷인 `amzn-s3-demo-bucket/transfer:UserName`를 사용합니다.

동적 권한 관리 접근 방식

Transfer Family 권한 아키텍처 이해

AWS Transfer Family 는 런타임 시 IAM 역할의 유효 권한을 제한할 수 있는 세션 정책을 통해 동적 권한 관리를 지원합니다. 이 접근 방식은 서비스 관리형 사용자와 사용자 지정 자격 증명 공급자 사용자 모두에서 작동하지만 Amazon S3(Amazon EFS 아님)와 파일을 주고받을 때만 지원됩니다.

모든 AWS Transfer Family 사용자는 다음으로 구성된 권한 모델을 사용합니다.

1. 기본 IAM 역할 - 사용자의 기본 권한을 정의합니다.
2. 선택적 세션 정책 - 런타임 시 기본 권한을 제한(범위 축소)합니다.

유효 권한은 기본 역할 권한과 세션 정책 권한의 교차입니다. 세션 정책은 권한을 제한할 수만 있으며 기본 역할이 허용하는 것 이상의 추가 권한을 부여할 수는 없습니다.

이 아키텍처는 두 사용자 유형에 모두 적용됩니다.

- 서비스 관리형 사용자 - 세션 정책은 사용자 설정에서 직접 구성할 수 있습니다.
- 사용자 지정 자격 증명 공급자 사용자 - 세션 정책을 인증 응답의 일부로 반환하거나에 저장할 수 있습니다. AWS Secrets Manager

권한 관리에 대한 두 가지 접근 방식

고유한 액세스 패턴이 필요한 Transfer Family 사용자를 위한 권한을 설계할 때 두 가지 주요 접근 방식 중에서 선택할 수 있습니다.

사용자당 역할 1개

각 Transfer Family 사용자에게 대해 해당 사용자의 필요에 맞는 특정 권한을 가진 별도의 IAM 역할을 생성합니다. 다음과 같은 경우이 접근 방식을 사용합니다.

- 각 사용자에게는 매우 다른 권한이 필요합니다.
- 권한 관리는 조직의 다양한 사용자가 처리합니다.
- 개별 사용자 액세스를 세밀하게 제어해야 합니다.

세션 정책과 역할 공유

광범위한 권한(예: 여러 사용자 홈 디렉터리가 포함된 전체 Amazon S3 버킷에 대한 액세스)이 있는 단일 IAM 역할을 사용하고 세션 정책을 적용하여 각 사용자를 특정 영역으로 제한합니다. 이 접근

방식은 각 사용자에게 대해 별도의 역할을 관리하는 것에 비해 관리 오버헤드를 크게 줄입니다. 다음과 같은 경우가 접근 방식을 사용합니다.

- 사용자에게는 비슷한 유형의 액세스가 필요하지만 다른 리소스에 대한 액세스가 필요합니다(예: 모든 사용자에게 읽기/쓰기 액세스가 필요하지만 각각 자신의 폴더에만 액세스).
- 역할을 관리를 간소화하고 수십 또는 수백 개의 개별 역할을 생성하지 않으려는 경우
- 사용자는 공유 버킷 내의 지정된 홈 디렉터리에만 액세스해야 합니다.
- 권한 관리는 조직 내에서 중앙 집중화됩니다.

예를 들어 사용자 "alice", "bob" 및 "charlie"에 대해 별도의 역할을 생성하는 대신 전체 `s3://company-transfers/` 버킷에 액세스할 수 있는 하나의 역할을 생성한 다음 세션 정책을 사용하여 alice를 `s3://company-transfers/alice/`로, bob을 `s3://company-transfers/bob/`로 제한할 수 있습니다.

세션 정책 구현

세션 정책은 사용자에게 할당된 기본 IAM 역할의 유효 권한을 제한하여 작동합니다. 최종 권한은 역할 권한과 세션 정책의 권한의 교집합입니다.

다음 두 가지 방법으로 동적 세션 정책을 구현할 수 있습니다.

변수 대체

세션 정책 `${transfer:HomeBucket}`에서 `${transfer:Username}`, `${transfer:HomeDirectory}` 및와 같은 Transfer Family 정책 변수를 사용합니다. 이러한 변수는 런타임 시 실제 값으로 자동 대체됩니다. 이러한 변수에 대한 자세한 내용은 [섹션을 참조하세요 Amazon S3 버킷을 위한 세션 정책 생성](#).

동적 생성

사용자 지정 자격 증명 공급자의 경우 Lambda 함수 또는 API Gateway 메온-the-fly 일부로 세션 정책을 즉시 생성합니다. 이 접근 방식을 사용하면 인증 시 사용자 속성, 그룹 멤버십 또는 외부 데이터 소스를 기반으로 고도로 사용자 지정된 정책을 생성할 수 있습니다.

세션 정책 JSON과 Policy 함께 라는 키를 값으로 AWS Secrets Manager 포함하여에 사전 생성된 세션 정책을 저장할 수도 있습니다. 이를 통해 사용자별 액세스 제어를 유지하면서 여러 사용자에게 걸쳐 동일한 광범위한 IAM 역할을 사용할 수 있습니다.

Note

세션 정책은 Amazon S3와의 파일 전송에만 지원됩니다. Amazon EFS 파일 시스템에는 적용되지 않습니다. Amazon EFS의 경우 권한은 UID/GID 및 파일 시스템 자체 내에 적용된 권한 비트에 의해 관리됩니다.

사용자 유형별 구현

서비스 관리형 사용자

서비스 관리형 사용자의 경우 AWS Transfer Family 콘솔, API 또는 CLI를 통해 사용자 구성에서 직접 세션 정책을 지정할 수 있습니다. 자세한 내용은 [서비스 관리형과 작업](#) 단원을 참조하십시오.

사용자 지정 자격 증명 공급자 사용자

사용자 지정 자격 증명 공급자 사용자의 경우 두 가지 방법으로 세션 정책을 제공할 수 있습니다.

- 세션 정책과 Policy 함께 라는 키를 값으로 포함 AWS Secrets Manager 시크를 통해
- 인증 결과의 일부로 Lambda 함수 응답 또는 API Gateway 응답에서 직접

자세한 내용은 [사용자 지정 자격 증명 공급자 솔루션](#) 단원을 참조하십시오.

예: 세션 정책을 사용하여 역할 관리 간소화

이 예제에서는 동적 권한 관리가 보안을 유지하면서 관리 오버헤드를 크게 줄이는 방법을 보여줍니다.

시나리오

조직에 파일을 전송하기 위해 SFTP 액세스 권한이 필요한 사용자가 50명 있습니다. 각 사용자는 라는 공유 Amazon S3 버킷 내의 자체 폴더에만 액세스해야 합니다 company-transfers. 세션 정책이 없으면 50개의 개별 IAM 역할을 생성해야 합니다.

기존 접근 방식(세션 정책 없음)

- 50개의 IAM 역할 생성: TransferRole-Alice, TransferRole-Bob, TransferRole-Charlie, 등
- 각 역할에는 해당 사용자의 폴더에 대한 특정 권한만 있습니다.
- 권한을 관리하려면 개별 역할을 업데이트해야 합니다.
- 새 사용자를 추가하려면 새 역할을 생성해야 합니다.

동적 접근 방식(세션 정책 사용)

- IAM 역할 1개 생성: 전체 버킷에 대한 광범위한 권한 TransferRole-Shared 있음
- 세션 정책을 사용하여 런타임 시 각 사용자를 특정 폴더로 제한
- 권한을 관리하려면 하나의 역할 또는 세션 정책 템플릿을 업데이트해야 합니다.
- 새 사용자를 추가하면 새 역할이 필요하지 않고 사용자 구성만 필요합니다.

구현

동적 접근 방식을 구현하는 방법은 다음과 같습니다(company-transfers 버킷을 실제 Amazon S3 버킷으로 대체하는 예제로 사용).

동적 권한 관리를 구현하려면

1. 광범위한 Amazon S3 권한을 사용하여 공유 IAM 역할 하나를 생성합니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetObject",
        "s3:PutObject",
        "s3:DeleteObject"
      ],
      "Resource": "arn:aws:s3:::company-transfers/*"
    },
    {
      "Effect": "Allow",
      "Action": "s3:ListBucket",
      "Resource": "arn:aws:s3:::company-transfers"
    }
  ]
}
```

2. 사용자의 폴더에 대한 액세스를 제한하는 세션 정책 템플릿을 생성합니다.

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



```

"Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "s3:GetObject",
      "s3:PutObject",
      "s3:DeleteObject"
    ],
    "Resource": "arn:aws:s3:::company-transfers/${transfer:Username}/*"
  },
  {
    "Effect": "Allow",
    "Action": "s3:ListBucket",
    "Resource": "arn:aws:s3:::company-transfers",
    "Condition": {
      "StringLike": {
        "s3:prefix": "${transfer:Username}*"
      }
    }
  }
]
}

```

3. 다음을 사용하여 각 사용자를 구성합니다.

- 공유 IAM 역할
- 세션 정책은 다음과 같이 적용됩니다.
 - 서비스 관리형 사용자: 사용자를 생성하거나 수정할 때 API 또는 CLI를 사용하여 정책 파라미터를 통해 JSON을 적용합니다(콘솔은 사전 정의된 정책 옵션만 제공).
 - 사용자 지정 자격 증명 공급자 사용자: 인증 중에 Lambda 함수 응답의 일부로 반환하거나 사용자의 자격 증명과 함께 "정책"이라는 키 AWS Secrets Manager 로에 저장합니다.
- 홈 디렉터리: /company-transfers/\${transfer:Username}/

AWS CloudTrail 에 대한 로깅 AWS Transfer Family

AWS Transfer Family 는 AWS CloudTrail 및 Amazon CloudWatch와 통합됩니다. CloudTrail과 CloudWatch는 서로 다르지만 보완적인 용도로 사용됩니다.

- 이 주제에서는 내에서 수행된 작업의 레코드를 생성하는 AWS 서비스인 CloudTrail과의 통합을 다룹니다. AWS 계정, 콘솔 로그인, AWS Command Line Interface 명령 및 SDK/API 작업과 같은 활동에 대한 API 작업을 지속적으로 모니터링하고 기록합니다. 이를 통해 누가 언제 어디서 어떤 조치를 취했는지에 대한 로그를 유지할 수 있습니다. CloudTrail은 환경의 AWS 모든 활동에 대한 기록을 제공하여 감사, 액세스 관리 및 규정 준수를 지원합니다. 자세한 내용은 [AWS CloudTrail 사용 설명서를](#) 참조하세요.
- [AWS Transfer Family 서버에 대한 Amazon CloudWatch 로깅](#)에서는 AWS 리소스 및 애플리케이션에 대한 모니터링 서비스인 CloudWatch와의 통합을 다룹니다. 지표와 로그를 수집하여 리소스 사용률, 애플리케이션 성능 및 전체 시스템 상태에 대한 가시성을 제공합니다. CloudWatch는 문제 해결, 경보 설정, 자동 크기 조정과 같은 운영 작업을 지원합니다. 자세한 내용은 [Amazon CloudWatch 사용 설명서를](#) 참조하세요.

트레일이란 지정한 S3 버킷에 이벤트를 로그 파일로 입력할 수 있게 하는 구성입니다. CloudTrail 로그 파일에는 하나 이상의 로그 항목이 포함될 수 있습니다. 이벤트는 모든 소스로부터의 단일 요청을 나타내며 요청 작업, 작업 날짜와 시간, 요청 파라미터 등에 대한 정보가 들어 있습니다. CloudTrail 로그 파일은 퍼블릭 API 작업의 순서가 지정된 스택 추적이 아니므로 특정 순서로 표시되지 않습니다.

에 대한 이벤트를 포함하여 AWS 계정에 이벤트를 지속적으로 기록하려면 추적을 AWS Transfer Family 생성합니다. CloudTrail은 추적을 사용하여 Amazon S3 버킷으로 로그 파일을 전송할 수 있습니다. 콘솔에서 추적을 생성하면 기본적으로 모든 AWS Region에 추적이 적용됩니다. 추적은 AWS 파티션의 모든 리전에서 이벤트를 로깅하고 지정한 Amazon S3 버킷으로 로그 파일을 전송합니다. 추가적으로, CloudTrail 로그에서 수집된 이벤트 데이터를 추가 분석 및 처리하도록 다른 AWS 서비스를 구성할 수 있습니다. 자세한 내용은 다음 자료를 참조하세요.

- [추적 생성 개요](#)
- [CloudTrail 지원 서비스 및 통합](#)
- [CloudTrail에 대한 Amazon SNS 알림 구성](#)
- [여러 리전에서 CloudTrail 로그 파일 받기](#) 및 [여러 계정에서 CloudTrail 로그 파일 받기](#)

모든 AWS Transfer Family 작업은 CloudTrail에서 로깅되며 [Actions](#)에 문서화됩니다. [API reference](#). 예를 들어 CreateServer, ListUsers, StopServer 작업을 직접 호출하면 CloudTrail 로그 파일에 항목이 생성됩니다.

모든 이벤트 또는 로그 항목에는 요청을 생성했던 사용자에게 관한 정보가 포함됩니다. ID 정보를 이용하면 다음을 쉽게 판단할 수 있습니다.

- 요청이 루트 또는 AWS Identity and Access Management 사용자 자격 증명으로 이루어졌는지 여부입니다.
- 역할 또는 페더레이션 사용자의 임시 자격 증명을 사용하여 요청이 생성되었는지 여부.
- 요청이 다른 AWS 서비스에 의해 이루어졌는지 여부입니다.

자세한 내용은 [CloudTrail userIdentity 요소](#)를 참조하세요.

추적을 생성하면 이벤트를 포함하여 CloudTrail 이벤트를 지속적으로 Amazon S3 버킷에 배포할 수 있습니다 AWS Transfer Family. 추적을 구성하지 않은 경우에도 CloudTrail 콘솔의 이벤트 기록에서 최신 이벤트를 볼 수 있습니다.

CloudTrail에서 수집한 정보를 사용하여 AWS Transfer Family에 수행된 요청, 요청이 수행된 IP 주소, 요청을 수행한 사람, 요청이 수행된 시간 및 추가 세부 정보를 확인할 수 있습니다.

CloudTrail에 대한 자세한 내용은 [AWS CloudTrail 사용 설명서](#)를 참조하세요.

주제

- [AWS CloudTrail 로깅 활성화](#)
- [서버 생성을 위한 로그 항목의 예](#)
- [데이터 액세스 로그 예제](#)

AWS CloudTrail 로깅 활성화

를 사용하여 AWS Transfer Family API 작업을 모니터링할 수 있습니다 AWS CloudTrail. API 작업을 모니터링하면 유용한 보안 및 운영 정보를 얻을 수 있습니다. [Amazon S3 객체 수준 로깅을 활성화한](#) 경우, 요청자 필드에 RoleSessionName이(가) [AWS:Role Unique Identifier]/username.sessionid@server-id(으)로 포함됩니다. AWS Identity and Access Management (IAM) 역할 고유 식별자에 대한 자세한 내용은 AWS Identity and Access Management 사용 설명서의 [고유 식별자](#)를 참조하세요.

⚠ Important

RoleSessionName의 최대 길이는 64자입니다. RoleSessionName 길이가 더 길면 server-id이(가) 잘립니다.

Amazon S3 데이터 이벤트 활성화

Amazon S3 버킷 AWS Transfer Family 에서를 통해 수행되는 파일 작업을 추적하려면 해당 버킷에 대한 데이터 이벤트를 활성화해야 합니다. 데이터 이벤트는 객체 수준 API 활동을 제공하며 AWS Transfer Family 사용자가 수행하는 파일 업로드, 다운로드 및 기타 작업을 추적하는 데 특히 유용합니다.

AWS Transfer Family 서버에 대해 Amazon S3 데이터 이벤트를 활성화하려면:

1. <https://console.aws.amazon.com/cloudtrail/>에서 CloudTrail 콘솔을 엽니다.
2. 탐색 창에서 추적을 선택한 다음 기존 추적을 선택하거나 새 추적을 생성합니다.
3. 데이터 이벤트에서 편집을 선택합니다.
4. 데이터 이벤트 유형에서 S3를 선택합니다.
5. 데이터 이벤트를 로깅할 Amazon S3 버킷을 선택합니다. 모든 버킷에 대한 데이터 이벤트를 로깅하거나 개별 버킷을 지정할 수 있습니다.
6. 읽기 이벤트, 쓰기 이벤트 또는 둘 다를 로깅할지 여부를 선택합니다.
7. 변경 사항 저장을 선택합니다.

데이터 이벤트를 활성화한 후 CloudTrail 추적에 대해 구성된 Amazon S3 버킷에서 이러한 로그에 액세스할 수 있습니다. 로그에는 작업을 수행한 사용자, 작업 타임스탬프, 영향을 받는 특정 객체, 수행된 작업에 userId 대해를 추적하는 데 도움이 되는 onBehalfOf 필드와 같은 세부 정보가 포함됩니다 AWS Transfer Family.

서버 생성을 위한 로그 항목의 예

다음 예제에서는 CreateServer 작업을 보여주는 CloudTrail 로그 항목(JSON 형식)을 보여줍니다.

```
{
  "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"
  }
}

```

데이터 액세스 로그 예제

CloudTrail 추적에 대해 Amazon S3 데이터 이벤트를 활성화하면를 통해 수행된 파일 작업을 추적할 수 있습니다 AWS Transfer Family. 이러한 로그는 누가 어떤 데이터에 언제 어떻게 액세스했는지 모니터링하는 데 도움이 됩니다.

성공적인 데이터 액세스를 위한 로그 항목의 예

다음 예제는 성공적인 파일 다운로드 작업에 대한 CloudTrail 로그 항목을 보여줍니다 AWS Transfer Family.

```

{
  "eventVersion": "1.09",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "AROEXAMPLEID:TransferSessionUser",
    "arn": "arn:aws:sts::123456789012:assumed-role/TransferS3AccessRole/TransferSessionUser",
    "accountId": "123456789012",
    "accessKeyId": "ASIAEXAMPLEKEY",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "AROEXAMPLEID",
        "arn": "arn:aws:iam::123456789012:role/TransferS3AccessRole",
        "accountId": "123456789012",
        "userName": "TransferS3AccessRole"
      }
    }
  }
}

```

```

    },
    "attributes": {
      "creationDate": "2025-07-15T16:12:05Z",
      "mfaAuthenticated": "true"
    }
  },
  "invokedBy": "transfer.amazonaws.com"
},
"eventTime": "2025-07-15T16:15:22Z",
"eventSource": "s3.amazonaws.com",
"eventName": "GetObject",
"awsRegion": "us-east-1",
"sourceIPAddress": "transfer.amazonaws.com",
"userAgent": "transfer.amazonaws.com",
"requestParameters": {
  "bucketName": "my-transfer-bucket",
  "key": "users/john.doe/reports/quarterly-report-2025-Q2.pdf",
  "Host": "my-transfer-bucket.s3.amazonaws.com",
  "x-amz-request-payer": "requester"
},
"responseElements": null,
"additionalEventData": {
  "SignatureVersion": "SigV4",
  "CipherSuite": "ECDHE-RSA-AES128-GCM-SHA256",
  "bytesTransferredIn": 0,
  "bytesTransferredOut": 2458732,
  "x-amz-id-2":
"EXAMPLE123456789+abcdefghijklmnopqrstuvwxyABCDEFGHIJKLMNOPQRSTUVWXYZ="
},
"requestID": "EXAMPLE123456789",
"eventID": "example12-3456-7890-abcd-ef1234567890",
"readOnly": true,
"resources": [
  {
    "type": "AWS::S3::Object",
    "ARN": "arn:aws:s3:::my-transfer-bucket/users/john.doe/reports/quarterly-
report-2025-Q2.pdf"
  },
  {
    "accountId": "123456789012",
    "type": "AWS::S3::Bucket",
    "ARN": "arn:aws:s3:::my-transfer-bucket"
  }
],

```

```

    "eventType": "AwsApiCall",
    "managementEvent": false,
    "recipientAccountId": "123456789012",
    "eventCategory": "Data",
    "requestParameters": {
        "x-amz-onBehalfOf": "john.doe.sessionid@s-abcd1234efgh5678"
    }
}

```

이 예제에서는 다음 중요 필드를 기록해 둡니다.

- `eventName`: 수행된 S3 API 작업을 나타냅니다(파일 다운로드용 `GetObject`).
- `requestParameters.bucketName` 및 `requestParameters.key`: 어떤 S3 객체에 액세스했는지 표시합니다.
- `additionalEventData.bytesTransferredOut`: 다운로드한 파일의 크기를 바이트 단위로 표시합니다.
- `requestParameters.x-amz-onBehalfOf`: 사용자 AWS Transfer Family 이름과 세션 ID를 포함하여 작업을 수행한 AWS Transfer Family 사용자를 추적할 수 있습니다.

`x-amz-onBehalfOf` 필드는 S3 API 호출을 작업을 시작한 특정 AWS Transfer Family 사용자와 다시 연결하므로 특히 중요합니다. 이 필드는 형식을 따릅니다 `username.sessionid@server-id`. 여기서

- `username`는 AWS Transfer Family 사용자 이름입니다.
- `sessionid`는 사용자 세션의 고유 식별자입니다.
- `server-id`는 AWS Transfer Family 서버의 ID입니다.

일반적인 데이터 액세스 작업

를 통해 데이터 액세스를 모니터링할 때 AWS Transfer Family 일반적으로 CloudTrail 로그에 다음과 같은 S3 API 작업이 표시됩니다.

AWS Transfer Family 로그의 일반적인 S3 작업

S3 API 작업	AWS Transfer Family 작업	설명
<code>GetObject</code>	파일 다운로드	사용자가 서버에서 파일을 다운로드했습니다.

S3 API 작업	AWS Transfer Family 작업	설명
PutObject	파일 업로드	사용자가 서버에 파일을 업로드했습니다.
DeleteObject	파일 삭제	사용자가 서버에서 파일을 삭제했습니다.
ListObjects 또는 ListObjectsV2	디렉터리 목록	디렉터리에 사용자가 나열한 파일
CopyObject	파일 복사	사용자가 서버 내에서 파일을 복사했습니다.

CloudTrail 로그에서 이러한 작업을 모니터링하면 AWS Transfer Family 서버를 통해 수행되는 모든 파일 활동을 추적하여 규정 준수 요구 사항을 충족하고 무단 액세스를 감지할 수 있습니다.

AWS Transfer Family 서버에 대한 Amazon CloudWatch 로깅

Amazon CloudWatch는를 포함한 AWS 리소스에 대한 포괄적인 가시성을 제공하는 강력한 모니터링 및 관찰성 서비스입니다 AWS Transfer Family.

- 실시간 모니터링: CloudWatch는 Transfer Family 리소스 및 애플리케이션을 실시간으로 모니터링하므로 성능을 추적하고 분석할 수 있습니다.
- 지표 수집: CloudWatch는 측정하여 분석에 사용할 수 있는 변수인 리소스 및 애플리케이션에 대한 다양한 지표를 수집하고 추적합니다.
- CloudWatch 홈 페이지: CloudWatch 홈 페이지에는 Transfer Family 및 사용하는 기타 AWS 서비스에 대한 지표가 자동으로 표시되므로 모니터링 데이터를 중앙에서 볼 수 있습니다.
- 사용자 지정 대시보드: CloudWatch에서 사용자 지정 대시보드를 생성하여 사용자 지정 애플리케이션 및 모니터링하도록 선택한 리소스와 관련된 지표를 표시할 수 있습니다.
- 경고 및 알림: CloudWatch를 사용하면 지표를 모니터링하고 특정 임계값 위반 시 알림 또는 자동화된 작업을 트리거하는 경보를 생성할 수 있습니다. 이는 Transfer Family 서버에서 파일 전송 활동을 모니터링하고 그에 따라 리소스를 조정하는 데 유용할 수 있습니다.
- 비용 최적화: CloudWatch에서 수집한 데이터를 사용하여 사용률이 낮은 리소스를 식별하고 인스턴스 중지 또는 삭제와 같은 조치를 취하여 비용을 최적화할 수 있습니다.

전반적으로 CloudWatch의 포괄적인 모니터링 기능은 Transfer Family 인프라와 해당 인프라에서 실행되는 애플리케이션을 관리하고 최적화하는 데 유용한 도구입니다.

Transfer Family 웹 앱의 CloudWatch 로깅에 대한 세부 정보는에서 확인할 수 있습니다[Transfer Family 웹 앱에 대한 CloudTrail 로깅](#).

Transfer Family에 대한 CloudWatch 로깅 유형

Transfer Family는 CloudWatch에 이벤트를 로깅하는 두 가지 방법을 제공합니다.

- JSON 정형 로깅
- 로깅 역할을 통한 로깅

Transfer Family 서버의 경우 원하는 로깅 메커니즘을 선택할 수 있습니다. 커넥터 및 워크플로의 경우 로깅 역할만 지원됩니다.

JSON 구조화 로깅

서버 이벤트 로깅의 경우 JSON 구조화된 로깅을 사용하는 것이 좋습니다. 이렇게 하면 CloudWatch 로그 쿼리를 활성화하는 보다 포괄적인 로깅 형식이 제공됩니다. 이러한 유형의 로깅의 경우 서버를 생성 (또는 서버의 로깅 구성을 편집)하는 사용자의 IAM 정책에 다음 권한이 포함되어야 합니다.

- logs:CreateLogDelivery
- logs>DeleteLogDelivery
- logs:DescribeLogGroups
- logs:DescribeResourcePolicies
- logs:GetLogDelivery
- logs:ListLogDeliveries
- logs:PutResourcePolicy
- logs:UpdateLogDelivery

다음은 예제 정책입니다.

```
{
  "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": "*"
    }
  ]
}
```

JSON 구조화 로깅 설정에 대한 자세한 내용은 [섹션을 참조하세요](#) [서버에 대한 로깅 생성, 업데이트 및 보기](#).

로깅 역할

서버에 연결된 관리형 워크플로와 커넥터에 대한 이벤트를 로깅하려면 로깅 역할을 지정해야 합니다. 액세스 권한을 설정하려면 리소스 기반 IAM 정책과 해당 액세스 정보를 제공하는 IAM 역할을 만듭니다. 다음은 서버 이벤트를 로깅할 수 있는 AWS 계정 있는에 대한 정책 예제입니다.

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

워크플로 이벤트를 로깅하도록 로깅 역할을 구성하는 방법에 대한 자세한 내용은 [섹션을 참조하세요](#) [워크플로에 대한 로깅 관리](#).

서버에 대한 로깅 생성, 업데이트 및 보기

모든 AWS Transfer Family 서버에 대해 구조화된 로깅을 제공합니다. 모든 신규 및 기존 Transfer Family 서버에 대해 구조화된 로깅을 사용하는 것이 좋습니다. 구조화된 로깅 사용의 이점은 다음과 같습니다.

- 구조화된 JSON 형식으로 로그를 수신합니다.
- JSON 형식의 필드를 자동으로 검색하는 Amazon CloudWatch 로그 인사이트를 사용하여 로그를 쿼리할 수 있습니다.
- AWS Transfer Family 리소스 간에 로그 그룹을 공유하면 여러 서버의 로그 스트림을 단일 로그 그룹으로 결합하여 모니터링 구성 및 로그 보존 설정을 더 쉽게 관리할 수 있습니다.
- CloudWatch 대시보드에 추가할 수 있는 집계된 지표와 시각화를 생성합니다.

- 로그 그룹을 사용하여 통합된 로그 지표, 시각화 및 대시보드를 생성하여 사용량 및 성능 데이터를 추적할 수 있습니다.

서버에 연결된 워크플로에 대한 로깅을 활성화하려면 로깅 역할을 사용해야 합니다.

Note

로깅 역할을 추가할 때 로깅 그룹은 항상 `/aws/transfer/your-serverID` 수 있습니다. 즉, 구조화된 서버 로그를 동일한 그룹에 전송하지 않는 한 두 개의 개별 로깅 그룹에 로깅하게 됩니다.

워크플로를 서버와 연결하려는 경우 로깅 역할을 추가해야 하는 경우의 기본 로그 그룹에 로깅하도록 구조화된 로깅을 설정할 수 있습니다 `/aws/transfer/your-serverID`.

로깅 그룹을 수정하려면 AWS Transfer Family API 참조의 [StructuredLogDestinations](#)를 참조하세요.

Transfer Family 콘솔을 사용하여 새 서버를 생성할 경우 기본값으로 로깅이 활성화됩니다. 서버를 생성한 후 UpdateServer API 작업을 사용하여 로깅 구성을 변경할 수 있습니다. 자세한 내용은 [StructuredLogDestinations](#)를 참조하세요.

현재 워크플로의 경우 로깅을 활성화하려면 로깅 역할을 지정해야 합니다.

- CreateServer 또는 UpdateServer API 작업을 사용하여 워크플로를 서버에 연결하면 시스템에서 로깅 역할을 자동으로 생성하지 않습니다. 워크플로 이벤트를 기록하려면 서버에 로깅 역할을 명시적으로 연결해야 합니다.
- Transfer Family 콘솔을 사용하여 서버를 생성하고 워크플로를 연결하는 경우 이름에 서버 ID가 포함된 로그 그룹에 로그가 전송됩니다. 형식은 `/aws/transfer/server-id`이며, 예를 들면 `/aws/transfer/s-1111aaaa2222bbbb3`와 같습니다. 서버 로그는 동일한 로그 그룹이나 다른 로그 그룹으로 전송될 수 있습니다.

콘솔에서 서버를 만들고 편집할 때 고려할 로깅 고려 사항

- 콘솔을 통해 만든 새 서버는 워크플로가 서버에 연결되어 있지 않는 한 구조화된 JSON 로깅만 지원합니다.
- 콘솔에서 새로 만든 서버의 경우 로깅 없음을 선택할 수 없습니다.
- 기존 서버는 언제든지 콘솔을 통해 구조화된 JSON 로깅을 활성화할 수 있습니다.

- 콘솔을 통해 구조화된 JSON 로깅을 활성화하면 기존 로깅 방법이 비활성화되므로 고객에게 요금이 두 배로 청구되지 않습니다. 단, 워크플로가 서버에 연결된 경우는 예외입니다.
- 구조화된 JSON 로깅을 활성화하면 나중에 콘솔을 통해 비활성화할 수 없습니다.
- 구조화된 JSON 로깅을 활성화하면 언제든지 콘솔을 통해 로그 그룹 대상을 변경할 수 있습니다.
- 구조화된 JSON 로깅을 활성화한 경우 API를 통해 두 로깅 타입을 모두 활성화한 경우 콘솔을 통해 로깅 역할을 편집할 수 없습니다. 단, 워크플로가 서버에 연결된 경우는 예외입니다. 하지만 로깅 역할은 추가 세부 정보에 계속 표시됩니다.

API 또는 SDK를 사용하여 서버를 만들고 편집할 때 로깅 고려 사항

- API를 통해 새 서버를 만드는 경우 로깅 타입 중 하나 또는 두 가지를 모두 구성하거나 로깅 없음을 선택할 수 있습니다.
- 기존 서버의 경우 언제든지 구조화된 JSON 로깅을 활성화 및 비활성화할 수 있습니다.
- API를 통해 언제든지 로그 그룹을 변경할 수 있습니다.
- API를 통해 언제든지 로깅 역할을 변경할 수 있습니다.

구조적 로깅을 활성화하려면 다음 권한이 있는 계정에 로그인해야 합니다

- logs:CreateLogDelivery
- logs>DeleteLogDelivery
- logs:DescribeLogGroups
- logs:DescribeResourcePolicies
- logs:GetLogDelivery
- logs:ListLogDeliveries
- logs:PutResourcePolicy
- logs:UpdateLogDelivery

정책 예제는 단원에서 확인할 수 있습니다 [CloudWatch 로깅 역할 구성](#).

주제

- [서버에 대한 로깅 생성](#)
- [서버의 로깅 업데이트](#)
- [서버 구성 보기](#)

서버에 대한 로깅 생성

새 서버를 만들 때 추가 세부 정보 구성 페이지에서 기존 로그 그룹을 지정하거나 새 로그 그룹을 만들 수 있습니다.

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.

로그 그룹 생성을 선택하면 CloudWatch 콘솔 (<https://console.aws.amazon.com/cloudwatch/>)에서 로그 그룹 생성 페이지가 열립니다. 자세한 내용은 [CloudWatch Logs의 로그 그룹 생성](#)을 참조하세요.

서버의 로깅 업데이트

로깅 세부 정보는 업데이트 시나리오에 따라 다릅니다.

Note

구조화된 JSON 로깅을 선택하면 드문 경우지만 Transfer Family에서 이전 형식의 로깅을 중지하지만 새 JSON 형식으로 로깅을 시작하는 데 시간이 걸리는 지연이 발생할 수 있습니다. 이로 인해 이벤트가 로그되지 않을 수 있습니다. 서비스 중단은 없겠지만 로깅 방법을 변경한 후 처음 1시간 동안은 로그가 삭제될 수 있으므로 파일 전송에 주의해야 합니다.

기존 서버를 편집하는 경우 옵션은 서버 상태에 따라 달라집니다.

- 서버에는 이미 로깅 역할이 활성화되어 있지만 구조화된 JSON 로깅은 활성화되어 있지 않습니다.

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 [↗](#)

i 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 ▼



i Workflows events will be delivered to a log group labelled with the server ID.

- 서버에 로깅이 활성화되지 않았습니다.

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 ▼



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

- 서버에는 이미 로깅 역할이 활성화되어 있지만 구조화된 JSON 로깅은 활성화되어 있지 않습니다.

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 ▼



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

- 서버에는 이미 구조화된 JSON 로깅이 활성화되어 있으며 지정된 로딩 역할도 있습니다.

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/s-



Create log group [↗](#)

Logging Role [Info](#)

Select an existing role from your account

AWSTransferLoggingAccess



[i](#) Workflows events will be delivered to a log group labelled with the server ID.

서버 구성 보기

서버 구성 페이지의 세부 정보는 시나리오에 따라 다릅니다.

시나리오에 따라 서버 구성 페이지는 다음 예 중 하나와 비슷할 수 있습니다.

- 로깅이 활성화되지 않았습니다.

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:

Workflow for partial uploads

-

TLS session resumption

Managed workflows execution role

-

Passive IP

Security Policy [Info](#)

TransferSecurityPolicy-2018-11

- 구조화된 JSON 로깅이 활성화되었습니다.

Additional details
Edit

Log group /aws/transfer/s	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:	Workflow for partial uploads -	TLS session resumption -
Security Policy Info TransferSecurityPolicy-2020-06	Managed workflows execution role -	Passive IP -

- 로깅 역할은 활성화되지만 구조화된 JSON 로깅은 활성화되지 않습니다.

Additional details
Edit

Log group -	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:lx39/	Workflow for partial uploads -	TLS session resumption -
Security Policy Info TransferSecurityPolicy-2018-11	Managed workflows execution role execution-role	Passive IP -

- 두 가지 타입의 로깅(로깅 역할 및 구조화된 JSON 로깅)이 모두 활성화됩니다.

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 -

워크플로에 대한 로깅 관리

CloudWatch는 워크플로 진행 상황 및 결과에 대한 통합 감사 및 로깅을 제공합니다. 또한 워크플로에 대한 여러 지표를 AWS Transfer Family 제공합니다. 이전 1분 간 시작, 성공적으로 완료, 및 실패한 워크플로 실행 수에 대한 지표를 볼 수 있습니다. Transfer Family에 대한 모든 CloudWatch 지표는 [Transfer Family 서버에 CloudWatch 지표 사용](#)에 설명되어 있습니다.

워크플로에 대한 Amazon CloudWatch Logs 보기

1. <https://console.aws.amazon.com/cloudwatch/>에서 Amazon CloudWatch 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 로그를 선택한 다음, 로그 그룹을 선택합니다.
3. 로그 그룹 페이지의 탐색 모음에서 AWS Transfer Family 서버에 적합한 리전을 선택합니다.
4. 서버에 해당하는 로그 그룹을 선택합니다.

예를 들어 서버 ID가 s-1234567890abcdef0인 경우 로그 그룹은 /aws/transfer/s-1234567890abcdef0입니다.

5. 서버의 로그 그룹 세부 정보 페이지에는 가장 최근의 로그 스트림이 표시됩니다. 탐색 중인 사용자에게 대한 로그 스트림은 두 가지입니다.

- 각 Secure Shell(SSH) File Transfer 프로토콜(SFTP) 세션별 로그 스트림
- 서버에서 실행 중인 워크플로의 로그 스트림 워크플로의 로그 스트림 형식은 **username.workflowID.uniqueStreamSuffix**입니다.

예를 들어 사용자가 mary-major이면 다음과 같은 로그 스트림이 있습니다.

```
mary-major-east.1234567890abcdef0
mary.w-abcdef01234567890.021345abcdef6789
```

Note

이 예에 나열된 16자리 영숫자 식별자는 가상의 식별자입니다. Amazon CloudWatch에 표시되는 값은 다릅니다.

mary-major-usa-east.1234567890abcdef0의 로그 이벤트 페이지에는 각 사용자 세션의 세부 정보가 표시되고 mary.w-abcdef01234567890.021345abcdef6789 로그 스트림에는 워크플로에 대한 세부 정보가 포함됩니다.

다음은 복사 단계가 포함된 워크플로(w-abcdef01234567890)를 기반으로 하는 mary.w-abcdef01234567890.021345abcdef6789의 샘플 로그 스트림입니다.

```
{
  "type": "ExecutionStarted",
  "details": {
    "input": {
      "initialFileLocation": {
        "bucket": "amzn-s3-demo-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": "amzn-s3-demo-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 로깅 역할 구성

액세스 권한을 설정하려면 리소스 기반 IAM 정책과 해당 액세스 정보를 제공하는 IAM 역할을 만듭니다.

Amazon CloudWatch 로깅을 사용 설정하려면 먼저 CloudWatch 로깅을 사용 설정하는 IAM 정책을 만들어야 합니다. 그런 다음 IAM 역할을 만들고 이 역할에 정책을 첨부합니다. 이렇게 하려면 [서버를 생성](#)하거나 [기존 서버를 편집](#)해야 합니다. 자세한 설명은 Amazon CloudWatch 사용자 가이드의 [Amazon CloudWatch란 무엇인가?](#)와 [Amazon CloudWatch Logs란 무엇인가?](#)를 참조하세요.

다음 예 IAM 정책을 사용하여 CloudWatch 로깅을 허용하세요.

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": "*"
}
]
}

```

위의 예 정책에서, **Resource**의 경우 *region-id* 및 *AWS ##*를 맞춤 값으로 대체하세요.

예: **"Resource": "arn:aws::logs:us-east-1:111122223333:log-group:/aws/transfer/*"**

그런 다음 역할을 생성하고 만든 CloudWatch 로그 정책을 연결합니다.

IAM 역할을 생성하여 정책을 연결하려면

1. 탐색 창에서 역할을 선택한 후 역할 생성을 선택합니다.

역할 생성 페이지에서 AWS 서비스가 선택되어 있는지 확인합니다.

2. 서비스 목록에서 전송을 선택하고, 다음: 권한을 선택합니다. 이렇게 하면 AWS Transfer Family와 IAM 역할 간에 신뢰 관계가 설정됩니다. 또한 `aws:SourceAccount` 및 `aws:SourceArn` 조건 키를 추가하여 혼동되는 부정 문제로부터 자신을 보호하세요. 자세한 설명은 다음 설명서를 참조하세요:

- 다음과 신뢰 관계를 설정하기 위한 절차 AWS Transfer Family: [신뢰 관계를 구축하기 위해](#)
- 혼동된 대리인 문제에 대한 설명: [혼동된 대리인 문제](#)

3. 권한 정책 연결 섹션에서, 방금 생성한 CloudWatch 로그 정책을 찾아 선택한 다음 다음: 태그를 선택합니다.
4. (옵션) 태그의 키와 값을 입력하고 Next: Review(다음: 검토)를 선택합니다.
5. 검토 페이지에 새 역할의 명칭과 설명을 입력한 다음 역할 생성을 선택합니다.
6. 로그를 보려면 Server ID(서버 ID)를 선택하여 서버 구성 페이지를 열고 로그 보기를 선택합니다. 로그 스트림을 볼 수 있는 CloudWatch 콘솔로 리디렉션됩니다.

서버의 CloudWatch 페이지에서는 사용자 인증(성공 및 실패), 데이터 업로드(PUT 작업), 데이터 다운로드(GET 작업) 레코드를 확인할 수 있습니다.

Transfer Family 로그 스트림 보기

Transfer Family 서버 로그를 보려면

1. 서버의 세부 정보 페이지로 이동합니다.
2. 로그 보기를 선택합니다. 그러면 Amazon CloudWatch가 열립니다.
3. 선택한 서버의 로그 그룹이 표시됩니다.

The screenshot shows the AWS CloudWatch console interface. On the left, the 'Logs' section is expanded, and 'Log groups' is selected. The main content area displays the details for a specific log group under the path '/aws/transfer/s-'. The 'Log group details' section includes the ARN, creation time (2 years ago), retention (Never expire), and stored bytes (39.39 MB). Below this, there are tabs for 'Log streams', 'Metric filters', 'Subscription filters', 'Contributor Insights', 'Tags', and 'Data protection - new'. The 'Log streams' tab is currently selected, showing a list of log streams with columns for 'Log stream' and 'Last event time'.

4. 로그 스트림을 선택하여 스트림의 세부 정보 및 개별 항목을 표시할 수 있습니다.
 - 오류 목록이 있는 경우 해당 목록을 선택하여 서버의 최신 오류에 대한 세부 정보를 볼 수 있습니다.

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

1m 30m 1h 12h Custom

▶	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...	Copy
▼	2023-03-23T16:12:56.857-04:00	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=	Copy
▶	2023-03-23T16:13:00.106-04:00	ERRORS AUTH_FAILURE Method=password User=debian Message= SourceIP=	

- 예 로그 스트림을 보려면 다른 항목을 선택하세요.

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

1m 30m 1h 12h Custom

▶	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			

- 서버에 연결된 관리형 워크플로가 있는 경우 워크플로 실행에 대한 로그를 볼 수 있습니다.

Note

워크플로의 로그 스트림 형식은 `username.workflowId.uniqueStreamSuffix`입니다. 예를 들어 `decrypt-user.w-a1111222233334444.aa1111bb2222`는 사용자 **decrypt-user** 및 워크플로 **w-a1111222233334444**의 로그 스트림 명칭일 수 있습니다.

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](#)

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-user/test.json.gpg", "versionId": "...", "etag": "..." }, "stepType": "DECRYPT", "stepName": "decrypt-step" }, "workflowId": "w-...", "executionId": "...", "transferDetails": { "serverId": "s-...", "username": "decrypt-user", "sessionId": "..." } } }</code>
2023-03-21T14:12:03.464-04:00	<code>{"type": "StepCompleted", "details": {"output": {}}, "stepType": "DECRYPT", "stepName": "decrypt-step"}, "workflowId": "w-..."</code>

Note

확장된 로그 항목의 경우 복사를 선택하여 항목을 클립보드에 복사할 수 있습니다. CloudWatch 로그에 대한 자세한 설명은 [로그 데이터 보기](#)를 참조하세요.

Amazon CloudWatch 경보 생성

다음 예제에서는 지표 AWS Transfer Family 를 사용하여 Amazon CloudWatch 경보를 생성하는 방법을 보여줍니다FilesIn.

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,
});
```

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
```

Amazon S3 S3 API 작업 로깅

Note

이 섹션은 Transfer Family 웹 앱에는 적용되지 않습니다.

파일 전송 사용자를 대신하여 이루어진 [S3 요청을 식별하기 위해 Amazon S3 액세스 로그를 사용](#)하는 경우, 파일 전송을 처리하는 데 맡겨진 IAM 역할을 표시하는 데 RoleSessionName이(가) 사용 됩니다. 또한 전송에 사용된 사용자 이름, 세션 ID, 서버 ID와 같은 추가 정보도 표시됩니다. 형식은 [AWS:Role Unique Identifier]/username.sessionid@server-id(이)며, 요청자 필드에 포함되어 있습니다. 예를 들어, 다음은 S3 버킷으로 복사된 파일에 대한 S3 액세스 로그의 샘플 요청자 필드 콘텐츠입니다.

```
arn:aws:sts::AWS-Account-ID:assumed-role/IamRoleName/
username.sessionid@server-id
```

위의 요청자 필드에는 IamRoleName(이)라고 불리는 IAM 역할이 표시됩니다. IAM 역할 고유 식별자에 대한 자세한 설명은 AWS Identity and Access Management 사용자 가이드에서 [고유 식별자](#)를 참조하세요.

혼동된 대리자 문제를 제한하는 예

혼동된 대리자 문제는 작업을 수행할 권한이 없는 엔터티가 권한이 더 많은 엔터티에게 작업을 수행하도록 강요할 수 있는 보안 문제입니다. 에서 AWS교차 서비스 가장은 혼동된 대리자 문제를 초래할 수 있습니다. 자세한 내용은 [교차 서비스 혼동된 대리자 방지](#)를 참조하세요.

Note

다음 예에서는 자신의 정보로 각각의 **### ## ## ###**를 바꿉니다.

이 예제에서는 서버에 워크플로가 연결되어 있지 않은 경우 워크플로의 ARN 세부 정보를 제거할 수 있습니다.

다음 예제 로깅/호출 정책은 계정의 모든 서버(및 워크플로)가 역할을 수임하도록 허용합니다.

```
{
```

```

"Version": "2012-10-17",
"Statement": [
  {
    "Sid": "AllowAllServersWithWorkflowAttached",
    "Effect": "Allow",
    "Principal": {
      "Service": "transfer.amazonaws.com"
    },
    "Action": "sts:AssumeRole",
    "Condition": {
      "StringEquals": {
        "aws:SourceAccount": "111122223333"
      },
      "ArnLike": {
        "aws:SourceArn": [
          "arn:aws:transfer:us-west-2:111122223333:server/*",
          "arn:aws:transfer:us-west-2:111122223333:workflow/*"
        ]
      }
    }
  }
]
}

```

다음 예제 로깅/호출 정책은 특정 서버(및 워크플로)가 역할을 수임하도록 허용합니다.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowSpecificServerWithWorkflowAttached",
      "Effect": "Allow",
      "Principal": {
        "Service": "transfer.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "111122223333"
        },
        "ArnEquals": {
          "aws:SourceArn": [

```

```
    "arn:aws:transfer:us-west-2:111122223333:server/server-id",
    "arn:aws:transfer:us-west-2:111122223333:workflow/workflow-  

id"
  ]
}
}
```

Transfer Family의 CloudWatch 로그 구조

이 주제에서는 JSON 구조화된 로그 항목과 레거시 로그 항목 모두에 대해 Transfer Family 로그에 채워진 필드를 설명합니다.

주제

- [Transfer Family에 대한 JSON 구조화된 로그](#)
- [Transfer Family에 대한 레거시 로그](#)

Transfer Family에 대한 JSON 구조화된 로그

다음 표에는 Transfer Family SFTP/FTP/FTPS 작업의 로그 항목 필드에 대한 세부 정보가 새 JSON 구조화 로그 형식으로 나와 있습니다.

필드	설명	입력 예
activity-type	The action by the user	사용 가능한 활동 유형 은 AUTH_FAILURE , , CONNECTED , DISCONNEC TED , ERROR, , EXIT_REAS ON , CLOSE, DELETE, CREATE_SYMLINK MKDIR, OPEN, PARTIAL_CLOSE , RENAME, RMDIR, SETSTAT, 입 니다TLS_RESUME_FAILURE

필드	설명	입력 예
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 암호화 알고리즘)	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 have a logical home directory, this value is always /	/user-home-bucket/test
kex	Specifies the negotiated SSH key exchange (KEX) for the connection (available KEX are listed in 암호화 알고리즘)	diffie-hellman-group14-sha256
message	Provides more information related to the error	<###>
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	/amzn-s3-demo-bucket/test-file-1.pdf

필드	설명	입력 예
ssh-public-key	The public key body for the user that is connecting	AAAAC3NzaC1lZDI1NT E5AAAAIA9OY0qV6XYV HaaOiWAcj2spDJVbgj rqDPY4pxd6GnHl
ssh-public-key-fingerprint	사용자 키를 나열할 때 서비스 관리 사용자를 위한 콘솔에 표시된 퍼블릭 키 지문입니다.	SHA256:BY3gNMHwTfj d4n2VuT4pTyLOk82zW Zj4KEYEu7y4r/0
 Note 콘솔에서 지문은 패딩 문자(있는 경우): 끝에 0~3개의 등호(=)로 표시됩니다. 로그 항목에서 이 패딩은 출력에서 제거됩니다.		
ssh-public-key-type	Type of public key: Transfer Family supports RSA-, ECDSA-, and ED25519-formatted keys	ssh-ed25519
resource-arn	A system-assigned, unique identifier for a specific resource (for example, a server)	arn:aws:transfer:ap-northeast-1:12346789012:server/s-1234567890akeu2js2
role	The IAM role of the user	arn:aws:iam::0293883675:role/testuser-role
session-id	A system-assigned, unique identifier for a single session	9ca9a0e1cec6ad9d
source-ip	Client IP address	18.323.0.129

필드	설명	입력 예
user	The end user's username	myname192
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

Transfer Family에 대한 레거시 로그

다음 표에는 다양한 Transfer Family 작업에 대한 로그 항목에 대한 세부 정보가 나와 있습니다.

Note

이러한 항목은 새 JSON 구조화된 로그 형식이 아닙니다.

다음 표에는 새로운 JSON 구조화된 로그 형식의 다양한 Transfer Family 작업에 대한 로그 항목에 대한 세부 정보가 포함되어 있습니다.

작업	Amazon CloudWatch Logs 내의 해당 로그
인증 실패 횟수	ERRORS AUTH_FAILURE Method=publickey User=lhr Message="RSA SHA256:Lfz3R2nmLY4raK+b7Rb1rSvUIbAE+a+Hxg0c7l1JIZ0" SourceIP=3.8.172.211
복사/태그 지정/삭제/복호화 워크플로	{"type":"StepStarted","details":{"input":{"fileLocation":{"backingStore":"EFS","filesystemId":"fs-12345678","path":"/lhr/regex.py"}}, "stepType":"TAG","stepName":"successful_tag_step"},"workflowId":"w-1111aaa2222bbbb3","executionId":"81234abcd-1234-efgh-5678-ijklmnopqr90","transferDetails":{"serv

작업	Amazon CloudWatch Logs 내의 해당 로그
	erId":"s-1234abcd5678efghi","username":"lhr", "sessionId":"1234567890abcdef0"}}
맞춤 단계 워크플로	{ "type":"CustomStepInvoked", "details":{ "output":{ "token":"MzM4Mjg5YWUtYTEzMy00YjIzLWI3OGMtYzU4OGI2ZjQyMzE5"}, "stepType":"CUSTOM", "stepName":"efs-s3_copy_2"}, "workflowId":"w-9283e49d33297c3f7", "executionId":"1234abcd-1234-efgh-5678-ijklmnopqr90", "transferDetails":{ "serverId":"s-zzzz1111aaaa22223", "username":"lhr", "sessionId":"1234567890abcdef0"} }}
삭제	lhr.33a8fb495ffb383b DELETE Path=/bucket/user/123.jpg
다운로드	lhr.33a8fb495ffb383b OPEN Path=/bucket/user/123.jpg Mode=READ llhr.33a8fb495ffb383b CLOSE Path=/bucket/user/123.jpg BytesOut=3618546
로그인/로그아웃	user.914984e553bcd6b6 CONNECTED SourceIP=1.22.111.222 User=lhr HomeDir=LOGICAL Client=SSH-2.0-OpenSSH_7.4 Role=arn:aws::iam::123456789012:role/sftp-s3-access user.914984e553bcd6b6 DISCONNECTED
명칭 바꾸기	lhr.33a8fb495ffb383b RENAME Path=/bucket/user/lambo.png NewPath=/bucket/user/ferrari.png

작업	Amazon CloudWatch Logs 내의 해당 로그
샘플 워크플로 오류 로그	<pre>{"type":"StepErrored","details":{"errorType":"BAD_REQUEST","errorMessage":"Cannot tag Efs file","stepType":"TAG","stepName":"successful_tag_step"},"workflowId":"w-1234abcd5678efghi","executionId":"81234abcd-1234-efgh-5678-ijklmnopqr90","transferDetails":{"serverId":"s-1234abcd5678efghi","username":"lhr","sessionId":"1234567890abcdef0"}}</pre>
Symlinks	<pre>lhr.eb49cf7b8651e6d5 CREATE_SYMLINK LinkPath=/fs-12345678/lhr/pqr.jpg TargetPath=abc.jpg</pre>
업로드	<pre>lhr.33a8fb495ffb383b OPEN Path=/bucket/user/123.jpg Mode=CREATE TRUNCATE WRITE lhr.33a8fb495ffb383b CLOSE Path=/bucket/user/123.jpg BytesIn=3618546</pre>

작업	Amazon CloudWatch Logs 내의 해당 로그
워크플로	<pre>{ "type": "ExecutionStarted", "details": { "input": { "initialFileLocation": { "backingStore": "EFS", "filesystemId": "fs-12345678", "path": "/lhr/regex.py" } }, "workflowId": "w-1111aaaa2222bbbb3", "executionId": "1234abcd-1234-efgh-5678-ijklmnopqr90", "transferDetails": { "serverId": "s-zzzz1111aaaa22223", "username": "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-1234-efgh-5678-ijklmnopqr90", "transferDetails": { "serverId": "s-18ca49dce5d842e0b", "username": "lhr", "sessionId": "1234567890abcdef0" } } }</pre>

CloudWatch 로그 항목 예

이 주제에서는 예제 로그 항목을 제공합니다.

주제

- [전송 세션 로그 항목의 예](#)
- [SFTP 커넥터의 로그 항목 예](#)
- [VPC Lattice 커넥터의 로그 항목 예제](#)
- [키 교환 알고리즘 실패에 대한 로그 항목 예제](#)

전송 세션 로그 항목의 예

이 예제에서 SFTP 사용자는 Transfer Family 서버에 연결하고 파일을 업로드한 다음 세션 연결을 해제합니다.

다음 로그 항목은 Transfer Family 서버에 연결하는 SFTP 사용자를 반영합니다.

```
{
  "role": "arn:aws:iam::500655546075:role/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": "/test/log-me",
  "ssh-public-key":
    "AAAAC3NzaC1lZDI1NTE5AAAAIA90Y0qV6XYVHaa0iWAcj2spDJVbgjrqpDPY4pxd6GnH1",
  "ssh-public-key-fingerprint": "SHA256:BY3gNMHwTfjd4n2VuT4pTyL0k82zWZj4KEYEu7y4r/0",
  "ssh-public-key-type": "ssh-ed25519",
  "user": "log-me",
  "kex": "ecdh-sha2-nistp256",
  "session-id": "9ca9a0e1cec6ad9d"
}
```

다음 로그 항목은 Amazon S3 버킷에 파일을 업로드하는 SFTP 사용자를 반영합니다.

```
{
  "mode": "CREATE|TRUNCATE|WRITE",
  "path": "/test/log-me/config-file",
  "activity-type": "OPEN",
  "resource-arn": "arn:aws:transfer:us-east-1:500655546075:server/s-3fe215d89f074ed2a",
  "session-id": "9ca9a0e1cec6ad9d"
}
```

다음 로그 항목은 SFTP 세션에서 연결 해제된 SFTP 사용자를 반영합니다. 먼저 클라이언트가 버킷에 대한 연결을 닫은 다음 클라이언트가 SFTP 세션의 연결을 끊습니다.

```
{
  "path": "/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"
  }

```

Note

사용 가능한 활동 유형은 AUTH_FAILURE, , CONNECTED, DISCONNECTED, ERROR, , EXIT_REASON, CLOSE, DELETE, CREATE_SYMLINKMKDIR, OPEN, PARTIAL_CLOSE, RENAME, RMDIR, SETSTAT, 입니다 TLS_RESUME_FAILURE.

SFTP 커넥터의 로그 항목 예

이 섹션에는 전송 성공 및 실패 모두에 대한 예제 로그가 포함되어 있습니다. 로그는 라는 로그 그룹에 생성됩니다. /aws/transfer/*connector-id* 여기서 *connector-id*는 SFTP 커넥터의 식별자입니다. StartFileTransfer 또는 StartDirectoryListing 명령을 실행하면 SFTP 커넥터에 대한 로그 항목이 생성됩니다.

이 로그 항목은 성공적으로 완료된 전송을 위한 것입니다.

```

{
  "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:account-id:connector/connector-id",
"local-directory-path": "/connectors-localbucket",
"bytes": 514,
"egress-type": "SERVICE_MANAGED"
}

```

이 로그 항목은 시간이 초과되어 성공적으로 완료되지 않은 전송을 위한 것입니다.

```

{
  "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:account-id:connector/connector-id",
  "local-directory-path": "/connectors-localbucket",
  "egress-type": "SERVICE_MANAGED"
}

```

이 로그 항목은 성공하는 SEND 작업을 위한 것입니다.

```

{
  "operation": "SEND",
  "timestamp": "2024-04-24T18:16:12.513207284Z",
  "connector-id": "connector-id",
  "transfer-id": "transfer-id",
  "file-transfer-id": "transfer-id/file-transfer-id",
  "url": "sftp://server-id.server.transfer.us-east-1.amazonaws.com",
  "file-path": "/amzn-s3-demo-bucket/my-test-folder/connector-metrics-us-east-1-2024-01-02.csv",
  "status-code": "COMPLETED",
  "start-time": "2024-04-24T18:16:12.295235884Z",
  "end-time": "2024-04-24T18:16:12.461840732Z",
  "account-id": "255443218509",
  "connector-arn": "arn:aws:transfer:us-east-1:account-id:connector/connector-id",
  "bytes": 275,
}

```



```
"egress-type": "SERVICE_MANAGED"
}
```

이전 로그 예제의 일부 키 필드에 대한 설명입니다.

- **timestamp**는 로그가 CloudWatch에 추가되는 시기를 나타냅니다. **start-time** 및는 커넥터가 실제로 전송을 시작하고 완료하는 시기를 **end-time** 나타냅니다.
- **transfer-id**는 각 **start-file-transfer** 요청에 할당된 고유 식별자입니다. 사용자가 단일 **start-file-transfer** API 작업에서 여러 파일 경로를 전달하는 경우 모든 파일은 동일한를 공유합니다 **transfer-id**.
- **file-transfer-id**는 전송된 각 파일에 대해 생성된 고유한 값입니다. 의 초기 부분은와 **file-transfer-id** 동일합니다 **transfer-id**.

VPC Lattice 커넥터의 로그 항목 예제

이 섹션에는 VPC Lattice 커넥터에 대한 예제 로그가 포함되어 있습니다. VPC Lattice 커넥터의 경우 로그에는 커넥터 구성 및 네트워크 설정에 대한 정보를 제공하는 추가 필드가 포함됩니다.

이 로그 항목은 성공적으로 완료된 VPC Lattice 커넥터 SEND 작업을 위한 것입니다.

```
{
  "operation": "SEND",
  "timestamp": "2025-09-05T14:20:19.577192454Z",
  "connector-id": "connector-id",
  "transfer-id": "transfer-id",
  "file-transfer-id": "transfer-id/file-transfer-id",
  "file-path": ""/amzn-s3-demo-bucket/my-test-folder/connector-vpc-lattice-us-east-1-2025-03-22.csv"",
  "status-code": "COMPLETED",
  "start-time": "2025-09-05T14:20:19.434072509Z",
  "end-time": "2025-09-05T14:20:19.481453346Z",
  "account-id": "account-id",
  "connector-arn": "arn:aws:transfer:us-east-1:account-id:connector/connector-id",
  "remote-directory-path": "/test-bucket/test-folder/",
  "bytes": 262,
  "egress-type": "VPC_LATTICE",
  "vpc-lattice-resource-configuration-arn": "arn:aws:vpc-lattice:us-east-1:account-id:resourceconfiguration/resource-configuration-arn-id",
  "vpc-lattice-port-number": 22
}
```

VPC Lattice 커넥터 로그에는 다음과 같은 추가 필드가 포함됩니다.

- `egress-type` - 커넥터의 송신 구성 유형
- `vpc-lattice-resource-configuration-arn` - 대상 SFTP 서버 위치를 정의하는 VPC Lattice 리소스 구성의 ARN
- `vpc-lattice-port-number` - VPC Lattice를 통해 SFTP 서버에 연결하기 위한 포트 번호

키 교환 알고리즘 실패에 대한 로그 항목 예제

이 섹션에는 키 교환 알고리즘(KEX)이 실패한 예제 로그가 포함되어 있습니다. 다음은 구조화된 로그에 대한 ERRORS 로그 스트림의 예입니다.

이 로그 항목은 호스트 키 유형 오류가 있는 예제입니다.

```
{
  "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"
}
```

이 로그 항목은 KEX 불일치가 있는 예제입니다.

```
{
  "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"
}
```

Transfer Family 서버에 CloudWatch 지표 사용

Note

Transfer Family 콘솔 자체에서도 Transfer Family에 대한 지표를 가져올 수 있습니다. 자세한 내용은 [콘솔 내 사용량 모니터링](#)를 참조하세요.

CloudWatch 지표를 사용하여 서버에 대한 정보를 얻을 수 있습니다. 지표는 CloudWatch에 게시된 시간 순서별 데이터 요소 집합을 나타냅니다. 지표를 사용할 때는 Transfer Family 네임스페이스, 지표 명칭 및 [차원](#)을 지정해야 합니다. 지표 수학에 대한 자세한 정보는 Amazon CloudWatch 사용자 가이드에서 [지표](#)를 참조하세요.

다음 표에서는 Transfer Family에 대한 CloudWatch 측정치를 설명합니다.

네임스페이스	지표	설명
AWS/Transfer	BytesIn	서버로 전송된 총 바이트 수. 보고 기준: - SFTP/FTP/FTPS의 경우: Transfer Family 서버에 대한 연결이 설정되는 동안 5분마다 내보내집니다. 해당 기간 동안 파일 또는 바이트가 전송되지 않으면 "0"이 발생합니다. - AS2의 경우: 고객이 AS2 서버에서 메시지를 수신하고 인바운드 메시지 처리가 완료되는 즉시 내보낸 경우 단위: 개 기간: 5분
	BytesOut	서버에서 전송된 총 바이트 수입니다. 보고 기준: SFTP/FTP/FTPS의 경우: Transfer Family 서버에 대한 연결이 설정되는 동안 5분마다 내보내집니다. 해당 기

네임스페이스	지표	설명
		간 동안 파일 또는 바이트가 전송되지 않으면 "0"이 발생합니다. AS2의 경우: 고객이 AS2 커넥터StartFile Transfer 에서를 호출하고 아웃바운드 메시지가 완료되는 즉시 내보낸 경우. 단위: 개 기간: 5분
	FilesIn	서버로 전송된 총 파일 수. AS2 프로토콜을 사용하는 서버의 경우 이 지표는 수신된 메시지 수를 나타냅니다. 보고 기준: - SFTP/FTP/FTPS의 경우: Transfer Family 서버에 대한 연결이 설정되는 동안 5분마다 내보내집니다. 해당 기간 동안 파일 또는 바이트가 전송되지 않으면 "0"이 발생합니다. - AS2의 경우: 고객이 AS2 서버에서 메시지를 수신하고 인바운드 메시지가 완료되는 즉시 내보낸 경우. 단위: 개 기간: 5분

네임스페이스	지표	설명
	FilesOut	서버에서 전송된 총 파일 수. 보고 기준: - SFTP/FTP/FTPS의 경우: Transfer Family 서버에 대한 연결이 설정되는 동안 5분마다 내보내집니다. 해당 기간 동안 파일 또는 바이트가 전송되지 않으면 "0"이 발생합니다. - AS2의 경우: 고객이 AS2 커넥터StartFile Transfer 에서를 호출하고 아웃바운드 메시지 처리가 완료되는 즉시 내보낸 경우. 단위: 개 기간: 5분
	InboundMessage	거래 파트너로부터 성공적으로 수신한 AS2 메시지의 총 개수. 보고 기준: 고객이 AS2 서버에서 메시지를 수신하고 인바운드 메시지 처리가 성공적으로 완료되는 즉시 내보낸 경우 단위: 개 기간: 5분

네임스페이스	지표	설명
	InboundFailedMessage	거래 파트너로부터 성공적으로 수신되지 못한 AS2 메시지의 총 개수. 즉, 거래 파트너가 메시지를 보냈지만 Transfer Family 서버가 메시지를 성공적으로 처리하지 못했습니다. 보고 기준: 고객이 AS2 서버에서 메시지를 수신하고 인바운드 메시지 처리가 성공적으로 완료되는 즉시 전송된 경우 단위: 개 기간: 5분
	OnUploadExecutionsStarted	서버에서 시작된 워크플로를 실행한 총 횟수. 보고 기준: 실행이 시작될 때마다 트리거됩니다. 단위: 개 기간: 1분
	OnUploadExecutionsSuccess	서버에서 성공적인 워크플로를 실행한 총 횟수. 보고 기준: 실행이 성공적으로 완료될 때마다 트리거됩니다. 단위: 개 기간: 1분
	OnUploadExecutionsFailed	서버에서 실패한 워크플로를 실행한 총 횟수. 보고 기준: 실행이 성공적으로 완료되지 않을 때마다 트리거됩니다. 단위: 개 기간: 1분

네임스페이스	지표	설명
	OutboundMessage	거래 파트너에게 성공적으로 전송된 AS2 메시지의 총 수입니다. 보고 기준: 고객이 AS2 커넥터StartFileTransfer 에서 전화를 걸어 아웃바운드 메시지 처리가 성공적으로 완료되는 즉시 내보낸 경우 단위: 개 기간: 5분
	OutboundFailedMessage	거래 파트너에게 성공적으로 전송되지 못한 AS2 메시지의 총 개수. 보고 기준: 고객이 AS2 커넥터StartFileTransfer 에서 전화를 걸어 아웃바운드 메시지 처리가 성공적으로 완료되는 즉시 내보낸 경우 단위: 개 기간: 5분

Transfer Family 크기

차원은 지표의 보안 인증에 속하는 명칭/값 쌍입니다. 차원에 대한 자세한 설명은 CloudWatch Developer Guide의 [차원](#)을 참조하세요.

다음 표에서는 Transfer Family의 CloudWatch 차원을 설명합니다.

차원	설명
ServerId	사용자의 고유 ID입니다.
ConnectorId	커넥터의 고유 ID입니다. AS2, OutboundMessage 및 OutboundFailedMessage

AWS 사용자 알림 와 함께 사용 AWS Transfer Family

AWS Transfer Family 이벤트에 대한 알림을 받으려면 [AWS 사용자 알림](#)를 사용하여 다양한 전송 채널을 설정할 수 있습니다. 이벤트가 지정한 규칙과 일치하면 알림을 받습니다.

이메일, [채팅 애플리케이션의 Amazon Q Developer](#) 채팅 알림 또는 [AWS Console Mobile Application](#) 푸시 알림을 비롯한 여러 채널을 통해 이벤트에 대한 알림을 받을 수 있습니다. [콘솔 알림 센터에서 알림을 볼 수도 있습니다](#). 사용자 알림 는 집계를 지원하므로 특정 이벤트 중에 수신하는 알림 수를 줄일 수 있습니다.

자세한 내용은 [AWS Transfer Family 관리형 워크플로를 사용하여 파일 전송 알림 사용자 지정](#) 블로그 게시물 및 [정의 AWS 사용자 알림](#) AWS 사용자 알림 사용 설명서의 .

쿼리를 사용하여 로그 항목 필터링

CloudWatch 쿼리를 사용하여 Transfer Family의 로그 항목을 필터링하고 식별할 수 있습니다. 이 단원에는 몇 가지 예제가 포함되어 있습니다.

1. 에 로그인 AWS Management Console 하고 <https://console.aws.amazon.com/cloudwatch/> CloudWatch 콘솔을 엽니다.
2. 쿼리 또는 규칙을 생성할 수 있습니다.
 - Logs Insights 쿼리를 생성하려면 왼쪽 탐색 패널에서 Logs Insights를 선택한 다음 쿼리에 대한 세부 정보를 입력합니다.
 - Contributor Insights 규칙을 생성하려면 왼쪽 탐색 패널에서 인사이트 > Contributor Insights를 선택한 다음 규칙의 세부 정보를 입력합니다.
3. 생성한 쿼리 또는 규칙을 실행합니다.

상위 인증 실패 기여자 보기

구조화된 로그에서 인증 실패 로그 항목은 다음과 비슷합니다.

```
{
  "method": "password",
  "activity-type": "AUTH_FAILURE",
  "source-ip": "999.999.999.999",
  "resource-arn": "arn:aws:transfer:us-east-1:999999999999:server/s-0123456789abcdef",
  "message": "Invalid user name or password",
```



```
"user": "exampleUser"
}
```

다음 쿼리를 실행하여 인증 실패에 대한 상위 기여자를 확인합니다.

```
filter @logStream = 'ERRORS'
| filter `activity-type` = 'AUTH_FAILURE'
| stats count() as AuthFailures by user, method
| sort by AuthFailures desc
| limit 10
```

CloudWatch Logs Insights를 사용하는 대신 CloudWatch Contributors Insights 규칙을 생성하여 인증 실패를 볼 수 있습니다. 다음과 유사한 규칙을 생성합니다.

```
{
  "AggregateOn": "Count",
  "Contribution": {
    "Filters": [
      {
        "Match": "$.activity-type",
        "In": [
          "AUTH_FAILURE"
        ]
      }
    ],
    "Keys": [
      "$.user"
    ]
  },
  "LogFormat": "JSON",
  "Schema": {
    "Name": "CloudWatchLogRule",
    "Version": 1
  },
  "LogGroupARNs": [
    "arn:aws:logs:us-east-1:999999999999:log-group:/customer/structured_logs"
  ]
}
```

파일이 열린 로그 항목 보기

구조화된 로그에서 파일 읽기 로그 항목은 다음과 비슷합니다.

```
{
  "mode": "READ",
  "path": "/fs-0df669c89d9bf7f45/avtester/example",
  "activity-type": "OPEN",
  "resource-arn": "arn:aws:transfer:us-east-1:999999999999:server/s-0123456789abcdef",
  "session-id": "0049cd844c7536c06a89"
}
```

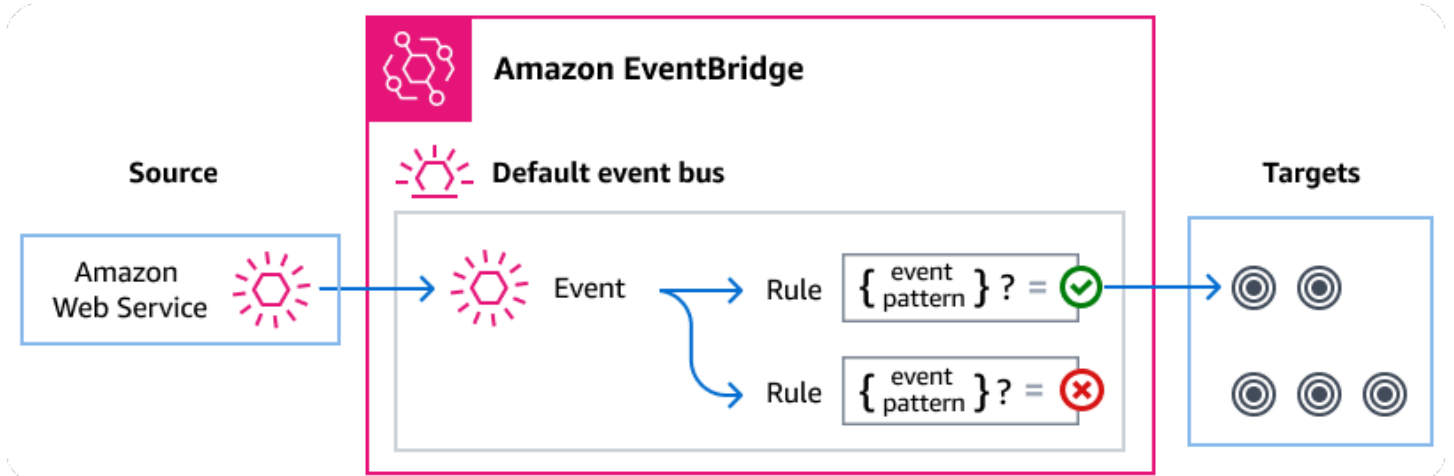
다음 쿼리를 실행하여 파일이 열렸음을 나타내는 로그 항목을 봅니다.

```
filter `activity-type` = 'OPEN'
| display @timestamp, @logStream, `session-id`, mode, path
```

를 사용하여 Transfer Family 이벤트 관리 Amazon EventBridge

Amazon EventBridge 는 이벤트를 사용하여 애플리케이션 구성 요소를 함께 연결하는 서버리스 서비스로, 확장 가능한 이벤트 기반 애플리케이션을 더 쉽게 구축할 수 있습니다. 이벤트 기반 아키텍처는 이벤트를 발생시키고 이에 응답함으로써 느슨하게 결합된 소프트웨어 시스템을 구축하는 방식입니다. 이벤트는 리소스나 환경의 변화를 나타냅니다.

많은 AWS 서비스와 마찬가지로 이벤트는 Transfer Family 생성하고 EventBridge 기본 이벤트 버스로 전송합니다. 기본 이벤트 버스는 모든 AWS 계정에 자동으로 프로비저닝됩니다. 이벤트 버스는 이벤트를 수신하여 0개 이상의 목적지 또는 대상에 전달하는 라우터입니다. 이벤트가 도착할 때 이벤트를 평가하는 이벤트 버스에 대한 규칙을 지정합니다. 각 규칙은 이벤트가 규칙의 이벤트 패턴과 일치하는지 여부를 확인합니다. 이벤트가 일치하는 경우 이벤트 버스는 이벤트를 하나 이상의 지정된 대상으로 보냅니다.



주제

- [Transfer Family 이벤트](#)
- [EventBridge 규칙을 사용하여 Transfer Family 이벤트 전송](#)
- [Amazon EventBridge 권한](#)
- [추가 EventBridge 리소스](#)
- [Transfer Family 이벤트 세부 정보 참조](#)

Transfer Family 이벤트

Transfer Family 는 이벤트를 기본 EventBridge 이벤트 버스로 자동 전송합니다. 각 규칙에 이벤트 패턴과 하나 이상의 대상이 포함된 이벤트 버스에 규칙을 생성할 수 있습니다.

규칙의 이벤트 패턴과 일치하는 이벤트는 최선의 노력 또는 내구성을 기반으로 지정된 대상으로 전달됩니다(일부 이벤트는 순서에 맞지 않게 전달될 수 있음). 이러한 전송 수준은 Amazon EventBridge [이벤트 참조의 AWS 서비스 이벤트에 대한 전송 수준에](#) 설명되어 있습니다.

- SFTP, FTPS 및 FTP 서버에 대한 서버 수준 이벤트는 최선을 다해 전달됩니다.
- SFTP 커넥터 이벤트는 내구성을 기반으로 전달됩니다.
- AS2 이벤트는 내구성을 기반으로 전달됩니다.

다음 이벤트는에서 생성됩니다 Transfer Family. 자세한 내용을 알아보려면 Amazon EventBridge 사용 설명서의 [EventBridge 이벤트](#)를 참조하세요.

SFTP, FTPS 및 FTP 서버 이벤트

다음 표에는 이벤트 유형별로 구성된 SFTP, FTPS 및 FTP 서버의 이벤트가 나열되어 있습니다.

파일 업로드 및 다운로드 이벤트

이벤트 세부 정보 유형	설명
FTP 서버 파일 다운로드 완료	FTP 프로토콜에 대한 파일이 성공적으로 다운로드되었습니다.
FTP 서버 파일 다운로드 실패	FTP 프로토콜에 대한 파일 다운로드 시도가 실패했습니다.
FTP 서버 파일 업로드 완료	FTP 프로토콜에 대한 파일이 성공적으로 업로드되었습니다.
FTP 서버 파일 업로드 실패	FTP 프로토콜에 대한 파일 업로드 시도가 실패했습니다.
FTPS 서버 파일 다운로드 완료	FTPS 프로토콜에 대한 파일이 성공적으로 다운로드되었습니다.
FTPS 서버 파일 다운로드 실패	FTPS 프로토콜에 대한 파일 다운로드 시도가 실패했습니다.
FTPS 서버 파일 업로드 완료	FTPS 프로토콜에 대한 파일이 성공적으로 업로드되었습니다.
FTPS 서버 파일 업로드 실패	FTPS 프로토콜에 대한 파일 업로드 시도가 실패했습니다.

이벤트 세부 정보 유형	설명
SFTP 서버 파일 다운로드 완료	SFTP 프로토콜에 대한 파일이 성공적으로 다운로드되었습니다.
SFTP 서버 파일 다운로드 실패	SFTP 프로토콜에 대한 파일 다운로드 시도가 실패했습니다.
SFTP 서버 파일 업로드 완료	SFTP 프로토콜에 대한 파일이 성공적으로 업로드되었습니다.
SFTP 서버 파일 업로드 실패	SFTP 프로토콜에 대한 파일 업로드 시도가 실패했습니다.

기타 파일 작업 이벤트

이벤트 세부 정보 유형	설명
FTP 서버 디렉터리 생성 완료	FTP 프로토콜에 대한 디렉터리가 성공적으로 생성되었습니다.
FTP 서버 디렉터리 생성 실패	FTP 프로토콜에 대한 디렉터리 생성 시도가 실패했습니다.
FTP 서버 디렉터리 삭제 완료	FTP 프로토콜에 대한 디렉터리가 성공적으로 삭제되었습니다.
FTP 서버 디렉터리 삭제 실패	FTP 프로토콜에 대한 디렉터리 삭제 시도가 실패했습니다.
FTP 서버 파일 삭제 완료	FTP 프로토콜에 대한 파일이 성공적으로 삭제되었습니다.
FTP 서버 파일 삭제 실패	FTP 프로토콜에 대한 파일 삭제 시도가 실패했습니다.
FTP 서버 파일 이름 변경 완료	FTP 프로토콜의 파일 이름이 성공적으로 변경되었습니다.
FTP 서버 파일 이름 변경 실패	FTP 프로토콜에 대한 파일 이름 변경 시도가 실패했습니다.
FTPS 서버 디렉터리 생성 완료	FTPS 프로토콜에 대한 디렉터리가 성공적으로 생성되었습니다.
FTPS 서버 디렉터리 생성 실패	FTPS 프로토콜에 대한 디렉터리 생성 시도가 실패했습니다.
FTPS 서버 디렉터리 삭제 완료	FTPS 프로토콜에 대한 디렉터리가 성공적으로 삭제되었습니다.
FTPS 서버 디렉터리 삭제 실패	FTPS 프로토콜에 대한 디렉터리 삭제 시도가 실패했습니다.
FTPS 서버 파일 삭제 완료	FTPS 프로토콜에 대한 파일이 성공적으로 삭제되었습니다.

이벤트 세부 정보 유형	설명
FTPS 서버 파일 삭제 실패	FTPS 프로토콜에 대한 파일 삭제 시도가 실패했습니다.
FTPS 서버 파일 이름 변경 완료	FTPS 프로토콜의 파일 이름이 성공적으로 변경되었습니다.
FTPS 서버 파일 이름 변경 실패	FTPS 프로토콜에 대한 파일 이름 변경 시도가 실패했습니다.
SFTP 서버 디렉터리 생성 완료	SFTP 프로토콜에 대한 디렉터리가 성공적으로 생성되었습니다.
SFTP 서버 디렉터리 생성 실패	SFTP 프로토콜에 대한 디렉터리 생성 시도가 실패했습니다.
SFTP 서버 디렉터리 삭제 완료	SFTP 프로토콜에 대한 디렉터리가 성공적으로 삭제되었습니다.
SFTP 서버 디렉터리 삭제 실패	SFTP 프로토콜에 대한 디렉터리 삭제 시도가 실패했습니다.
SFTP 서버 파일 삭제 완료	SFTP 프로토콜에 대한 파일이 성공적으로 삭제되었습니다.
SFTP 서버 파일 삭제 실패	SFTP 프로토콜에 대한 파일 삭제 시도가 실패했습니다.
SFTP 서버 파일 이름 변경 완료	SFTP 프로토콜의 파일 이름이 성공적으로 변경되었습니다.
SFTP 서버 파일 이름 변경 실패	SFTP 프로토콜에 대한 파일 이름 변경 시도가 실패했습니다.

SFTP 커넥터 이벤트

Note

이러한 이벤트는 Amazon EventBridge 이벤트 참조의 [AWS 서비스 이벤트에 대한 전송 수준에 설명된 대로 내구성 있는 수준으로](#) EventBridge에 전달됩니다. EventBridge

이벤트 세부 정보 유형	설명
SFTP 커넥터 파일 전송 완료	커넥터에서 원격 SFTP 서버로의 파일 전송이 성공적으로 완료되었습니다.
SFTP 커넥터 파일 전송 실패	커넥터에서 원격 SFTP 서버로의 파일 전송이 실패했습니다.
SFTP 커넥터 파일 검색 완료	원격 SFTP 서버에서 커넥터로의 파일 전송이 성공적으로 완료되었습니다.
SFTP 커넥터 파일 검색 실패	원격 SFTP 서버에서 커넥터로의 파일 전송이 실패했습니다.
SFTP 커넥터 디렉터리 목록 작성 완료	성공적으로 완료된 호출을 나열하는 시작 파일 디렉터리입니다.
SFTP 커넥터 디렉터리 나열 실패	실패한 시작 파일 디렉터리 목록입니다.
SFTP 커넥터 원격 이동 완료	파일 또는 디렉터리가 원격 서버에서 성공적으로 이동되었거나 이름이 변경되었습니다.
SFTP 커넥터 원격 이동 실패	원격 서버에서 파일 또는 디렉터를 이동하거나 이름을 바꾸지 못했습니다.
SFTP 커넥터 원격 삭제 완료	원격 서버에서 파일 또는 디렉터리가 성공적으로 삭제되었습니다.
SFTP 커넥터 원격 삭제 실패	원격 서버에서 파일 또는 디렉터를 삭제하지 못했습니다.

AS2 이벤트

Note

이러한 이벤트는 Amazon EventBridge 이벤트 참조의 [AWS 서비스 이벤트에 대한 전송 수준에 설명된 대로 내구성 있는 수준으로](#) EventBridge에 전달됩니다. EventBridge

이벤트 세부 정보 유형	설명
AS2 페이로드 수신 완료	AS2 메시지의 페이로드가 수신되었습니다.
AS2 페이로드 수신 실패	AS2 메시지의 페이로드가 수신되지 않았습니다.
AS2 페이로드 전송 완료	AS2 메시지의 페이로드가 성공적으로 전송되었습니다.
AS2 페이로드 전송 실패	AS2 메시지의 페이로드가 전송되지 않았습니다.
AS2 MDN 수신 완료	AS2 메시지에 대한 메시지 처리 알림이 수신되었습니다.
AS2 MDN 수신 실패	AS2 메시지에 대한 메시지 처리 알림이 수신되지 않았습니다.
AS2 MDN 전송 완료	AS2 메시지에 대한 메시지 처리 알림이 성공적으로 전송되었습니다.
AS2 MDN 전송 실패	AS2 메시지에 대한 메시지 처리 알림이 전송되지 않았습니다.

EventBridge 규칙을 사용하여 Transfer Family 이벤트 전송

EventBridge 기본 이벤트 버스가 대상에 Transfer Family 이벤트를 보내도록 하려면 원하는 이벤트의 데이터와 일치하는 Transfer Family 이벤트 패턴이 포함된 규칙을 생성해야 합니다.

Amazon EventBridge에서 AWS Transfer Family 이벤트를 캡처하려면

1. 에 로그인 AWS Management Console 하고 <https://console.aws.amazon.com/events/> Amazon EventBridge 콘솔을 엽니다.
2. 탐색 창에서 규칙을 선택한 다음 규칙 생성을 선택합니다.
3. 규칙에 대한 설명이 포함된 이름을 입력하고 선택적으로 설명을 입력합니다.
4. 규칙 유형에서 이벤트 패턴이 있는 규칙을 선택한 후 다음을 선택합니다.
5. 이벤트 소스 섹션에서 AWS 이벤트 또는 EventBridge 파트너 이벤트를 선택합니다.
6. 생성 방법 섹션에서 패턴 양식 사용을 선택합니다.
7. 이벤트 패턴 섹션에서 다음 정보를 제공합니다.
 - a. 이벤트 소스에서 AWS 서비스를 선택합니다.
 - b. AWS 서비스에서 전송을 선택합니다.

- c. 이벤트 유형에서 규칙을 트리거할 Transfer Family 이벤트 유형을 선택합니다.

이벤트 유형 선택에 따라 이벤트 유형 사양 1 섹션이 표시될 수 있습니다.
 - d. 이벤트 유형 사양 1 섹션이 표시되면 캡처하려는 특정 이벤트를 선택합니다(또는 임의의 이벤트를 선택하여 선택한 이벤트 유형에 대한 모든 이벤트를 캡처).
 - e. (선택 사항) 이벤트 패턴 편집기를 사용하여 이벤트 세부 정보에 대한 필터를 지정합니다.
 - f. 다음을 선택합니다.
8. 대상 선택에서 사용할 수 있는 선택 항목 중에서 대상을 선택합니다. 다음 사용 가능한 대상 중에서 선택합니다.
- AWS 서비스. 인기 있는 옵션은 서버리스 컴퓨팅을 위한 Lambda 함수, 메시지 처리를 위한 Amazon SQS 대기열, 알림을 위한 Amazon SNS 주제, AWS Step Functions 워크플로 오케스트레이션입니다.
 - EventBridge API 대상. 외부의 HTTP 엔드포인트로 이벤트를 보내려면 API 대상을 대상으로 사용할 AWS 수 있습니다.
 - EventBridge 이벤트 버스. 동일한 계정 및 리전 또는 다른 계정 또는 리전의 다른 이벤트 버스로 이벤트를 보낼 수 있습니다.

이벤트 버스 규칙 생성에 대한 포괄적인 지침은 Amazon EventBridge 사용 설명서의 [Creating rules that react to events](#)를 참조하세요.

대상 선택에 대한 도움말은 Amazon EventBridge 사용 설명서의 [대상 선택](#)을 참조하세요.

9. 대상에 대한 추가 옵션을 구성한 후 다음을 선택합니다.
10. (선택 사항) 규칙에 태그를 추가하고 다음을 선택합니다.
11. 검토 및 생성 화면에서 모든 것이 양호한 경우 규칙 생성을 선택합니다.

이벤트에 대한 Transfer Family 이벤트 패턴 생성

가 이벤트를 기본 이벤트 버스로 Transfer Family 전송할 때는 각 규칙에 대해 정의된 이벤트 패턴을 EventBridge 사용하여 이벤트를 규칙의 대상으로 전달해야 하는지 여부를 결정합니다. 이벤트 패턴이 원하는 Transfer Family 이벤트의 데이터와 일치합니다. 각 이벤트 패턴은 다음을 포함하는 JSON 객체입니다.

- 이벤트를 전송하는 서비스를 식별하는 `source` 속성입니다. Transfer Family 이벤트의 경우 소스는 `aws.transfer`.

- (선택 사항) 일치시킬 이벤트 유형의 배열을 포함하는 `detail-type` 속성입니다.
- (선택 사항) 일치시킬 다른 이벤트 데이터를 포함하는 `detail` 속성입니다.

예를 들어 다음 이벤트 패턴은의 모든 이벤트와 일치합니다. Transfer Family

```
{
  "source": ["aws.transfer"]
}
```

다음 이벤트 패턴 예제는 모든 SFTP 커넥터 이벤트와 일치합니다.

```
{
  "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"]
}
```

다음 이벤트 패턴 예제는 모든 Transfer Family 실패 이벤트와 일치합니다.

```
{
  "source": ["aws.transfer"],
  "detail-type": [{"wildcard", "*Failed"}]
}
```

다음 이벤트 패턴 예제는 사용자 `### ##`에 대한 성공적인 SFTP 다운로드와 일치합니다.

```
{
  "source": ["aws.transfer"],
  "detail-type": ["SFTP Server File Download Completed"],
  "detail": {
    "username": [username]
  }
}
```

자세한 내용은 EventBridge 사용 설명서의 [이벤트 패턴](#)을 참조하세요.

에서 이벤트에 대한 Transfer Family 이벤트 패턴 테스트 EventBridge

EventBridge 샌드박스를 사용하면 규칙을 생성하거나 편집하는 광범위한 프로세스를 완료할 필요 없이 이벤트 패턴을 빠르게 정의하고 테스트할 수 있습니다. 샌드박스를 사용하여 이벤트 패턴을 정의하고 샘플 이벤트를 사용하여 패턴이 원하는 이벤트와 일치하는지 확인할 수 있습니다. EventBridge 샌드박스에서 직접 해당 이벤트 패턴을 사용하여 새 규칙을 생성할 수 있는 옵션을 제공합니다.

자세한 내용은 EventBridge 사용 설명서의 [이 EventBridge 샌드박스를 사용하여 이벤트 패턴 테스트를 참조](#) 하세요.

Amazon EventBridge 권한

Transfer Family 에는 이벤트를 전달하는 데 추가 권한이 필요하지 않습니다 Amazon EventBridge.

지정하는 대상에는 특정 권한 또는 구성이 필요할 수 있습니다. 대상에 특정 서비스를 사용하는 방법에 대한 자세한 내용은 Amazon EventBridge 사용 설명서의 [Amazon EventBridge 대상](#)을 참조하세요.

추가 EventBridge 리소스

를 사용하여 이벤트를 처리하고 관리하는 방법에 대한 자세한 내용은 [Amazon EventBridge 사용 설명서](#)의 다음 주제를 참조 EventBridge 하세요.

- 이벤트 버스의 작동 방식에 대한 자세한 내용은 [Amazon EventBridge 이벤트 버스](#)를 참조하세요.
- 이벤트 구조에 대해 자세히 알아보려면 [이벤트](#)를 참조하세요.
- 가 이벤트를 규칙과 일치 EventBridge 시킬 때 사용할 이벤트 패턴을 구성하는 방법에 대한 자세한 내용은 [이벤트 패턴](#)을 참조하세요.
- EventBridge 에서 처리하는 이벤트를 지정하는 규칙을 생성하는 방법에 대한 자세한 내용은 [규칙](#)을 참조하세요.
- 가 일치하는 이벤트를 EventBridge 전송하는 서비스 또는 기타 대상을 지정하는 방법에 대한 자세한 내용은 [대상을 참조](#)하세요.

Transfer Family 이벤트 세부 정보 참조

AWS 서비스의 모든 이벤트에는 이벤트에 대한 메타데이터가 포함된 공통 필드 세트가 있습니다. 이러한 메타데이터에는 이벤트의 소스인 AWS 서비스, 이벤트가 생성된 시간, 이벤트가 발생한 계정 및 리

전 등이 포함될 수 있습니다. 이러한 일반 필드에 대한 정의는 Amazon EventBridge 사용 설명서의 [이벤트 구조 참조](#)를 참조하세요.

또한 각 이벤트에는 해당 특정 이벤트와 관련된 데이터를 포함하는 detail 필드가 있습니다. 다음 참조에서는 다양한 Transfer Family 이벤트에 대한 세부 정보 필드를 정의합니다.

EventBridge 를 사용하여 Transfer Family 이벤트를 선택하고 관리할 때는 다음 사항을 고려하세요.

- 의 모든 이벤트에 대한 source 필드는 로 Transfer Family 설정됩니다aws.transfer.
- detail-type 필드는 이벤트 유형을 지정합니다.

예를 들어 FTP Server File Download Completed입니다.

- detail 필드는 해당 특정 이벤트와 관련된 데이터를 포함합니다.

Transfer Family 이벤트와 일치하는 규칙을 활성화하는 이벤트 패턴을 구성하는 방법에 대한 자세한 내용은 Amazon EventBridge 사용 안내서의 [이벤트 패턴](#)을 참조하세요.

이벤트 및 이벤트 EventBridge 처리 방법에 대한 자세한 내용은 Amazon EventBridge 사용 설명서의 [Amazon EventBridge 이벤트를](#) 참조하세요.

주제

- [SFTP, FTPS 및 FTP 서버 이벤트](#)
- [SFTP 커넥터 이벤트](#)
- [AS2 이벤트](#)

SFTP, FTPS 및 FTP 서버 이벤트

다음은 SFTP, FTPS 및 FTP 서버 이벤트에 대한 세부 정보 필드입니다.

- FTP 서버 디렉터리 생성 완료
- FTP 서버 디렉터리 생성 실패
- FTP 서버 디렉터리 삭제 완료
- FTP 서버 디렉터리 삭제 실패
- FTP 서버 파일 삭제 완료
- FTP 서버 파일 삭제 실패
- FTP 서버 파일 다운로드 완료

- FTP 서버 파일 다운로드 실패
- FTP 서버 파일 이름 변경 완료
- FTP 서버 파일 이름 변경 실패
- FTP 서버 파일 업로드 완료
- FTP 서버 파일 업로드 실패
- FTPS 서버 디렉터리 생성 완료
- FTPS 서버 디렉터리 생성 실패
- FTPS 서버 디렉터리 삭제 완료
- FTPS 서버 디렉터리 삭제 실패
- FTPS 서버 파일 삭제 완료
- FTPS 서버 파일 삭제 실패
- FTPS 서버 파일 다운로드 완료
- FTPS 서버 파일 다운로드 실패
- FTPS 서버 파일 이름 변경 완료
- FTPS 서버 파일 이름 변경 실패
- FTPS 서버 파일 업로드 완료
- FTPS 서버 파일 업로드 실패
- SFTP 서버 디렉터리 생성 완료
- SFTP 서버 디렉터리 생성 실패
- SFTP 서버 디렉터리 삭제 완료
- SFTP 서버 디렉터리 삭제 실패
- SFTP 서버 파일 삭제 완료
- SFTP 서버 파일 삭제 실패
- SFTP 서버 파일 다운로드 완료
- SFTP 서버 파일 다운로드 실패
- SFTP 서버 파일 이름 변경 완료
- SFTP 서버 파일 이름 변경 실패
- SFTP 서버 파일 업로드 완료
- SFTP 서버 파일 업로드 실패

source 및 detail-type 필드는 Transfer Family 이벤트에 대한 특정 값을 포함하므로 아래에 포함되어 있습니다. 모든 이벤트에 포함된 다른 메타데이터 필드의 정의는 Amazon EventBridge 사용 설명서의 [이벤트 구조 참조](#)를 참조하세요.

```
{
  . . . ,
  "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",
    "original-file-path" : "string",
    "renamed-file-path" : "string",
    "directory-path" : "string",
    "server-id" : "string",
    "username" : "string",
    "session-id" : "string",
    "start-timestamp" : "string"
  }
}
```

detail-type

이벤트의 유형을 식별합니다.

이 이벤트의 경우 값은 이전에 나열된 SFTP, FTPS 또는 FTP 서버 이벤트 이름 중 하나입니다.

source

이벤트를 생성한 서비스를 식별합니다. Transfer Family 이벤트의 경우 이 값은 `aws.transfer`.

detail

이벤트에 대한 정보를 포함하는 JSON 객체입니다. 이벤트를 생성하는 서비스에 따라 이 필드의 내용이 결정됩니다.

이 이벤트의 경우 데이터에는 다음이 포함됩니다.

`failure-code`

전송이 실패한 이유에 대한 범주입니다. 값: PARTIAL_UPLOAD | PARTIAL_DOWNLOAD | UNKNOWN_ERROR

`status-code`

전송 성공 여부입니다. 값: COMPLETED | FAILED.

`protocol`

전송에 사용되는 프로토콜입니다. 값: SFTP | FTPS | FTP

`bytes`

전송되는 바이트 수입니다.

`client-ip`

전송과 관련된 클라이언트의 IP 주소

`failure-message`

전송이 실패한 경우 전송이 실패한 이유에 대한 세부 정보입니다.

`end-timestamp`

전송에 성공하려면 파일 처리가 완료된 시점의 타임스탬프입니다.

`etag`

개체 태그(Amazon S3 파일에만 사용됨).

`file-path`

전송, 삭제 또는 달리 운영되는 파일의 경로입니다.

`original-file-path`

파일 이름 바꾸기 이벤트의 경우 이름을 바꾸기 전에 파일의 원래 경로입니다.

`renamed-file-path`

파일 이름 변경 이벤트의 경우 이름 변경 후 파일의 새 경로입니다.

`directory-path`

디렉터리 생성 및 삭제 이벤트의 경우 디렉터리의 경로입니다.

server-id

Transfer Family 서버의 고유 ID입니다.

username

전송을 수행하는 사용자입니다.

session-id

전송 세션의 고유 식별자입니다.

start-timestamp

성공적인 전송을 위해 파일 처리가 시작되는 시점의 타임스탬프입니다.

Example SFTP 서버 파일 다운로드 실패 예제 이벤트

다음 예제는 SFTP 서버에서 다운로드가 실패한 이벤트를 보여줍니다(사용 중인 스토리지 Amazon EFS).

```
{
  "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 FTP 서버 파일 업로드 완료됨 예제 이벤트

다음 예제는 FTP 서버에서 업로드가 성공적으로 완료된 이벤트를 보여줍니다(사용 중인 스토리지 Amazon S3).

```
{
  "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": "/amzn-s3-demo-bucket/test/1mb_file",
    "server-id": "s-1111aaaa2222bbbb3",
    "username": "test",
    "session-id": "event-ID",
    "start-timestamp": "2024-01-29T16:31:42.462088327Z"
  }
}
```

Example SFTP 서버 파일 삭제 완료됨 예제 이벤트

다음 예제는 SFTP 서버에서 파일이 성공적으로 삭제된 이벤트를 보여줍니다.

```
{
  "version": "0",
  "id": "event-ID",
  "detail-type": "SFTP Server File Delete Completed",
  "source": "aws.transfer",
```

```

"account": "958412138249",
"time": "2025-05-15T14:30:27Z",
"region": "us-east-1",
"resources": [
  "arn:aws:transfer:us-east-1:958412138249:server/s-1234abcd5678efghi"
],
"detail": {
  "status-code": "COMPLETED",
  "protocol": "SFTP",
  "client-ip": "IP-address",
  "end-timestamp": "2025-05-15T14:30:27.749749117Z",
  "file-path": "/fs-1234abcd5678efghi/user0/test-file-to-delete.txt",
  "server-id": "s-1234abcd5678efghi",
  "username": "test",
  "session-id": "session-ID",
  "start-timestamp": "2025-05-15T14:30:26.706282454Z"
}
}

```

Example SFTP 서버 파일 이름 바꾸기 완료된 예제 이벤트

다음 예제는 SFTP 서버에서 파일 이름이 성공적으로 변경된 이벤트를 보여줍니다.

```

{
  "version": "0",
  "id": "event-ID",
  "detail-type": "SFTP Server File Rename Completed",
  "source": "aws.transfer",
  "account": "958412138249",
  "time": "2025-05-15T15:45:12Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:transfer:us-east-1:958412138249:server/s-1234abcd5678efghi"
  ],
  "detail": {
    "status-code": "COMPLETED",
    "protocol": "SFTP",
    "client-ip": "IP-address",
    "end-timestamp": "2025-05-15T15:45:12.749749117Z",
    "original-file-path": "/fs-1234abcd5678efghi/user0/old-filename.txt",
    "renamed-file-path": "/fs-1234abcd5678efghi/user0/new-filename.txt",
    "server-id": "s-1234abcd5678efghi",
    "username": "test",
    "session-id": "session-ID",
  }
}

```

```

    "start-timestamp": "2025-05-15T15:45:11.706282454Z"
  }
}

```

Example SFTP Server Directory 완료된 예제 이벤트 생성

다음 예제는 SFTP 서버에서 디렉터리가 성공적으로 생성된 이벤트를 보여줍니다.

```

{
  "version": "0",
  "id": "event-ID",
  "detail-type": "SFTP Server Directory Create Completed",
  "source": "aws.transfer",
  "account": "958412138249",
  "time": "2025-05-15T16:20:05Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:transfer:us-east-1:958412138249:server/s-1234abcd5678efghi"
  ],
  "detail": {
    "status-code": "COMPLETED",
    "protocol": "SFTP",
    "client-ip": "IP-address",
    "end-timestamp": "2025-05-15T16:20:05.749749117Z",
    "directory-path": "/fs-1234abcd5678efghi/user0/new-directory",
    "server-id": "s-1234abcd5678efghi",
    "username": "test",
    "session-id": "session-ID",
    "start-timestamp": "2025-05-15T16:20:04.706282454Z"
  }
}

```

SFTP 커넥터 이벤트

Note

이러한 이벤트는 Amazon EventBridge 이벤트 참조의 [AWS 서비스 이벤트에 대한 전송 수준에 설명된 대로 내구성 있는 수준으로](#) EventBridge에 전달됩니다. EventBridge

다음은 SFTP 커넥터 이벤트에 대한 세부 정보 필드입니다.

- SFTP 커넥터 파일 전송 완료
- SFTP 커넥터 파일 전송 실패
- SFTP 커넥터 파일 검색 완료
- SFTP 커넥터 파일 검색 실패
- SFTP 커넥터 디렉터리 목록 작성 완료
- SFTP 커넥터 디렉터리 나열 실패
- SFTP 커넥터 원격 이동 완료
- SFTP 커넥터 원격 이동 실패
- SFTP 커넥터 원격 삭제 완료
- SFTP 커넥터 원격 삭제 실패

source 및 detail-type 필드는 Transfer Family 이벤트에 대한 특정 값을 포함하므로 아래에 포함되어 있습니다. 모든 이벤트에 포함된 다른 메타데이터 필드의 정의는 Amazon EventBridge 사용 설명서의 [이벤트 구조 참조](#)를 참조하세요.

```
{
  . . . ,
  "detail-type": "string",
  "source": "aws.transfer",
  . . . ,
  "detail": {
    "operation" : "string",
    "max-items" : "number",
    "connector-id" : "string",
    "output-directory-path" : "string",
    "listing-id" : "string",
    "transfer-id" : "string",
    "file-transfer-id" : "string",
    "url" : "string",
    "file-path" : "string",
    "status-code" : "string",
    "failure-code" : "string",
    "failure-message" : "string",
    "start-timestamp" : "string",
    "end-timestamp" : "string",
    "local-directory-path" : "string",
    "remote-directory-path" : "string",
    "item-count" : "number"
```

```

    "truncated" : "boolean"
    "bytes" : "number",
    "egress-type" : "string",
    "vpc-lattice-resource-configuration-arn" : "string",
    "vpc-lattice-port-number" : "number",
    "local-file-location" : {
        "domain" : "string",
        "bucket" : "string",
        "key" : "string"
    },
    "output-file-location" : {
        "domain" : "string",
        "bucket" : "string",
        "key" : "string"
    }
}
}
}

```

detail-type

이벤트의 유형을 식별합니다.

이 이벤트의 경우 값은 이전에 나열된 SFTP 커넥터 이벤트 이름 중 하나입니다.

source

이벤트를 생성한 서비스를 식별합니다. Transfer Family 이벤트의 경우 이 값은 `aws.transfer`입니다.

detail

이벤트에 대한 정보를 포함하는 JSON 객체입니다. 이벤트를 생성하는 서비스는 이 필드의 내용을 결정합니다.

이 이벤트의 경우 데이터에는 다음이 포함됩니다.

max-items

반환할 디렉터리/파일 이름의 최대 수입니다.

operation

StartFileTransfer 요청이 파일을 보내는지 아니면 가져오는지 여부입니다. 값: SEND | RETRIEVE.

connector-id

사용 중인 SFTP 커넥터의 고유 식별자입니다.

output-directory-path

파일/디렉터리 목록의 결과를 저장할 Amazon S3의 경로(버킷 및 접두사)입니다.

listing-id

StartDirectoryListing API 작업의 고유 식별자입니다. 이 식별자를 사용하여 CloudWatch 로그를 확인하여 요청 나열 상태를 확인할 수 있습니다.

transfer-id

전송 이벤트(StartFileTransfer요청)의 고유 식별자입니다.

file-transfer-id

전송 중인 파일의 고유 식별자입니다.

url

파트너의 AS2 또는 SFTP 엔드포인트의 URL입니다.

file-path

전송 또는 검색 중인 위치 및 파일입니다.

status-code

전송 성공 여부입니다. 값: FAILED | COMPLETED.

failure-code

전송이 실패한 경우 전송이 실패한 이유 코드입니다. 값: ASSUME_ROLE_ERROR | UNSUPPORTED_OPERATION_ERROR | CONNECTION_ERROR | READ_FILE_ERROR | WRITE_FILE_ERROR | TIMEOUT_ERROR | LIST_DIRECTORY_ERROR | SEND_FILE_NOT_FOUND | RETRIEVE_FILE_NOT_FOUND | DIRECTORY_NOT_FOUND | FILE_NOT_FOUND | PATH_NOT_FOUND | RENAME_ERROR | DELETE_ERROR | INTERNAL_ERROR | UNKNOWN_ERROR.

failure-message

전송 실패의 경우 전송 실패 이유에 대한 세부 정보입니다.

start-timestamp

전송에 성공하려면 파일 처리가 시작되는 시점의 타임스탬프입니다.

end-timestamp

전송에 성공하려면 파일 처리가 완료될 때의 타임스탬프입니다.

local-directory-path

RETRIEVE 요청의 경우 검색된 파일을 배치할 위치입니다.

remote-directory-path

SEND 요청의 경우 파트너의 SFTP 서버에 파일을 배치할 파일 디렉터리입니다. 사용자가 StartFileTransfer 요청에 전달 RemoteDirectoryPath 한의 값입니다. 파트너의 SFTP 서버에서 기본 디렉터를 지정할 수 있습니다. 그렇다면이 필드는 비어 있습니다.

item-count

목록 요청에 대해 반환된 항목(디렉터리 및 파일) 수입니다.

truncated

목록 출력에 원격 디렉터리에 포함된 모든 항목이 포함되어 있는지 여부입니다.

bytes

전송 중인 바이트 수입니다. 전송 실패의 경우 값은 0입니다.

egress-type

커넥터의 송신 구성 유형입니다. 값: SERVICE_MANAGED 또는 VPC_LATTICE.

vpc-lattice-resource-configuration-arn

대상 SFTP 서버 위치를 정의하는 VPC_LATTICE 리소스 구성의 ARN입니다. 서비스 관리형 커넥터의 경우이 필드는 null입니다.

vpc-lattice-port-number

VPC_LATTICE를 통해 SFTP 서버에 연결하기 위한 포트 번호입니다.

local-file-location

이 파라미터에는 AWS 스토리지 파일의 위치에 대한 세부 정보가 포함되어 있습니다.

domain

사용 중인 스토리지입니다. 현재 유일한 값은 s3입니다.

bucket

Amazon S3의 객체에 대한 컨테이너입니다.

key

Amazon S3의 객체에 할당된 이름입니다.

output-file-location

이 파라미터에는 디렉터리 목록의 결과를 스토리지에 저장할 위치의 세부 정보가 포함되어 있습니다 AWS .

domain

사용 중인 스토리지입니다. 현재 유일한 값은 입니다S3.

bucket

Amazon S3의 객체에 대한 컨테이너입니다.

key

Amazon S3의 객체에 할당된 이름입니다.

Example SFTP 커넥터 파일 전송 실패 예제 이벤트

다음 예제는 원격 SFTP 서버로 파일을 전송하려고 시도하는 동안 SFTP 커넥터가 실패한 이벤트를 보여줍니다.

```
{
  "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": "/amzn-s3-demo-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": "amzn-s3-demo-bucket",
      "key": "testfile.txt"
    }
  }
}

```

Example SFTP 커넥터 파일 검색 완료된 예제 이벤트

다음 예제는 SFTP 커넥터가 원격 SFTP 서버에서 전송된 파일을 성공적으로 검색한 이벤트를 보여줍니다.

```

{
  "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": "/amzn-s3-demo-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": "amzn-s3-demo-bucket",
        "key": "testfile.txt"
    }
}

```

Example SFTP 커넥터 디렉터리 목록 완료됨 예제 이벤트

다음 예제는 시작 디렉터리 목록 호출이 원격 SFTP 서버에서 목록 파일을 검색한 이벤트를 보여줍니다.

```

{
  "version": "0",
  "id": "event-ID",
  "detail-type": "SFTP Connector Directory Listing Completed",
  "source": "aws.transfer",
  "account": "123456789012",
  "time": "2024-01-24T18:28:08Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:transfer:us-east-1:123456789012:connector/c-f1111aaaa2222bbbb3"
  ],
  "detail": {
    "max-items": 10000,
    "connector-id": "c-fc68000012345aa18",
    "output-directory-path": "/amzn-s3-demo-bucket/example/file-listing-output",
    "listing-id": "123456-23aa-7980-abc1-1a2b3c4d5e",
    "url": "sftp://s-21a23456789012a.server.transfer.us-east-1.amazonaws.com",

    "status-code": "COMPLETED",
    "remote-directory-path": "/home",
    "item-count": 10000,
    "truncated": true,
    "start-timestamp": "2024-01-24T18:28:07.632388Z",
    "end-timestamp": "2024-01-24T18:28:07.774898Z",
    "output-file-location": {
      "domain": "S3",
      "bucket": "amzn-s3-demo-bucket",
      "key": "c-fc1ab90fd0d047e7a-70987273-49nn-4006-bab1-1a7290cc412ba.json"
    }
  }
}

```

AS2 이벤트

Note

이러한 이벤트는 Amazon EventBridge 이벤트 참조의 [AWS 서비스 이벤트에 대한 전송 수준에](#) 설명된 대로 내구성 있는 수준으로 EventBridge에 전달됩니다. EventBridge

다음은 AS2 이벤트에 대한 세부 정보 필드입니다.

- AS2 페이로드 수신 완료
- AS2 페이로드 수신 실패
- AS2 페이로드 전송 완료
- AS2 페이로드 전송 실패
- AS2 MDN 수신 완료
- AS2 MDN 수신 실패
- AS2 MDN 전송 완료
- AS2 MDN 전송 실패

source 및 detail-type 필드는 Transfer Family 이벤트에 대한 특정 값을 포함하므로 아래에 포함되어 있습니다. 모든 이벤트에 포함된 다른 메타데이터 필드의 정의는 Amazon EventBridge 사용 설명서의 [이벤트 구조 참조](#)를 참조하세요.

```
{
  . . . ,
  "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

이벤트의 유형을 식별합니다.

이 이벤트의 경우 값은 이전에 나열된 AS2 이벤트 중 하나입니다.

source

이벤트를 생성한 서비스를 식별합니다. Transfer Family 이벤트의 경우 이 값은 `aws.transfer`입니다.

detail

이벤트에 대한 정보를 포함하는 JSON 객체입니다. 이벤트를 생성하는 서비스에 따라 이 필드의 내용이 결정됩니다.

s3-attributes

전송 중인 파일의 Amazon S3 버킷과 키를 식별합니다. MDN 이벤트의 경우 MDN 파일의 버킷과 키도 식별합니다.

file-bucket

Amazon S3의 객체에 대한 컨테이너입니다.

file-key

Amazon S3의 객체에 할당된 이름입니다.

json-bucket

COMPLETED 또는 FAILED 전송의 경우 JSON 파일의 컨테이너입니다.

json-key

COMPLETED 또는 FAILED 전송의 경우 Amazon S3의 JSON 파일에 할당된 이름입니다.

mdn-bucket

MDN 이벤트의 경우 MDN 파일의 컨테이너입니다.

mdn-key

MDN 이벤트의 경우 Amazon S3의 MDN 파일에 할당된 이름입니다.

mdn-subject

MDN 이벤트의 경우 메시지 처리에 대한 텍스트 설명입니다.

mdn-message-id

MDN 이벤트의 경우 MDN 메시지의 고유 ID입니다.

disposition

MDN 이벤트의 경우 처리 범주입니다.

bytes

메시지의 바이트 수입니다.

as2-from

메시지를 보내는 AS2 거래 파트너입니다.

as2-message-id

전송 중인 AS2 메시지의 고유 식별자입니다.

as2-to

메시지를 수신하는 AS2 거래 파트너입니다.

connector-id

Transfer Family 서버에서 거래 파트너로 전송되는 AS2 메시지의 경우 사용 중인 AS2 커넥터의 고유 식별자입니다.

client-ip

서버 이벤트(거래 파트너에서 Transfer Family 서버로 전송)의 경우 전송과 관련된 클라이언트의 IP 주소입니다.

agreement-id

서버 이벤트의 경우 AS2 계약의 고유 식별자입니다.

server-id

서버 이벤트의 경우 Transfer Family 서버의 고유 ID만 해당됩니다.

requester-file-name

페이로드 이벤트의 경우 전송 중에 수신된 파일의 원래 이름입니다.

message-subject

메시지 제목에 대한 텍스트 설명입니다.

start-timestamp

전송에 성공하려면 파일 처리가 시작되는 시점의 타임스탬프입니다.

end-timestamp

전송에 성공하려면 파일 처리가 완료될 때의 타임스탬프입니다.

status-code

AS2 메시지 전송 프로세스의 상태에 해당하는 코드입니다. 유효한 값: COMPLETED | FAILED | PROCESSING.

failure-code

전송이 실패한 경우 전송이 실패한 이유의 범주입니다.

failure-message

전송 실패의 경우 전송 실패 이유에 대한 세부 정보입니다.

transfer-id

전송 이벤트의 고유 식별자입니다.

Example AS2 페이로드 수신 완료된 예제 이벤트

```
{
  "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": "amzn-s3-demo-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 AS2 MDN 수신 실패 예제 이벤트

```
{
  "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": "amzn-s3-demo-bucket1",
    "file-key": "/as2Integ/TestOutboundWrongCert.dat",
    "file-bucket": "amzn-s3-demo-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"
}
```


의 보안 AWS Transfer Family

의 클라우드 보안 AWS 이 최우선 순위입니다. AWS 고객은 보안에 가장 민감한 조직의 요구 사항을 충족하도록 구축된 데이터 센터 및 네트워크 아키텍처의 이점을 누릴 수 있습니다.

보안은 AWS 와 사용자 간의 공동 책임입니다. [공동 책임 모델](#)은 이를 클라우드의 보안과 클라우드 내 보안으로 설명합니다.

AWS 서비스 가 특정 규정 준수 프로그램의 범위 내에 있는지 알아보려면 [AWS 서비스 규정 준수 프로그램 제공 범위 내](#)를 참조하고 관심 있는 규정 준수 프로그램을 선택합니다. 일반 정보는 [AWS 규정 준수 프로그램](#).

를 사용하여 타사 감사 보고서를 다운로드할 수 있습니다 AWS Artifact. 자세한 내용은 [Downloading Reports inDownloading AWS Artifact](#)을 참조하세요.

사용 시 규정 준수 책임은 데이터의 민감도, 회사의 규정 준수 목표 및 관련 법률과 규정에 따라 AWS 서비스 결정됩니다. 사용 시 규정 준수 책임에 대한 자세한 내용은 [AWS 보안 설명서를](#) AWS 서비스참조하세요.

이 설명서는를 사용할 때 공동 책임 모델을 적용하는 방법을 이해하는 데 도움이 됩니다 AWS Transfer Family. 다음 주제에서는 보안 및 규정 준수 목표를 충족하도록 AWS Transfer Family 를 구성하는 방법을 보여줍니다. 또한 AWS Transfer Family 리소스를 모니터링하고 보호하는 데 도움이 되는 다른 AWS 서비스를 사용하는 방법을 알아봅니다.

기존 애플리케이션을 수정하거나 서버 인프라를 관리할 필요 AWS 없이에서 확장 가능하고 안전한 파일 전송 아키텍처를 구축하는 방법에 대한 규범적 지침과 실습을 제공하는 워크숍을 제공합니다. 이 워크숍의 세부 정보는 [여기](#)에서 확인할 수 있습니다.

주제

- [VPC 연결 보안 이점](#)
- [AWS Transfer Family 서버에 대한 보안 정책](#)
- [AWS Transfer Family SFTP 커넥터에 대한 보안 정책](#)
- [에서 하이브리드 포스트 양자 키 교환 사용 AWS Transfer Family](#)
- [데이터 보호 및 암호화](#)
- [Transfer Family에서 SSH 및 PGP 키 관리](#)
- [에 대한 자격 증명 및 액세스 관리 AWS Transfer Family](#)

- [예 대한 규정 준수 검증 AWS Transfer Family](#)
- [의 복원력 AWS Transfer Family](#)
- [VPC와 AWS Transfer Family APIs 간에 프라이빗 연결 생성](#)
- [의 인프라 보안 AWS Transfer Family](#)
- [웹 애플리케이션 방화벽 추가](#)
- [교차 서비스 혼동된 대리자 방지](#)
- [AWS AWS Transfer Family에 대한 관리형 정책](#)

VPC 연결 보안 이점

VPC 송신 유형의 SFTP 커넥터는 VPC 간 리소스 액세스를 통해 향상된 보안 이점을 제공합니다.

- 네트워크 격리: 모든 트래픽은 VPC 환경 내에 유지되므로 프라이빗 엔드포인트 연결을 위해 퍼블릭 인터넷에서 완전한 네트워크 격리가 가능합니다.
- 소스 IP 제어: 원격 SFTP 서버는 VPC CIDR 범위의 IP 주소만 표시하므로 연결에 사용되는 소스 IP 주소를 완전히 제어할 수 있습니다.
- 프라이빗 엔드포인트 액세스: 프라이빗 IP 주소를 사용하여 VPC의 SFTP 서버에 직접 연결하여 퍼블릭 인터넷에 노출되지 않도록 합니다.
- 하이브리드 연결: 추가 인터넷 노출 없이 설정된 VPN 또는 Direct Connect 연결을 통해 온프레미스 SFTP 서버에 안전하게 액세스합니다.
- VPC 보안 제어: 기존 VPC 보안 그룹, NACLs 및 라우팅 정책을 활용하여 SFTP 커넥터 트래픽을 제어하고 모니터링합니다.

VPC Lattice 보안 모델

SFTP 커넥터에 대한 VPC 연결은 서비스 네트워크와 함께 AWS VPC Lattice를 사용하여 안전한 멀티 테넌트 액세스를 제공합니다.

- 혼동된 대리자 방지: 인증 및 권한 부여 검사는 커넥터가 구성된 특정 리소스에만 액세스할 수 있도록 하여 무단 테넌트 간 액세스를 방지합니다.
- IPv6-only 서비스 네트워크: IPv6 주소 지정을 사용하여 잠재적 IP 주소 충돌을 방지하고 보안 격리를 강화합니다.
- 전달 액세스 세션(FAS): 임시 자격 증명 처리를 사용하면 장기 자격 증명 스토리지 또는 수동 리소스 공유가 필요하지 않습니다.

- 리소스 수준 액세스 제어: 각 커넥터는 특정 리소스 구성과 연결되어 개별 SFTP 서버에 대한 세분화된 액세스 제어를 보장합니다.

VPC 연결에 대한 보안 모범 사례

VPC 송신 유형 커넥터를 사용하는 경우 다음 보안 모범 사례를 따르세요.

- 보안 그룹: 필요한 리소스 간에만 SFTP 트래픽(포트 22)을 허용하도록 보안 그룹을 구성합니다. 소스 및 대상 IP 범위를 필요한 최소 범위로 제한합니다.
- Resource Gateway 배치: 가능하면 프라이빗 서브넷에 Resource Gateway를 배포하고 고가용성을 위해 두 개 이상의 가용 영역에 걸쳐 있는지 확인합니다.
- 네트워크 모니터링: VPC 흐름 로그 및 Amazon CloudWatch를 사용하여 네트워크 트래픽 패턴을 모니터링하고 비정상적인 활동을 감지합니다.
- 액세스 로깅: 커넥터 로깅을 활성화하여 파일 전송 활동을 추적하고 규정 준수 요구 사항에 대한 감사 추적을 유지합니다.
- 리소스 구성 관리: 리소스 구성을 정기적으로 검토하고 업데이트하여 올바른 SFTP 서버를 가리키고 적절한 네트워크 설정을 사용하는지 확인합니다.

AWS Transfer Family 서버에 대한 보안 정책

의 서버 보안 정책을 AWS Transfer Family 사용하면 서버와 연결된 암호화 알고리즘(메시지 인증 코드(MACs), 키 교환(KEXs), 암호 제품군, 콘텐츠 암호화 암호 및 해시 알고리즘) 세트를 제한할 수 있습니다.

AWS Transfer Family 는 하이브리드 키 교환 알고리즘을 사용하는 포스트 양자 보안 정책을 지원하여 기존 암호화 방법과 포스트 양자 알고리즘을 결합하여 향후 양자 컴퓨팅 위협에 대한 보안을 강화합니다. 자세한 내용은 [에서 하이브리드 포스트 양자 키 교환 사용 AWS Transfer Family](#) 단원을 참조하십시오.

지원되는 키 알고리즘 목록은 [암호화 알고리즘](#) 섹션을 참조하세요. 서버 호스트 키 및 서비스 관리 사용자 키와 함께 사용할 수 있는 지원되는 키 알고리즘 목록은 [Transfer Family에서 SSH 및 PGP 키 관리](#)를 참조하세요.

Note

2025년부터 모든 새로운 AWS Transfer Family 보안 정책에는 하이브리드 키 교환 알고리즘을 사용한 포스트 쿼텀 암호화 지원이 포함됩니다. 포스트 양자 보안에 대한 자세한 내용은 섹션을 참조하세요 [에서 하이브리드 포스트 양자 키 교환 사용 AWS Transfer Family](#).

Note

서버를 최신 보안 정책으로 업데이트하는 것이 좋습니다.

- TransferSecurityPolicy-2024-01는 콘솔, API 또는 CLI를 사용하여 서버를 생성할 때 서버에 연결된 기본 보안 정책입니다.
- CloudFormation을 사용하여 Transfer Family 서버를 생성하고 기본 보안 정책을 수락하면 서버에 할당됩니다 TransferSecurityPolicy-2018-11.

클라이언트 호환성이 우려되는 경우 기본 정책을 사용하는 대신 서버를 생성하거나 업데이트할 때 사용할 보안 정책을 명확하게 명시하십시오. 이 정책은 변경될 수 있습니다. 서버의 보안 정책을 변경하려면 섹션을 참조하세요 [보안 정책 편집](#).

Note

이전의 포스트 양자 정책(TransferSecurityPolicy-PQ-SSH-Experimental-2023-04 및 TransferSecurityPolicy-PQ-SSH-FIPS-Experimental-2023-04)은 더 이상 사용되지 않습니다. 대신 새 정책을 사용하는 것이 좋습니다.

Transfer Family의 보안에 대한 자세한 내용은 다음 블로그 게시물을 참조하세요.

- [AWS Transfer Family 서버 보안을 개선하기 위한 6가지 팁](#)
- [Transfer Family가 안전하고 규정을 준수하는 관리형 파일 전송 솔루션을 구축하는 데 도움이 되는 방법](#)

주제

- [암호화 알고리즘](#)

- [보안 정책 세부 정보](#)

암호화 알고리즘

호스트 키의 경우 다음 알고리즘을 지원합니다.

- rsa-sha2-256
- rsa-sha2-512
- ecdsa-sha2-nistp256
- ecdsa-sha2-nistp384
- ecdsa-sha2-nistp521
- ssh-ed25519

또한 다음 보안 정책을 허용합니다ssh-rsa.

- TransferSecurityPolicy-2018-11
- TransferSecurityPolicy-2020-06
- TransferSecurityPolicy-FIPS-2020-06
- TransferSecurityPolicy-FIPS-2023-05
- TransferSecurityPolicy-FIPS-2024-01

Note

항상 RSA 키 유형과 지원되는 알고리즘 중 하나일 수 있는 ssh-rsaRSA 호스트 키 알고리즘의 차이점을 이해하는 것이 중요합니다.

다음은 각 보안 정책에 지원되는 암호화 알고리즘 목록입니다.

Note

다음 표 및 정책에서 다음과 같은 알고리즘 유형 사용에 유의하세요.

- SFTP 서버는 SshCiphers, SshKexs 및 SshMacs 섹션에서만 알고리즘을 사용합니다.
- FTPS 서버는 TlsCiphers 섹션의 알고리즘만 사용합니다.

- FTP 서버는 암호화를 사용하지 않으므로 이러한 알고리즘을 사용하지 마십시오.
- AS2 서버는 ContentEncryptionCiphers 및 HashAlgorithms 섹션의 알고리즘만 사용합니다. 이 섹션에서는 파일 콘텐츠를 암호화하고 서명하는 데 사용되는 알고리즘을 정의합니다.
- FIPS-2024-05 및 FIPS-2024-01 보안 정책은 FIPS-2024-05가 ssh-rsa 알고리즘을 지원하지 않는다는 점을 제외하면 동일합니다.
- Transfer Family는 기존 정책과 매우 유사한 새로운 제한 정책을 도입했습니다.
 - TransferSecurityPolicy-Restricted-2018-11 및 TransferSecurityPolicy-2018-11 보안 정책은 제한적 정책이 chacha20-poly1305@openssh.com 암호를 지원하지 않는다는 점을 제외하면 동일합니다.
 - TransferSecurityPolicy-Restricted-2020-06 및 TransferSecurityPolicy-2020-06 보안 정책은 제한 정책이 chacha20-poly1305@openssh.com 암호를 지원하지 않는다는 점을 제외하면 동일합니다.

* 다음 표에서는 chacha20-poly1305@openssh.com 암호가 비제한 정책에만 포함됩니다.

보안 정책	TransferSecurityPolicy-2025-03	TransferSecurityPolicy-2025-08	TransferSecurityPolicy-2025-08	TransferSecurityPolicy-2025-07	TransferSecurityPolicy-2025-01	TransferSecurityPolicy-2024-01	TransferSecurityPolicy-2023-05	TransferSecurityPolicy-2023-05	TransferSecurityPolicy-2020-06	TransferSecurityPolicy-2020-06	TransferSecurityPolicy-2018-11
	FIPS	SshA	AS2R	Restricted	FIPS	FIPS	FIPS	FIPS	FIPS	FIPS	FIPS
SshCiphers											
aes128-ctr	◆	◆	◆	◆	◆				◆	◆	◆
aes128-gcm	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆

보안 정책	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS
	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo
	licy-2025-	licy-2025-	licy-2025-	licy-2025-	licy-2024-	licy-2024-	licy-2023-	licy-2023-	licy-2020-	licy-2020-	licy-2018-	
	-03	FIPS	SshA	AS2R	-01	FIPS	-05	FIPS	-03	-06	FIPS	-11
		-2025-08	ditCom	stricted		-2024-01/		-2023-05		및	-2020-06	및
			iant-2025-	2025-07		TransferS				TransferS		TransferS
			-02			ecurityPo				ecurityPo		ecurityPo
						licy-				licy-		licy-
						FIPS				Rest		Rest
						-2024-05				ricted-20		ricted-20
										20-06		18-11
m@openssh.com												
aes192-ctr	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆
aes256-ctr	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆
aes256-gc	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆
m@openssh.com												
chacha20-poly1305@openssh.com									◆*		◆*	
SshKexs												
mlkem768x25519-sha256	◆		◆									

보안 정책	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS
	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo
	licy-2025	licy-	licy-	licy-	licy-2024	licy-	licy-2023	licy-	licy-2022	licy-2020	licy-	licy-2018
	-03	FIPS	SshA	AS2R	-01	FIPS	-05	FIPS	-03	-06	FIPS	-11
		-2025-08	ditCom	restricted		-2024-01/		-2023-05		및	-2020-06	및
mlkem768n	◆			◆		TransferS				TransferS		TransferS
nistp256-						ecurityPo				ecurityPo		ecurityPo
s						licy-				licy-		licy-
ha256						FIPS				Rest		Rest
						-2024-05				riected-20		riected-20
										20-06		18-11
mlkem1024	◆			◆								
nistp384-												
sha384												
curve2551		◆	◆	◆		◆		◆				◆
9-												
sha256												
curve2551		◆	◆	◆		◆		◆				◆
9-												
sha256@												
libssh.or												
g												
diffie-												◆
he												
llman-												
gro												
up14-												
sha1												

보안 정책	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS
	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo
	licy-2025-	licy-2025-	licy-2025-	licy-2025-	licy-2024-	licy-2024-	licy-2023-	licy-2023-	licy-2022-	licy-2020-	licy-2020-	licy-2018-	
	-03	FIPS	SshA	AS2R	-01	FIPS	-05	FIPS	-03	-06	FIPS	-11	
		-2025-08	ditComes	stricted		-2024-01/		-2023-05		및	-2020-06	및	
			iant-2025	2025-07		TransferS				TransferS		TransferS	
			-02			ecurityPo				ecurityPo		ecurityPo	
						licy-				licy-		licy-	
						FIPS				Rest		Rest	
						-2024-05				ricted-20		ricted-20	
diffie-he llman- gro up14- sha2 56										◆	◆	◆	
diffie-he llman- gro up16- sha5 12	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	
diffie-he llman- gro up18- sha5 12	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	

보안 정책	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS
	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo
	licy-2025	licy-	licy-	licy-	licy-2024	licy-	licy-2023	licy-	licy-2022	licy-2020	licy-	licy-2018
	-03	FIPS	SshA	AS2R	-01	FIPS	-05	FIPS	-03	-06	FIPS	-11
		-2025-08	ditCom	restricted		-2024-01/		-2023-05		및	-2020-06	및
			iant-2025	2025-07		TransferS				TransferS		TransferS
			-02			ecurityPo				ecurityPo		ecurityPo
						licy-				licy-		licy-
						FIPS				Rest		Rest
						-2024-05				ricted-20		ricted-20
										20-06		18-11
diffie-	◆	◆	◆	◆	◆		◆	◆	◆	◆	◆	◆
he												
llman-												
gro												
up-												
exchan												
ge-												
sha256												
ecdh-	◆	◆		◆	◆	◆				◆	◆	◆
sha2												
-												
nistp256												
ecdh-	◆	◆		◆	◆	◆				◆	◆	◆
sha2												
-												
nistp384												
ecdh-	◆	◆		◆	◆	◆				◆	◆	◆
sha2												
-												
nistp521												
SshMacs												

보안 정책	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS
	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo
	licy-2025-	licy-2025-	licy-2025-	licy-2025-	licy-2024-	licy-2024-	licy-2023-	licy-2023-	licy-2022-	licy-2022-	licy-2021-	licy-2018-	
	-03	FIPS	SshA	AS2R	-01	FIPS	-05	FIPS	-03	-06	FIPS	-11	
		-2025-08	ditCom	restricted		-2024-01/		-2023-05		및	-2020-06	및	
			iant-2025-	2025-07		TransferS				TransferS		TransferS	
			-02			ecurityPo				ecurityPo		ecurityPo	
						licy-				licy-		licy-	
						FIPS				Rest		Rest	
						-2024-05				ricted-20		ricted-20	
hmac-													◆
sha1													
hmac-													◆
sha1													
-													
etm@open													
ssh.com													
hmac-								◆	◆	◆	◆		
sha2													
-256													
hmac-	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	
sha2													
-256-													
etm@													
openssh.c													
om													
hmac-								◆	◆	◆	◆		
sha2													
-512													

암호화 알고리즘 616

보안 정책	TransferS ecurity-2025-03	TransferS ecurity-2025-08	TransferS ecurity-2025-08	TransferS ecurity-2025-08	TransferS ecurity-2025-08	TransferS ecurity-2025-08	TransferS ecurity-2025-08	TransferS ecurity-2025-08	TransferS ecurity-2025-08	TransferS ecurity-2025-08	TransferS ecurity-2025-08	TransferS ecurity-2025-08
	licy-2025-03	licy-2025-08	licy-2025-08	licy-2025-08	licy-2025-08	licy-2025-08	licy-2025-08	licy-2025-08	licy-2025-08	licy-2025-08	licy-2025-08	licy-2025-08
	-03	FIPS	SshA	AS2R	-01	FIPS	-05	FIPS	-03	-06	FIPS	-11
		-2025-08	ditCom	stricted		-2024-01/		-2023-05		및	-2020-06	및
			iant-2025-02	2025-07		TransferS				TransferS		TransferS
						ecurityPo				ecurityPo		ecurityPo
						licy-				licy-		licy-
						FIPS				Rest		Rest
						-2024-05				ricted-20		ricted-20
										20-06		18-11
aes256-cbc	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆
aes192-cbc	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆
aes128-cbc	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆
3des-cbc	◆	◆	◆		◆	◆	◆	◆	◆	◆	◆	◆
HashAlgorithms												
sha256	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆
sha384	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆
sha512	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆
sha1	◆	◆	◆		◆	◆	◆	◆	◆	◆	◆	◆
TlsCiphers												

보안 정책	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS	TransferS
	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo	ecurityPo
	licy-2025-	licy-2025-	licy-2025-	licy-2025-	licy-2024-	licy-2024-	licy-2023-	licy-2023-	licy-2022-	licy-2020-	licy-2020-	licy-2018-
	-03	FIPS	SshA	AS2R	-01	FIPS	-05	FIPS	-03	-06	FIPS	-11
		-2025-08ditCom	restricted			-2024-01/		-2023-05		및	-2020-06	및
		iant-2025-07				TransferS				TransferS		TransferS
		-02				ecurityPo				ecurityPo		ecurityPo
						licy-				licy-		licy-
						FIPS				Rest		Rest
						-2024-05				ricted-20		ricted-20
										20-06		18-11
TLS_ECDHE	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆
_ECDSA_WI												
TH_AES_12												
8_CBC_SHA												
256												
TLS_ECDHE	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆
_ECDSA_WI												
TH_AES_12												
8_GCM_SHA												
256												
TLS_ECDHE	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆
_ECDSA_WI												
TH_AES_25												
6_CBC_SHA												
384												
TLS_ECDHE	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆
_ECDSA_WI												
TH_AES_25												
6_GCM_SHA												
384												

보안 정책	TransferSecurityPolicy-2025-03	TransferSecurityPolicy-2025-08	TransferSecurityPolicy-2025-08	TransferSecurityPolicy-2025-08	TransferSecurityPolicy-2025-07	TransferSecurityPolicy-2024-01	TransferSecurityPolicy-2024-05	TransferSecurityPolicy-2023-05	TransferSecurityPolicy-2023-03	TransferSecurityPolicy-2023-06	TransferSecurityPolicy-2020-06	TransferSecurityPolicy-2018-11
		FIPS	SshA	AS2R			FIPS		FIPS		FIPS	
		-2025-08	ditCom	stricted		-2024-01/		-2023-05		-2023-06		-2020-06
			iant-2025-02	2025-07		TransferSecurityPolicy-				TransferSecurityPolicy-		TransferSecurityPolicy-
						FIPS				Rest		Rest
						-2024-05				stricted-2020-06		stricted-2018-11
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆
TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆	◆

[illegible]

보안 정책 세부 정보

다음 섹션에는 각 보안 정책의 JSON 표현이 포함되어 있습니다.

TransferSecurityPolicy-2025-03

다음은 TransferSecurityPolicy-2025-03 보안 정책을 보여줍니다.

```
{
  "SecurityPolicy": {
    "Fips": false,
    "SecurityPolicyName": "TransferSecurityPolicy-2025-03",
    "SshCiphers": [
      "aes256-gcm@openssh.com",
      "aes128-gcm@openssh.com",
      "aes128-ctr",
      "aes256-ctr",

```



```
    "aes192-ctr"
  ],
  "SshKexs": [
    "mlkem768x25519-sha256",
    "mlkem768nistp256-sha256",
    "mlkem1024nistp384-sha384",
    "ecdh-sha2-nistp256",
    "ecdh-sha2-nistp384",
    "ecdh-sha2-nistp521",
    "curve25519-sha256",
    "curve25519-sha256@libssh.org",
    "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"
  ],
  "ContentEncryptionCiphers": [
    "aes256-cbc",
    "aes192-cbc",
    "aes128-cbc",
    "3des-cbc"
  ],
  "HashAlgorithms": [
    "sha256",
    "sha384",
    "sha512",
    "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"
  ],
  "Type": "SERVER",
  "Protocols": [
    "SFTP",
```

```

    "FTPS"
  ]
}
}

```

TransferSecurityPolicy-FIPS-2025-03

다음은 TransferSecurityPolicy-FIPS-2025-03 보안 정책을 보여줍니다.

```

{
  "SecurityPolicy": {
    "Fips": true,
    "SecurityPolicyName": "TransferSecurityPolicy-FIPS-2025-03",
    "SshCiphers": [
      "aes256-gcm@openssh.com",
      "aes128-gcm@openssh.com",
      "aes256-ctr",
      "aes192-ctr",
      "aes128-ctr"
    ],
    "SshKexs": [
      "mlkem768x25519-sha256",
      "mlkem768nistp256-sha256",
      "mlkem1024nistp384-sha384",
      "ecdh-sha2-nistp256",
      "ecdh-sha2-nistp384",
      "ecdh-sha2-nistp521",
      "diffie-hellman-group-exchange-sha256",
      "diffie-hellman-group16-sha512",
      "diffie-hellman-group18-sha512"
    ],
    "SshMacs": [
      "hmac-sha2-512-etm@openssh.com",
      "hmac-sha2-256-etm@openssh.com"
    ],
    "ContentEncryptionCiphers": [
      "aes256-cbc",
      "aes192-cbc",
      "aes128-cbc",
      "3des-cbc"
    ],
    "HashAlgorithms": [
      "sha256",

```

```

        "sha384",
        "sha512",
        "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"
    ],
    "Type": "SERVER",
    "Protocols": [
        "SFTP",
        "FTPS"
    ]
}
}

```

TransferSecurityPolicy-AS2Restricted-2025-07

이 보안 정책은 레거시 암호화 알고리즘을 제외하여 보안을 강화해야 하는 AS2 파일 전송을 위해 설계되었습니다. 3DES 및 SHA-1과 같은 약한 알고리즘에 대한 지원을 제거하면서 최신 AES 암호화 및 SHA-2 해시 알고리즘을 지원합니다. SHA-1

Note

이 보안 정책은 3DES(ContentEncryptionCiphers)를 지원하지 않고 SHA1(HashAlgorithms)을 지원하지 않는다는 점을 제외하면 TransferSecurityPolicy-2025-03과 동일합니다. SHA1 HashAlgorithms 여기에는 포스트 양자 암호화 알고리즘(mlkem* KEXs)을 포함하여 2025년 3월의 모든 알고리즘이 포함됩니다.

```

{
  "SecurityPolicy": {
    "Fips": false,
    "SecurityPolicyName": "TransferSecurityPolicy-AS2Restricted-2025-07",
    "SshCiphers": [

```

```

    "aes256-gcm@openssh.com",
    "aes128-gcm@openssh.com",
    "aes128-ctr",
    "aes256-ctr",
    "aes192-ctr"
  ],
  "SshKexs": [
    "mlkem768x25519-sha256",
    "mlkem768nistp256-sha256",
    "mlkem1024nistp384-sha384",
    "ecdh-sha2-nistp256",
    "ecdh-sha2-nistp384",
    "ecdh-sha2-nistp521",
    "curve25519-sha256",
    "curve25519-sha256@libssh.org",
    "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"
  ],
  "ContentEncryptionCiphers": [
    "aes256-cbc",
    "aes192-cbc",
    "aes128-cbc"
  ],
  "HashAlgorithms": [
    "sha256",
    "sha384",
    "sha512"
  ],
  "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"
  ],
  "Type": "SERVER",

```

```

        "Protocols": [
            "SFTP",
            "FTPS"
        ]
    }
}

```

TransferSecurityPolicy-SshAuditCompliant-2025-02

다음은 TransferSecurityPolicy-SshAuditCompliant-2025-02 보안 정책을 보여줍니다.

Note

이 보안 정책은 ssh-audit 도구에서 제공하는 권장 사항을 중심으로 설계되었으며 해당 도구를 100% 준수합니다.

```

{
  "SecurityPolicy": {
    "Fips": false,
    "Protocols": [
      "SFTP",
      "FTPS"
    ],
    "SecurityPolicyName": "TransferSecurityPolicy-SshAuditCompliant-2025-02",
    "SshCiphers": [
      "aes128-gcm@openssh.com",
      "aes256-gcm@openssh.com",
      "aes128-ctr",
      "aes256-ctr",
      "aes192-ctr"
    ],
    "SshKexs": [
      "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"
    ]
  }
}

```

```

    ],
    "ContentEncryptionCiphers": [
        "aes256-cbc",
        "aes192-cbc",
        "aes128-cbc",
        "3des-cbc"
    ],
    "HashAlgorithms": [
        "sha256",
        "sha384",
        "sha512",
        "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"
    ],
    "Type": "SERVER"
}
}

```

TransferSecurityPolicy-2024-01

다음은 TransferSecurityPolicy-2024-01 보안 정책을 보여줍니다.

```

{
  "SecurityPolicy": {
    "Fips": false,
    "SecurityPolicyName": "TransferSecurityPolicy-2024-01",
    "SshCiphers": [
        "aes128-gcm@openssh.com",
        "aes256-gcm@openssh.com",
        "aes128-ctr",
        "aes256-ctr",
        "aes192-ctr"
    ],
    "SshKexs": [

```

```

        "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"
    ],
    "ContentEncryptionCiphers": [
        "aes256-cbc",
        "aes192-cbc",
        "aes128-cbc",
        "3des-cbc"
    ],
    "HashAlgorithms": [
        "sha256",
        "sha384",
        "sha512",
        "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"
    ]
}
}

```

TransferSecurityPolicy-FIPS-2024-01/TransferSecurityPolicy-FIPS-2024-05

다음은 TransferSecurityPolicy-FIPS-2024-01 및 TransferSecurityPolicy-FIPS-2024-05 보안 정책을 보여줍니다.

Note

FIPS 서비스 엔드포인트 및 TransferSecurityPolicy-FIPS-2024-01 및 TransferSecurityPolicy-FIPS-2024-05 보안 정책은 일부 AWS 리전에서만 사용할 수 있습니다. 자세한 내용은 AWS 일반 참조에서 [AWS Transfer Family 엔드포인트 및 할당량](#)을 참조하세요.

이 두 보안 정책의 유일한 차이점은 TransferSecurityPolicy-FIPS-2024-01이 ssh-rsa 알고리즘을 지원하고 TransferSecurityPolicy-FIPS-2024-05는 지원하지 않는다는 것입니다.

```
{
  "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-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"
    ],
    "ContentEncryptionCiphers": [
      "aes256-cbc",
      "aes192-cbc",
      "aes128-cbc",
      "3des-cbc"
    ],
    "HashAlgorithms": [
      "sha256",
      "sha384",
      "sha512",
```



```

        "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"
    ]
}

```

TransferSecurityPolicy-2023-05

다음은 TransferSecurityPolicy-2023-05 보안 정책을 보여줍니다.

```

{
  "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"
    ],
    "ContentEncryptionCiphers": [
      "aes256-cbc",
      "aes192-cbc",

```

```

        "aes128-cbc",
        "3des-cbc"
    ],
    "HashAlgorithms": [
        "sha256",
        "sha384",
        "sha512",
        "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"
    ]
}
}

```

TransferSecurityPolicy-FIPS-2023-05

에 대한 FIPS 인증 세부 정보는에서 확인할 AWS Transfer Family 수 있습니다. <https://csrc.nist.gov/projects/cryptographic-module-validation-program/validated-modules/search/all>

다음은 TransferSecurityPolicy-FIPS-2023-05 보안 정책을 보여줍니다.

Note

FIPS 서비스 엔드포인트 및 TransferSecurityPolicy-FIPS-2023-05 보안 정책은 일부 AWS 리전에서만 사용할 수 있습니다. 자세한 내용은 AWS 일반 참조에서 [AWS Transfer Family 엔드포인트 및 할당량](#)을 참조하세요.

```

{
  "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"
    ],
    "ContentEncryptionCiphers": [
        "aes256-cbc",
        "aes192-cbc",
        "aes128-cbc",
        "3des-cbc"
    ],
    "HashAlgorithms": [
        "sha256",
        "sha384",
        "sha512",
        "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"
    ]
}

```

TransferSecurityPolicy-2022-03

다음은 TransferSecurityPolicy-2022-03 보안 정책을 보여줍니다.

```
{
```

```

"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"
  ],
  "ContentEncryptionCiphers": [
    "aes256-cbc",
    "aes192-cbc",
    "aes128-cbc",
    "3des-cbc"
  ],
  "HashAlgorithms": [
    "sha256",
    "sha384",
    "sha512",
    "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"
  ]
}

```

```
}
}
```

TransferSecurityPolicy-2020-06 및 TransferSecurityPolicy-Restricted-2020-06

다음은 TransferSecurityPolicy-2020-06 보안 정책을 보여줍니다.

Note

TransferSecurityPolicy-Restricted-2020-06 및 TransferSecurityPolicy-2020-06 보안 정책은 제한 정책이 chacha20-poly1305@openssh.com 암호를 지원하지 않는다는 점을 제외하면 동일합니다.

```
{
  "SecurityPolicy": {
    "Fips": false,
    "SecurityPolicyName": "TransferSecurityPolicy-2020-06",
    "SshCiphers": [
      "chacha20-poly1305@openssh.com", //Not included in TransferSecurityPolicy-
Restricted-2020-06
      "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"
  ],
  "ContentEncryptionCiphers": [
    "aes256-cbc",
    "aes192-cbc",
    "aes128-cbc",
    "3des-cbc"
  ],
  "HashAlgorithms": [
    "sha256",
    "sha384",
    "sha512",
    "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"
  ]
}
}

```

TransferSecurityPolicy-FIPS-2020-06

에 대한 FIPS 인증 세부 정보는에서 확인할 AWS Transfer Family 수 있습니다. <https://csrc.nist.gov/projects/cryptographic-module-validation-program/validated-modules/search/all>

다음은 TransferSecurityPolicy-FIPS-2020-06 보안 정책을 보여줍니다.

Note

FIPS 서비스 엔드포인트 및 TransferSecurityPolicy-FIPS-2020-06 보안 정책은 일부 AWS 리전에서만 사용할 수 있습니다. 자세한 내용은AWS 일반 참조에서 [AWS Transfer Family 엔드포인트 및 할당량](#)을 참조하세요.

```
{
```

```

"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"
  ],
  "ContentEncryptionCiphers": [
    "aes256-cbc",
    "aes192-cbc",
    "aes128-cbc",
    "3des-cbc"
  ],
  "HashAlgorithms": [
    "sha256",
    "sha384",
    "sha512",
    "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"
  ]
}
}

```

TransferSecurityPolicy-2018-11 및 TransferSecurityPolicy-Restricted-2018-11

다음은 TransferSecurityPolicy-2018-11 보안 정책을 보여줍니다.

Note

TransferSecurityPolicy-Restricted-2018-11 및 TransferSecurityPolicy-2018-11 보안 정책은 제한적 정책이 chacha20-poly1305@openssh.com 암호를 지원하지 않는다는 점을 제외하면 동일합니다.

```

{
  "SecurityPolicy": {
    "Fips": false,
    "SecurityPolicyName": "TransferSecurityPolicy-2018-11",
    "SshCiphers": [
      "chacha20-poly1305@openssh.com", //Not included in TransferSecurityPolicy-
Restricted-2018-11
      "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"
    ],
    "ContentEncryptionCiphers": [
      "aes256-cbc",
      "aes192-cbc",
      "aes128-cbc",
      "3des-cbc"
    ],
    "HashAlgorithms": [
      "sha256",
      "sha384",
      "sha512",
      "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"
    ]
  }
}

```

AWS Transfer Family SFTP 커넥터에 대한 보안 정책

의 SFTP 커넥터 보안 정책을 AWS Transfer Family 사용하면 SFTP 커넥터와 연결된 암호화 알고리즘(메시지 인증 코드(MACs), 키 교환(KEXs) 및 암호 제품군) 세트를 제한할 수 있습니다. 다음은 각 SFTP 커넥터 보안 정책에 대해 지원되는 암호화 알고리즘 목록입니다.

Note

TransferSFTPConnectorSecurityPolicy-2024-03는 SFTP 커넥터에 적용되는 기본 보안 정책입니다.

커넥터의 보안 정책을 변경할 수 있습니다. Transfer Family 왼쪽 탐색 창에서 커넥터를 선택하고 커넥터를 선택합니다. 그런 다음 Sftp 구성 섹션에서 편집을 선택합니다. 암호화 알고리즘 옵션 섹션의 보안 정책 필드의 드롭다운 목록에서 사용 가능한 보안 정책을 선택합니다.

암호화 알고리즘

호스트 키의 경우 SFTP 커넥터는 ed25519를 제외하고 Transfer Family 서버에 지원되는 모든 알고리즘을 지원합니다.

- rsa-sha2-256
- rsa-sha2-512
- ecdsa-sha2-nistp256
- ecdsa-sha2-nistp384
- ecdsa-sha2-nistp521

또한 호스트 키의 경우를 지원ssh-rsa하지만에 대해서만 지원됩니다TransferSFTPConnectorSecurityPolicy-2023-07.

인증을 위해 SFTP 커넥터는 다음 키 유형을 지원합니다.

- ssh-rsa
- ecdsa

SFTP 커넥터 보안 정책 세부 정보

다음 표에는 각 SFTP 커넥터 보안 정책에서 지원하는 특정 암호화 알고리즘이 나와 있습니다.

보안 정책	TransferSFTPConnectorSecurityPolicy-FIPS-2024-10	TransferSFTPConnectorSecurityPolicy-2024-03	TransferSFTPConnectorSecurityPolicy-2023-07
Ciphers			
aes128-ctr			◆
aes128-gcm@openssh.com	◆	◆	◆
aes192-ctr		◆	◆
aes256-ctr		◆	◆
aes256-gcm@openssh.com	◆	◆	◆
Kexs			
curve25519-sha256		◆	◆
curve25519-sha256@libssh.org		◆	◆
diffie-hellman-group14-sha1			◆
diffie-hellman-group16-sha512		◆	◆
diffie-hellman-group18-sha512		◆	◆
diffie-hellman-group-exchange-sha256		◆	◆

보안 정책	TransferSFTPConnectorSecurityPolicy-FIPS-2024-10	TransferSFTPConnectorSecurityPolicy-2024-03	TransferSFTPConnectorSecurityPolicy-2023-07
ecdh-sha2-nistp256	◆		
ecdh-sha2-nistp384	◆		
ecdh-sha2-nistp521	◆		
Macs			
hmac-sha2-512-etm@openssh.com		◆	◆
hmac-sha2-256-etm@openssh.com		◆	◆
hmac-sha2-512	◆	◆	◆
hmac-sha2-256	◆	◆	◆
hmac-sha1			◆
hmac-sha1-96			◆
Host Key Algorithms			
rsa-sha2-256	◆	◆	◆
rsa-sha2-512	◆	◆	◆
ecdsa-sha2-nistp256	◆	◆	◆
ecdsa-sha2-nistp384		◆	◆
ecdsa-sha2-nistp521		◆	◆
ssh-rsa			◆

에서 하이브리드 포스트 양자 키 교환 사용 AWS Transfer Family

Transfer Family는 Secure Shell(SSH) 프로토콜에 대한 하이브리드 포스트 양자 키 설정 옵션을 지원합니다. 네트워크 트래픽을 기록하고 향후 복호화를 위해 양자 컴퓨터에 저장하는 것이 이미 가능하기 때문에 포스트 쿼텀 키 설정이 필요합니다. 이를 지금 저장 나중 수확 공격이라고 합니다.

이 옵션은 Transfer Family에 연결하여 Amazon 심플 스토리지 서비스(Amazon S3) 스토리지 또는 Amazon Elastic File System(Amazon EFS)에서 안전하게 파일을 주고 받을 때 사용할 수 있습니다. SSH의 포스트 쿼텀 하이브리드 키 설정은 포스트 쿼텀 키 설정 메커니즘을 도입하여 기존 키 교환 알고리즘과 함께 사용합니다. 기존 암호군으로 생성된 SSH 키는 현재 기술로는 무차별 암호 대입 공격으로부터 안전합니다. 그러나 향후 대규모 양자 컴퓨팅이 등장하더라도 기존 암호화는 보안을 유지하지 못할 것으로 예상됩니다.

Transfer Family 연결을 통해 전달되는 데이터의 장기 기밀 유지에 의존하는 조직의 경우, 대규모 양자 컴퓨터를 사용할 수 있게 되기 전에 포스트 쿼텀 암호화로 마이그레이션하는 계획을 고려해야 합니다.

현재 암호화된 데이터를 잠재적 향후 공격으로부터 보호하기 위해 AWS 는 양자 방지 또는 양자 사후 알고리즘 개발에 암호화 커뮤니티에 참여하고 있습니다. Transfer Family에서 클래식 및 포스트 쿼텀 요소를 결합하는 하이브리드 포스트 쿼텀 키 교환 암호 제품군을 구현했습니다.

이러한 하이브리드 암호 제품군은 대부분의 AWS 지역에서 프로덕션 워크로드에 사용할 수 있습니다. 그러나 하이브리드 암호 제품군의 성능 특성 및 대역폭 요건은 클래식 키 교환 메커니즘의 해당 요건과 다르기 때문에 Transfer Family 연결에 대해 이 제품군을 테스트하는 것이 좋습니다.

[포스트 쿼텀 암호화](#) 보안 블로그 게시물에서 포스트 쿼텀 암호화에 대해 자세히 알아보세요.

목차

- [SSH의 포스트 쿼텀 하이브리드 키 교환 소개](#)
- [Transfer Family에서 포스트 쿼텀 하이브리드 키 설정이 작동하는 방식](#)
 - [ML-KEM을 사용하는 이유](#)
 - [포스트 쿼텀 하이브리드 SSH 키 교환 및 암호화 요건 \(FIPS 140\)](#)
- [Transfer Family의 포스트 쿼텀 하이브리드 키 교환 테스트](#)
 - [SFTP 엔드포인트에서 포스트 쿼텀 하이브리드 키 교환을 활성화하세요.](#)
 - [포스트 쿼텀 하이브리드 키 교환을 지원하는 SFTP 클라이언트 설정](#)
 - [SFTP에서 포스트 쿼텀 하이브리드 키 교환 확인](#)

SSH의 포스트 퀀텀 하이브리드 키 교환 소개

Transfer Family는 클래식 [Elliptic Curve Diffie-Hellman\(ECDH\)](#) 키 교환 알고리즘과 ML-KEM을 모두 사용하는 포스트 양자 하이브리드 키 교환 암호 제품군을 지원합니다. ML-KEM은 [미국 국립표준기술연구소\(NIST\)](#)가 첫 번째 표준 포스트 양자 키 계약 알고리즘으로 지정한 포스트 양자 퍼블릭 키 암호화 및 키 설정 알고리즘입니다.

클라이언트와 서버는 여전히 ECDH 키 교환을 수행합니다. 또한 서버는 포스트 퀀텀 공유 비밀을 클라이언트의 포스트 퀀텀 KEM 퍼블릭 키로 캡슐화하는데, 이는 클라이언트의 SSH 키 교환 메시지에 알려집니다. 이 전략은 기존 키 교환의 높은 신뢰성과 제안된 포스트 퀀텀 키 교환의 보안을 결합하여 ECDH 또는 포스트 퀀텀 공유 비밀이 침해되지 않는 한 핸드셰이크를 보호할 수 있도록 합니다.

Transfer Family에서 포스트 퀀텀 하이브리드 키 설정이 작동하는 방식

AWS 는 최근의 SFTP 파일 전송에서 포스트 양자 키 교환에 대한 지원을 발표했습니다 AWS Transfer Family. Transfer Family는 SFTP 및 기타 프로토콜을 사용하여 AWS 스토리지 서비스로 business-to-business 파일 전송을 안전하게 확장합니다. SFTP는 SSH를 통해 실행되는 FTP (파일 전송 프로토콜) 의 보다 안전한 버전입니다. Transfer Family의 포스트 퀀텀 키 교환 지원은 SFTP를 통한 데이터 전송의 보안 기준을 높입니다.

Transfer Family의 포스트 양자 하이브리드 키 교환 SFTP 지원에는 P25ML-KEM-768Curve25519P384의 결합이 포함됩니다. ML-KEM-1024 P256 [포스트 퀀텀 하이브리드 SSH 키 교환 초안](#)에는 다음과 같은 해당 SSH 키 교환 방법이 명시되어 있습니다.

- mlkem768nistp256-sha256
- mlkem1024nistp384-sha384
- mlkem768x25519-sha256

ML-KEM을 사용하는 이유

AWS 는 상호 운용 가능한 표준화된 알고리즘을 지원하기 위해 최선을 다하고 있습니다. ML-KEM은 [NIST 포스트 양자 암호화 프로젝트에서 표준화하고 승인한 유일한 포스트 양자](#) 키 교환 알고리즘입니다. 표준 본문은 이미 ML-KEM을 프로토콜에 통합하고 있습니다. 일부 AWS API 엔드포인트에서 TLS의 ML-KEM을 AWS 이미 지원합니다.

이 약정의 일환으로 AWS 는 ML-KEM을 SSH용 P256과 같은 NIST 승인 곡선과 결합하는 포스트 양자 암호화에 대한 초안 제안을 IETF에 제출했습니다. 고객의 보안을 강화하기 위해 SFTP 및 SSH에서 포

스트 양자 키 교환을 AWS 구현하는 것은 해당 초안을 따릅니다. 우리의 제안이 IETF에서 채택되어 표준이 될 때까지 향후 업데이트를 지원할 계획입니다.

초안이 표준화로 발전함에 따라 새로운 키 교환 방법(섹션에 나열됨 [Transfer Family에서 포스트 퀀텀 하이브리드 키 설정이 작동하는 방식](#))이 변경될 수 있습니다.

Note

포스트 양자 알고리즘 지원은 현재 TLS의 포스트 양자 하이브리드 키 교환 AWS KMS (에서 [하이브리드 포스트 양자 TLS 사용 참조 AWS KMS](#)) AWS Certificate Manager 및 AWS Secrets Manager API 엔드포인트에 사용할 수 있습니다.

포스트 퀀텀 하이브리드 SSH 키 교환 및 암호화 요건 (FIPS 140)

FIPS 규정 준수가 필요한 고객을 위해 Transfer Family는 FIPS 140 인증 오픈 소스 암호화 라이브러리 - AWS LC를 사용하여 SSH에서 AWS FIPS 승인 암호화를 제공합니다. Transfer Family의 TransferSecurityPolicy-FIPS-2025-03에서 지원되는 포스트 양자 하이브리드 키 교환 방법은 [NIST의 SP 800-56Cr2\(섹션 2\)](#)에 따라 FIPS 승인을 받았습니다. 독일 연방 정보보안청([BSI](#))과 프랑스 국립정보시스템청([ANSSI](#))에서도 이러한 포스트 퀀텀 하이브리드 키 교환 방법을 권장합니다.

Transfer Family의 포스트 퀀텀 하이브리드 키 교환 테스트

이 섹션에서는 포스트 퀀텀 하이브리드 키 교환을 테스트하기 위해 수행하는 단계를 설명합니다.

1. [SFTP 엔드포인트에서 포스트 퀀텀 하이브리드 키 교환을 활성화하세요..](#)
2. 앞서 언급한 초안 사양의 지침을 따라 포스트 퀀텀 하이브리드 키 교환을 지원하는 SFTP 클라이언트(예: [포스트 퀀텀 하이브리드 키 교환을 지원하는 SFTP 클라이언트 설정](#))를 사용하세요.
3. Transfer Family 서버를 사용하여 파일을 전송합니다.
4. [SFTP에서 포스트 퀀텀 하이브리드 키 교환 확인.](#)

SFTP 엔드포인트에서 포스트 퀀텀 하이브리드 키 교환을 활성화하세요.

Transfer Family에서 새 SFTP 서버 엔드포인트를 생성할 때 또는 기존 SFTP 엔드포인트에서 암호화 알고리즘 옵션을 편집하여 SSH 정책을 선택할 수 있습니다. 다음 스냅샷은 SSH 정책을 업데이트 AWS Management Console 하는의 예를 보여줍니다.

Cryptographic algorithm options Info

Security Policy

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

TransferSecurityPolicy-PQ-SSH-Experimental-2023-04 ▲

Find a security policy

TransferSecurityPolicy-2018-11
TransferSecurityPolicy-2020-06
TransferSecurityPolicy-2022-03
TransferSecurityPolicy-2023-05
TransferSecurityPolicy-2024-01
TransferSecurityPolicy-2025-03 ✓
TransferSecurityPolicy-Restricted-2018-11
TransferSecurityPolicy-Restricted-2020-06

to it. Additi

포스트 양자 키 교환을 지원하는 SSH 정책 이름은 TransferSecurityPolicy-2025-03 및 TransferSecurityPolicy-FIPS-2025-03입니다. Transfer Family 정책에 대한 자세한 설명은 [AWS Transfer Family 서버에 대한 보안 정책](#)를 참조하세요.

포스트 쿼텀 하이브리드 키 교환을 지원하는 SFTP 클라이언트 설정

SFTP Transfer Family 엔드포인트에서 올바른 포스트 쿼텀 SSH 정책을 선택한 후 Transfer Family에서 포스트 쿼텀 SFTP를 실험해 볼 수 있습니다. 테스트할 로컬 시스템에 최신 OpenSSH 클라이언트 (예: 버전 9.9)를 설치합니다.

Note

클라이언트가 이전에 나열된 하나 이상의 ML-KEM 알고리즘을 지원하는지 확인합니다. 다음 명령을 실행하여 OpenSSH 버전에 지원되는 알고리즘을 볼 수 있습니다 `ssh -Q kex`.

다음 명령과 같이 포스트 양자 하이브리드 키 교환 방법을 사용하여 예제 SFTP 클라이언트를 실행하여 SFTP 엔드포인트(예: `s-1111aaaa2222bbbb3.server.transfer.us-west-2.amazonaws.com`)에 연결할 수 있습니다.


```
sftp -v -o \
  KexAlgorithms=mlkem768x25519-sha256 \
  -i username_private_key_PEM_file \
  username@server-id.server.transfer.region-id.amazonaws.com
```

이전 명령에서 다음 항목을 자신의 정보로 바꿉니다.

- *username_private_key_PEM_file*을 SFTP 사용자의 프라이빗 키 PEM 인코딩 파일로 대체합니다.
- *### ##*을 인스턴스의 사용자 이름으로 바꿉니다.
- *## ID*를 Transfer Family 서버 ID로 교체
- *## ID*를 Transfer Family 서버가 위치한 실제 지역으로 바꾸세요.

SFTP에서 포스트 쿼텀 하이브리드 키 교환 확인

SFTP를 Transfer Family로 연결하는 동안 포스트 쿼텀 하이브리드 키 교환이 사용되었는지 확인하려면 클라이언트 출력을 확인하세요. 선택적으로 패킷 캡처 프로그램을 사용할 수 있습니다. OpenSSH 9.9 클라이언트를 사용하는 경우 출력은 다음과 비슷해야 합니다(간결성을 위해 관련 없는 정보 생략).

```
% sftp -o KexAlgorithms=mlkem768x25519-sha256 -v -o IdentitiesOnly=yes -
i username_private_key_PEM_file username@s-1111aaaa2222bbbb3.server.transfer.us-
west-2.amazonaws.com
OpenSSH_9.9p2, OpenSSL 3.4.1 11 Feb 2025
debug1: Reading configuration data /Users/username/.ssh/config
debug1: /Users/username/.ssh/config line 146: Applying options for *
debug1: Reading configuration data /Users/username/.ssh/bastions-config
debug1: Reading configuration data /opt/homebrew/etc/ssh/ssh_config
debug1: Connecting to s-1111aaaa2222bbbb3.server.transfer.us-west-2.amazonaws.com
[xxx.yyy.zzz.nnn] port 22.
debug1: Connection established.
[...]
debug1: Local version string SSH-2.0-OpenSSH_9.9
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 /Users/username/.ssh/known_hosts2: No such file or
directory
[...]
debug1: SSH2_MSG_KEXINIT sent
```

```

debug1: SSH2_MSG_KEXINIT received
debug1: kex: algorithm: mlkem768x25519-sha256
debug1: kex: host key algorithm: ssh-ed25519
debug1: kex: server->client cipher: aes128-ctr MAC: hmac-sha2-256-etm@openssh.com
compression: none
debug1: kex: client->server cipher: aes128-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:Ic1Ti0cdDmFdStj06rfU0cmmNccwAha/ASH2unr6zX0
[...]
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 s-1111aaaa2222bbbb3.server.transfer.us-west-2.amazonaws.com
([xxx.yyy.zzz.nnn]:22) using "publickey".
debug1: channel 0: new session [client-session] (inactive timeout: 0)
[...]
Connected to s-1111aaaa2222bbbb3.server.transfer.us-west-2.amazonaws.com.
sftp>

```

출력은 포스트 쿼텀 하이브리드 mlkem768x25519-sha256 방법을 사용하여 클라이언트 협상이 이루어졌으며 SFTP 세션이 성공적으로 설정되었음을 보여줍니다.

데이터 보호 및 암호화

AWS [공동 책임 모델](#) AWS Transfer Family (전송 패밀리)의 데이터 보호에 적용됩니다. 이 모델에 설명된 대로 AWS 는 모든 AWS 클라우드를 실행하는 글로벌 인프라를 보호할 책임이 있습니다. 사용자는 이 인프라에 호스팅되는 콘텐츠에 대한 통제 권한을 유지할 책임이 있습니다. 이 콘텐츠에는 사용되는 AWS 서비스에 대한 보안 구성 및 관리 작업이 포함됩니다. 데이터 프라이버시에 대한 자세한 설명은 [데이터 프라이버시 FAQ](#)를 참조하세요. 유럽의 데이터 보호에 대한 자세한 설명은 AWS 보안 블로그의 [AWS 공동 책임 모델 및 GDPR](#) 블로그 게시물을 참조하세요.

데이터 보호를 위해 AWS 계정 자격 증명을 보호하고를 사용하여 개별 사용자 계정을 설정하는 것이 좋습니다 AWS IAM Identity Center. 이러한 방식에서는 각 사용자에게 자신의 직무를 충실히 이행하는데 필요한 권한만 부여됩니다. 또한 다음과 같은 방법으로 데이터를 보호하는 것이 좋습니다.

- 각 계정에 다중 인증(MFA)을 사용합니다.

- SSL/TLS를 사용하여 AWS 리소스와 통신합니다. TLS 1.2를 지원합니다.
- 를 사용하여 API 및 사용자 활동 로깅을 설정합니다 AWS CloudTrail.
- AWS 서비스 내의 모든 기본 보안 제어와 함께 AWS 암호화 솔루션을 사용합니다.
- Amazon S3에 저장된 개인 데이터를 검색하고 보호하는 데 도움이 되는 Amazon Macie와 같은 고급 관리형 보안 서비스를 사용합니다.
- 명령줄 인터페이스 또는 API를 통해 AWS 에 액세스할 때 FIPS 140-2 검증된 암호화 모듈이 필요한 경우 FIPS 엔드포인트를 사용합니다. 사용 가능한 FIPS 엔드포인트에 대한 자세한 설명은 [Federal Information Processing Standard\(FIPS\) 140-2](#)를 참조하세요.

명칭 필드와 같은 자유 형식 필드에 고객 계정 번호와 같은 중요 식별 정보를 절대 입력하지 마세요. 여기에는 콘솔 AWS CLI, API 또는 AWS SDKs를 사용하여 Transfer Family 또는 기타 AWS 서비스를 사용하는 경우가 포함됩니다. Transfer Family 서비스 구성에 입력하는 모든 구성 데이터 또는 다른 서비스의 구성이 진단 로그에 포함되도록 선택될 수 있습니다. 외부 서버에 URL을 제공할 때 해당 서버에 대한 요청을 검증하기 위해 보안 인증 정보를 URL에 포함시키지 마세요.

반면, Transfer Family 서버를 오가는 업로드 및 다운로드 작업의 데이터는 완전히 비공개로 취급되며 SFTP 또는 FTPS 연결과 같은 암호화된 채널 외부에 존재하지 않습니다. 이 데이터는 승인된 사람만 액세스할 수 있습니다.

Transfer Family의 데이터 암호화

AWS Transfer Family 는 Amazon S3 버킷에 대해 설정한 기본 암호화 옵션을 사용하여 데이터를 암호화합니다. 버킷에서 암호화를 활성화한 경우 모든 객체는 버킷에 저장될 때 암호화됩니다. 객체는 Amazon S3 관리형 키(SSE-S3) 또는 AWS Key Management Service () 관리형 키(SSE-KMS AWS KMS)를 사용한 서버 측 암호화를 사용하여 암호화됩니다. 자세한 설명은 Amazon Simple Storage Service 사용자 가이드의 [서버 측 암호화를 사용하여 데이터 보호](#)를 참조하세요.

다음 단계에서는 데이터를 암호화하는 방법을 보여줍니다 AWS Transfer Family.

에서 암호화를 허용하려면 AWS Transfer Family

1. Amazon S3 버킷에 대한 기본 암호화를 활성화합니다. 자세한 설명은 Amazon Simple Storage Service 사용자 가이드의 [Amazon S3 버킷에 대한 Amazon S3 기본 암호화](#)를 참조하세요.
2. 사용자에게 연결된 AWS Identity and Access Management (IAM) 역할 정책을 업데이트하여 필요한 AWS Key Management Service (AWS KMS) 권한을 부여합니다.
3. 사용자에게 대한 세션 정책을 사용하는 경우 세션 정책은 필요한 AWS KMS 권한을 부여해야 합니다.

다음 예제는 AWS KMS 암호화가 활성화된 Amazon S3 버킷과 AWS Transfer Family 함께를 사용할 때 필요한 최소 권한을 부여하는 IAM 정책을 보여줍니다. 사용하고 있는 경우, 사용자 IAM 역할 정책과 세션 정책 모두에 이 예 정책을 포함시킵니다.

```
{
  "Sid": "Stmt1544140969635",
  "Action": [
    "kms:Decrypt",
    "kms:Encrypt",
    "kms:GenerateDataKey",
    "kms:GetPublicKey",
    "kms:ListKeyPolicies"
  ],
  "Effect": "Allow",
  "Resource": "arn:aws:kms:region:account-id:key/kms-key-id"
}
```

Note

이 정책에서 지정하는 KMS 키 ID는 1단계에서 기본 암호화에 대해 지정한 것과 같아야 합니다.

루트 또는 사용자에게 사용되는 IAM 역할은 AWS KMS 키 정책에서 허용되어야 합니다. AWS KMS 키 정책에 대한 자세한 내용은 AWS Key Management Service 개발자 안내서의 [KMS에서 AWS 키 정책 사용을](#) 참조하세요.

AWS Transfer Family 저장 시 암호화

AWS Transfer Family 는 파일 전송 서비스이므로 저장 스토리지 데이터를 관리하지 않습니다. 가 AWS Transfer Family 지원하는 스토리지 서비스 및 시스템은 해당 상태의 데이터를 보호할 책임이 있습니다. 그러나에서 저장 데이터를 관리하는 몇 가지 서비스 관련 데이터가 AWS Transfer Family 있습니다.

무엇을 암호화하나요?

유틸리티를 AWS Transfer Family 처리하는 유일한 데이터는 파일 전송 서버를 운영하고 전송을 처리하는 데 필요한 세부 정보와 관련이 있습니다.는 Amazon DynamoDB에 전체 유틸리티 암호화를 사용하여 다음 데이터를 AWS Transfer Family 저장합니다.

- 서버 구성(예: 서버 설정, 프로토콜 구성 및 엔드포인트 세부 정보).

- SSH 퍼블릭 키 및 사용자 메타데이터를 포함한 사용자 인증 데이터.
- 워크플로 실행 세부 정보 및 단계 구성.
- 타사 시스템의 커넥터 구성 및 인증 자격 증명. 이러한 자격 증명은 AWS Transfer Family 관리형 암호화 키를 사용하여 암호화됩니다.

키 관리

가 서버 실행 및 전송 처리와 관련된 정보를 DynamoDB에 저장하는 데 AWS Transfer Family 사용하는 암호화 키는 관리할 수 없습니다. 이 정보에는 서버 구성, 사용자 인증 데이터, 워크플로 세부 정보 및 커넥터 자격 증명이 포함됩니다.

암호화되지 않는 것은 무엇인가요?

AWS Transfer Family 는 저장 시 스토리지 데이터를 암호화하는 방법을 제어하지 않지만 지원하는 최고 수준의 보안으로 스토리지 위치를 구성하는 것이 좋습니다. 예를 들어 Amazon S3 관리형 암호화 키(SSE-S3) 또는 AWS KMS 키(SSE-KMS)를 사용하여 객체를 암호화할 수 있습니다.

AWS 스토리지 서비스가 저장 데이터를 암호화하는 방법에 대해 자세히 알아봅니다.

- [Amazon S3](#)
- [Amazon EFS](#)

Transfer Family에서 SSH 및 PGP 키 관리

이 섹션에서는 SSH 키를 생성하는 방법과 키를 회전하는 방법 등 SSH 키에 대한 정보를 확인할 수 있습니다. 에서 Transfer Family를 사용하여 키를 관리하는 AWS Lambda 방법에 대한 자세한 내용은 블로그 게시물 [AWS Transfer Family 및를 사용하여 사용자 셀프 서비스 키 관리 활성화 AWS Lambda](#)를 참조하세요. 여러 SSH 키가 있는 사용자의 자동 배포 및 관리는 섹션을 참조하세요 [Transfer Family Terraform 모듈](#).

Note

AWS Transfer Family 는 SSH 인증을 위해 RSA, ECDSA 및 ED25519 키를 허용합니다.

이 섹션에서는 프리티 굿 프라이버시 (PGP) 키를 생성하고 관리하는 방법도 다룹니다.

다양한 사용 사례에 대한 권장 사항을 포함하여 지원되는 모든 암호화 및 키 알고리즘에 대한 포괄적인 개요는 섹션을 참조하세요 [암호화 및 키 알고리즘 개요](#).

암호화 및 키 알고리즘 개요

AWS Transfer Family 는 다양한 목적으로 다양한 유형의 알고리즘을 지원합니다. 특정 사용 사례에 사용할 알고리즘을 이해하면 안전하고 호환되는 파일 전송을 보장하는 데 도움이 됩니다.

알고리즘 빠른 참조

사용 사례	권장 알고리즘	FIPS 규정 준수	참고
SSH/SFTP 인증	RSA(rsa-sha2-256/512), ECDSA 또는 ED25519	RSA: 예, ECDSA: 예, ED25519: 아니요	모든 SSH 클라이언트 및 서버와 호환
PGP 키 생성	RSA 또는 ECC(NIST)	예	워크플로 복호화의 경우
PGP 파일 암호화	AES-256	예	PGP 소프트웨어에서 결정

SSH 인증 알고리즘

이러한 알고리즘은 클라이언트와 AWS Transfer Family 서버 간의 SSH/SFTP 인증에 사용됩니다. 사용자 인증 또는 서버 호스트 키를 위한 SSH 키 페어를 생성할 때 다음 중 하나를 선택합니다.

RSA(권장)

모든 SSH 클라이언트 및 서버와 호환되며 FIPS를 준수합니다. 보안 강화를 위해 SHA-2 해싱과 함께 사용:

- rsa-sha2-256 - 대부분의 사용 사례에 권장
- rsa-sha2-512 - 더 높은 보안 옵션

ED25519

현대적이고 효율적입니다. 강력한 보안 기능을 갖춘 더 작은 키 크기:

- ssh-ed25519 - 빠르고 안전하지만 FIPS를 준수하지 않음

ECDSA

타원 곡선 옵션. 보안과 성능의 균형:

- `ecdsa-sha2-nistp256` - 표준 곡선
- `ecdsa-sha2-nistp384` - 보안 곡선 강화
- `ecdsa-sha2-nistp521` - 가장 높은 보안 곡선

Note

이전 보안 정책에 대해 `SHA1ssh-rsa`를 지원합니다. 자세한 내용은 [암호화 알고리즘](#)을 참조하세요.

올바른 SSH 알고리즘 선택

- 대부분의 사용자: `rsa-sha2-256` 또는와 함께 RSA 사용 `rsa-sha2-512`
- FIPS 규정 준수의 경우: RSA 또는 ECDSA 알고리즘 사용
- 최신 환경의 경우: ED25519는 뛰어난 보안 및 성능을 제공합니다.

PGP 암호화 및 복호화 알고리즘

PGP(Pretty Good Privacy)는 두 가지 유형의 알고리즘을 함께 사용하여 워크플로에서 파일을 암호화하고 해독합니다.

1. 키 페어 알고리즘 - 암호화 및 디지털 서명을 위한 퍼블릭/프라이빗 키 페어를 생성하는 데 사용됩니다.
2. 대칭 알고리즘 - 실제 파일 데이터를 암호화하는 데 사용됩니다(키 페어 알고리즘은 대칭 키를 암호화함).

PGP 키 페어 알고리즘

워크플로 복호화를 위해 PGP 키 페어를 생성할 때 다음 알고리즘 중 하나를 선택합니다.

RSA(권장)

대부분의 사용자에게 권장됩니다. 광범위하게 지원되고, 잘 확립되고, FIPS를 준수합니다. 보안과 호환성의 적절한 균형을 제공합니다.

ECC(Elliptic Curve Cryptography)

강력한 보안을 유지하면서 키 크기가 작은 RSA보다 더 효율적입니다.

- NIST 곡선 - 널리 지원되는 표준 곡선 및 FIPS 준수
- BrainPool 곡선 - 특정 규정 준수 요구 사항에 대한 대체 곡선
- Curve25519 - 효율적인 계산으로 강력한 보안을 제공하는 최신 고성능 곡선

ElGamal

레거시 알고리즘. 이전 시스템과의 호환성을 지원합니다. 새 구현에는 RSA 또는 ECC를 사용합니다.

PGP 키 생성에 대한 자세한 지침은 섹션을 참조하세요 [PGP 키 생성](#).

PGP 대칭 암호화 알고리즘

이러한 알고리즘은 실제 파일 데이터를 암호화합니다. 사용되는 알고리즘은 PGP 소프트웨어가 PGP 파일을 생성한 방식에 따라 달라집니다.

FIPS 준수 알고리즘(규정 환경에 권장)

- AES-128, AES-192, AES-256 - 고급 암호화 표준(권장)
- 3DES - 트리플 데이터 암호화 표준(레거시, 가능하면 AES 사용)

기타 지원되는 알고리즘

- IDEA, CAST5, Blowfish, DES, TwoFish, CAMELLIA-128, CAMELLIA-192, CAMELLIA-256

Note

워크플로를 사용할 AWS Transfer Family 때 대칭 알고리즘을 직접 선택하지 않습니다. 이는 암호화된 파일을 생성하는 데 사용되는 PGP 소프트웨어에 의해 결정됩니다. 그러나 AES-256과 같은 FIPS 호환 알고리즘을 선호하도록 PGP 소프트웨어를 구성할 수 있습니다.

지원되는 대칭 알고리즘에 대한 자세한 내용은 섹션을 참조하세요 [지원되는 대칭 암호화 알고리즘](#).

서비스 관리 사용자를 위한 SSH 키 생성

서버를 설정해 서비스 관리형 인증 메서드를 사용하는 사용자를 인증할 수 있습니다. 이때 사용자 이름과 SSH 키는 서비스 내에 보관됩니다. 사용자의 퍼블릭 SSH 키는 서버에 사용자의 속성으로 업로드됩니다. 이 키는 서버에서 표준 키 기반 인증 프로세스의 일부로 사용됩니다. 각 사용자는 파일에 여러 퍼블릭 SSH 키를 개별 서버와 함께 보유할 수 있습니다. 사용자당 저장할 수 있는 키 수에 대한 제한은 Amazon Web Services 일반 참조의 [AWS Transfer Family 엔드포인트 및 할당량](#)을 참조하세요.

서비스 관리형 인증 방법 대신 사용자 지정 자격 증명 공급자 또는를 사용하여 사용자를 인증할 수 있습니다 AWS Directory Service for Microsoft Active Directory. 자세한 내용은 [사용자 지정 자격 증명 공급자와 작업](#) 또는 [Microsoft Active Directory용 AWS Directory Service 사용](#)을 참조하세요.

서버는 한 가지 방법 (서비스 관리형, 디렉터리 서비스 또는 맞춤 ID 제공자)을 사용해서만 사용자를 인증할 수 있으며, 서버를 만든 후에는 이 방법을 변경할 수 없습니다.

주제

- [macOS, Linux 또는 Unix에서 SSH 키 생성](#)
- [Microsoft Windows에서 SSH 키 생성](#)
- [SSH2 키를 SSH 퍼블릭 키 형식으로 변환](#)

macOS, Linux 또는 Unix에서 SSH 키 생성

macOS, Linux 또는 Unix 작동 시스템에서는 ssh-keygen 명령을 사용하여 SSH 퍼블릭 키와 키 쌍이라고도 하는 SSH 프라이빗 키를 생성합니다.

Note

다음 예에서는 패스프레이즈를 지정하지 않았는데, 이 경우 도구에서 패스프레이즈를 입력한 다음 반복하여 확인하도록 요청합니다. 패스프레이즈를 만들면 프라이빗 키를 더 잘 보호할 수 있고 전반적인 시스템 보안도 향상될 수 있습니다. 패스프레이즈는 복구할 수 없습니다: 암호를 잊은 경우 새 키를 생성해야 합니다.

그러나 Transfer Family 서버는 시작 시 패스프레이즈를 요청할 수 없으므로 서버 호스트 키를 생성하는 경우 명령에서 -N "" 옵션을 지정하거나 메시지가 표시되면 **Enter**을 두 번 눌러 빈 암호를 지정해야 합니다.

macOS, Linux 또는 Unix 작동 시스템에서 SSH 키를 만들려면

1. macOS, Linux 또는 Unix 작동 시스템에서는 명령 터미널을 엽니다.
2. AWS Transfer Family 는 RSA, ECDSA 및 ED25519-formatted의 키를 허용합니다. 생성하려는 키 쌍의 타입에 따라 적절한 명령을 선택합니다.

팁: *key_name*를 SSH 키 페어 파일의 실제 이름으로 바꿉니다.

- RSA 4096비트 키 쌍 생성:

```
ssh-keygen -t rsa -b 4096 -f key_name
```

- ECDSA 521비트 키 쌍을 생성하려면 (ECDSA의 비트 크기는 256, 384, 521입니다):

```
ssh-keygen -t ecdsa -b 521 -f key_name
```

- ED25519 키 쌍을 생성하려면:

```
ssh-keygen -t ed25519 -f key_name
```

다음은 ssh-keygen 출력의 예입니다.

```
ssh-keygen -t rsa -b 4096 -f key_name
Generating public/private rsa key pair.

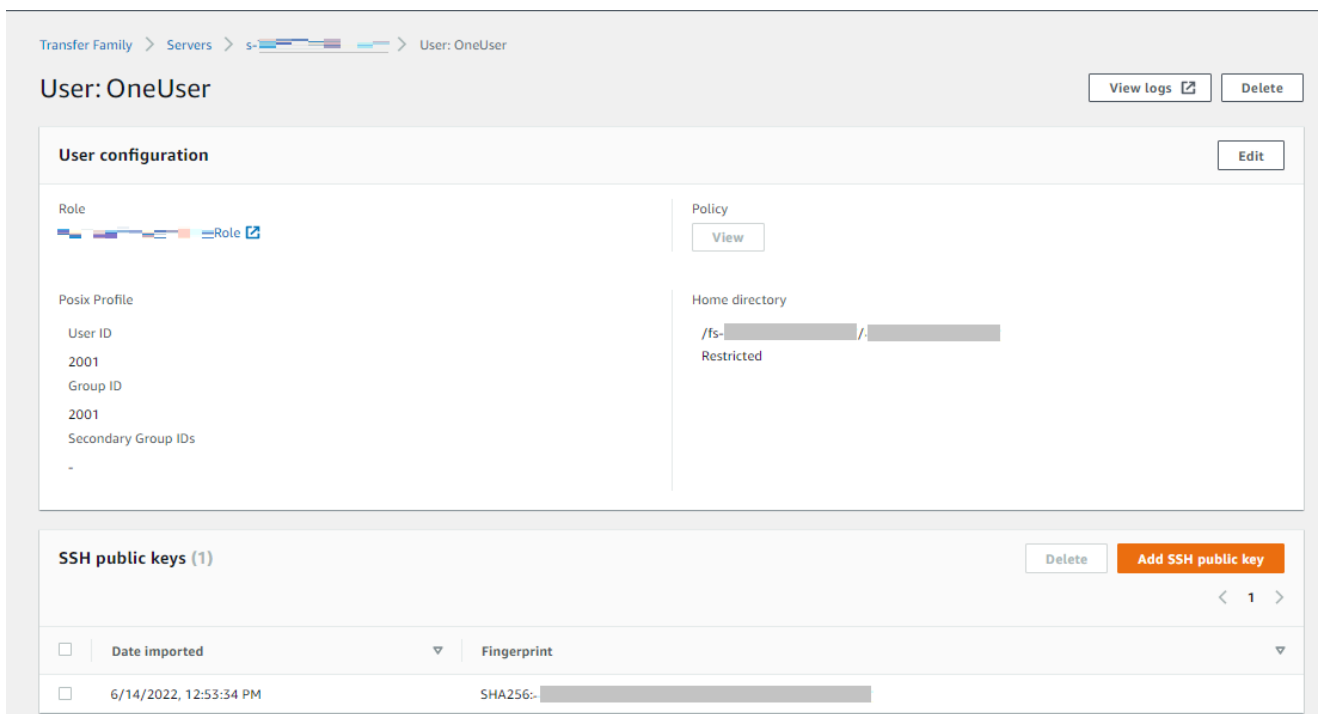
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in key_name.
Your public key has been saved in key_name.pub.
The key fingerprint is:
SHA256:8tDDwPmanTFcEzjTwPGETVW0GW1nVz+gtCCE8hL7PrQ bob.amazon.com
The key's randomart image is:
+---[RSA 4096]-----+
|  . ....E          |
| .   = ...         |
| . . . = ..o       |
| . o + oo =        |
| + = .S.= *        |
| . o o ..B + o     |
|   .o+. * .        |
|   =o**+.          |
```

```
| ..*o*+. |
+-----[SHA256]-----+
```

팁: 위와 같이 ssh-keygen 명령을 실행하면 퍼블릭 키와 프라이빗 키가 현재 디렉터리의 파일로 생성됩니다.

이제 SSH 키 페어를 사용할 준비가 되었습니다. 3단계와 4단계에 따라 서비스 관리형 사용자의 SSH 퍼블릭 키를 저장합니다. 이러한 사용자는 Transfer Family 서버 엔드포인트에서 파일을 전송할 때 키를 사용합니다.

3. **key_name**.pub 파일로 이동하여 엽니다.
4. 텍스트를 복사하여 서비스 관리 사용자의 SSH 퍼블릭 키에 붙여넣습니다.
 - a. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 연 다음 탐색 창에서 서버를 선택합니다.
 - b. 서버 페이지에서 업데이트하려는 사용자가 포함된 서버의 서버 ID를 선택합니다.
 - c. 퍼블릭 키를 추가할 사용자를 선택합니다.
 - d. SSH 퍼블릭 키 창에서 SSH 퍼블릭 키 추가를 선택합니다.



- e. 생성한 퍼블릭 키 텍스트를 SSH 퍼블릭 키 텍스트 상자에 붙여넣은 다음 키 추가를 선택합니다.

새 키는 SSH 퍼블릭 키 창에 나열됩니다.

	Date imported	Fingerprint
☐	6/14/2022, 12:53:34 PM	SHA256:-
☐	10/20/2022, 4:26:51 PM	SHA256:-

Microsoft Windows에서 SSH 키 생성

Windows에는 OpenSSH가 내장 기능으로 포함되어 있으며, 이를 사용하여 Linux 또는 macOS에서와 동일한 형식으로 SSH 키를 생성할 수 있습니다. 또는 PuTTY의 키 생성기(PuTTYgen)와 같은 타사 도구를 사용할 수 있습니다.

Windows 내장 OpenSSH 사용

Windows의 최신 버전에는 기본적으로 OpenSSH가 포함됩니다. macOS/Linux 섹션에 설명된 것과 동일한 `ssh-keygen` 명령을 사용할 수 있습니다.

1. Windows PowerShell 또는 명령 프롬프트를 엽니다.
2. 생성하려는 키 유형에 따라 다음 명령 중 하나를 실행합니다.

- RSA 4096비트 키 쌍 생성:

```
ssh-keygen -t rsa -b 4096 -f key_name
```

- ECDSA 521비트 키 페어를 생성하려면

```
ssh-keygen -t ecdsa -b 521 -f key_name
```

- ED25519 키 쌍을 생성하려면:

```
ssh-keygen -t ed25519 -f key_name
```

3. macOS/Linux 섹션과 동일한 단계에 따라 퍼블릭 키를 업로드합니다 AWS Transfer Family.

PuTTYgen(타사 도구) 사용

PuTTY와 같은 Windows용 일부 타사 SSH 클라이언트는 서로 다른 키 형식을 사용합니다. PuTTY는 프라이빗 키에 PPK 형식을 사용합니다. PuTTY 또는 WinSCP와 같은 관련 도구를 사용하는 경우 PuTTYgen을 사용하여 이 형식으로 키를 생성할 수 있습니다.

Note

WinSCP를 .ppk 형식이 아닌 프라이빗 키 파일과 함께 제공하면, 클라이언트는 키를 .ppk 형식으로 변환할 기회를 제공합니다.

PuTTYgen을 사용하여 SSH 키를 생성하는 방법에 대한 자습서는 [SSH.com 웹 사이트를](https://www.ssh.com/ssh-key/) 참조하십시오.

SSH2 키를 SSH 퍼블릭 키 형식으로 변환

AWS Transfer Family는 SSH 형식의 퍼블릭 키만 허용합니다. SSH2 퍼블릭 키가 있는 경우 이를 변환해야 합니다. SSH2 퍼블릭 키의 형식은 다음과 같습니다:

```
----- BEGIN SSH2 PUBLIC KEY -----
Comment: "rsa-key-20160402"
AAAAB3NzaC1yc2EAAAABJQAAAQEAiL0jjDdFqK/kYThqKt7THrjABTPWvXmB3URI
:
:
----- END SSH2 PUBLIC KEY -----
```

SSH 퍼블릭 키의 형식은 다음과 같습니다.

```
ssh-rsa AAAAB3NzaC1yc2EAAAABJQAAA...
```

다음 명령을 실행하여 SSH2-formatted의 퍼블릭 키를 SSH 형식의 퍼블릭 키로 변환합니다. **ssh2-key**를 SSH2 키 이름으로 바꾸고 **ssh-key**를 SSH 키 이름으로 바꿉니다.

```
ssh-keygen -i -f ssh2-key.pub > ssh-key.pub
```

SSH 키 교체

보안상 SSH 키 교체는 보안 모범 사례로 권장됩니다. 일반적으로 이러한 교체는 보안 정책의 일부로 지정되며, 자동화된 방식으로 구현됩니다. 보안 수준에 따라, 대단히 민감한 통신의 경우 SSH 키 쌍을 한 번만 사용할 수도 있습니다. 이렇게 하면 저장된 키 때문에 발생하는 모든 위험을 제거할 수 있습니다. 하지만 SSH 보안 인증을 일정 기간 보관하고, SFTP 사용자에게 과도한 부담을 주지 않는 주기를 설정하는 방법이 더 보편적입니다. 가장 일반적인 주기는 3개월입니다.

Note

코드형 인프라를 사용한 자동화된 SSH 키 교체는 섹션을 참조하세요 [Transfer Family Terraform 모듈](#).

SSH 키 교체는 2가지 방식으로 실행합니다.

- 콘솔에서 새 SSH 퍼블릭 키를 업로드하고 기존 SSH 퍼블릭 키를 삭제할 수 있습니다.
- API를 사용하면 [DeletesSHPublicKey](#) API를 사용하여 사용자의 SSH (Secure Shell) 퍼블릭 키를 삭제하고 [ImportSSHPublic Key](#) API를 사용하여 사용자 계정에 새 Secure Shell(SSH) 퍼블릭 키를 추가하여 기존 사용자를 업데이트할 수 있습니다.

Console

콘솔에서 키 로테이션을 수행하려면

1. <https://console.aws.amazon.com/transfer/> AWS Transfer Family 콘솔을 엽니다.
2. 서버 페이지로 이동합니다.
3. 서버 ID 옆에서 식별자를 선택하여 서버 세부 정보 페이지를 표시합니다.

4. 사용자에서 SSH 퍼블릭 키를 교체하려는 사용자의 확인란을 선택한 다음 작업을 선택하고 키 추가를 선택하여 키 추가 페이지를 확인합니다.

또는

사용자 세부 정보 페이지를 보려면 사용자 이름을 선택한 다음 SSH 퍼블릭 키 추가를 선택하여 키 추가 페이지를 확인합니다.

5. 새 SSH 퍼블릭 키를 입력하고 키 추가를 선택합니다.

Important

SSH 퍼블릭 키의 형식은 생성한 키 타입에 따라 다릅니다.

- RSA 키의 경우, 그 형식은 ssh-rsa *string*입니다.
- ED25519 키의 경우, 그 형식은 ssh-ed25519 *string*입니다.
- ECDSA 키의 경우, 키는 생성한 키 크기에 따라 ecdsa-sha2-nistp256, ecdsa-sha2-nistp384, 또는 ecdsa-sha2-nistp521(으)로 시작합니다. 그런 다음 다른 키 타입과 마찬가지로 시작 문자열 뒤에 *string*이(가) 옵니다.

사용자 세부 정보 페이지로 돌아오면 방금 입력한 새 SSH 퍼블릭 키가 SSH 퍼블릭 키 섹션에 표시됩니다.

6. 삭제할 보안 인증 옆의 확인란을 선택하고 삭제를 선택합니다.
7. 단어 delete를 입력하여 삭제 작업을 확인한 다음 삭제를 선택합니다.

API

API를 사용하여 키 로테이션을 수행하려면

1. macOS, Linux 또는 Unix 작동 시스템에서는 명령 터미널을 엽니다.
2. 다음 명령을 입력하여 삭제하려는 SSH 키를 검색합니다. 이 명령을 사용하려면, *serverID*를 Transfer Family 서버의 서버 ID로 바꾸고 *username*를 사용자 이름으로 바꾸세요.

```
aws transfer describe-user --server-id='serverID' --user-name='username'
```

이 명령은 사용자에 대한 세부 정보를 반환합니다. "SshPublicKeyId": 필드 내용을 복사합니다. 이 값은 이 절차의 뒷부분에서 입력해야 합니다.

```
"SshPublicKeys": [ { "SshPublicKeyBody": "public-key", "SshPublicKeyId":
"keyID",
    "DateImported": 1621969331.072 } ],
```

3. 다음으로, 사용자의 새 SSH 키를 가져옵니다. 프롬프트에 다음 명령을 입력합니다. 이 명령을 사용하려면 *serverID*를 Transfer Family 서버의 서버 ID로 바꾸고 *username*를 사용자 이름으로 바꾸고 *public-key*를 새 퍼블릭 키의 지문으로 바꾸세요.

```
aws transfer import-ssh-public-key --server-id='serverID' --user-name='username'
--ssh-public-key-body='public-key'
```

이 명령이 성공하면 출력이 반환되지 않습니다.

4. 마지막으로 다음 명령을 실행하여 이전 키를 삭제합니다. 이 명령을 사용하려면 *serverID*를 Transfer Family 서버의 서버 ID로 바꾸고 *username*를 사용자 이름으로 바꾸고 *keyID-from-step-2*를 이 절차의 2단계에서 복사한 키 ID 값으로 바꾸세요.

```
aws transfer delete-ssh-public-key --server-id='serverID' --user-name='username'
--ssh-public-key-id='keyID-from-step-2'
```

5. (옵션) 이전 키가 더 이상 존재하지 않는지 확인하려면 2단계를 반복합니다.

PGP 키 생성

Transfer Family가 워크플로를 통해 처리하는 파일에 프리티 굿 프라이버시(PGP) 복호화를 사용할 수 있습니다. 워크플로 단계에서 복호화를 사용하려면 PGP 키를 제공합니다. 권장 사항 및 FIPS 규정 준수를 포함하여 PGP 키 알고리즘에 대한 자세한 내용은 섹션을 참조하세요 [PGP 키 페어 알고리즘](#).

AWS 스토리지 블로그에는 Transfer Family 관리형 워크플로, [PGP 및를 사용한 암호화 및 암호화 해제를 사용하여 코드를 작성하지 않고 파일을 간단하게 해독하는 방법을 설명하는 게시물이 있습니다 AWS Transfer Family](#).

PGP 키를 생성하는 데 사용하는 연산자는 운영 체제와 사용 중인 키 생성 소프트웨어의 버전에 따라 다릅니다.

Linux 또는 Unix를 사용하는 경우 패키지 설치 프로그램을 사용하여 gpg를 설치하세요. Linux 배포판에 따라 다음 명령 중 하나를 사용할 수 있습니다.

```
sudo yum install gnupg
```



```
sudo apt-get install gnupg
```

Windows 또는 macOS의 경우, <https://gnupg.org/download/>에서 필요한 것을 다운로드할 수 있습니다.

PGP 키 생성기 소프트웨어를 설치한 후 `gpg --full-gen-key` 또는 `gpg --gen-key` 명령을 실행하여 키 쌍을 생성합니다.

Note

GnuPG 버전 2.3.0 이상을 사용하는 경우 `gpg --full-gen-key`를 실행해야 합니다. 생성할 키 타입을 묻는 메시지가 표시되면 RSA 또는 ECC를 선택합니다. ECC를 선택하는 경우 타원 곡선에 Curve25519 대해 NIST BrainPool 및 중에서 선택할 수 있습니다.

유용한 **gpg** 하위 명령

다음은 gpg에 유용한 몇 가지 하위 명령입니다:

- `gpg --help` – 이 명령에는 사용 가능한 옵션이 나열되며 몇 가지 예가 포함될 수 있습니다.
- `gpg --list-keys` – 이 명령은 생성한 모든 키 페어의 세부 정보를 나열합니다.
- `gpg --fingerprint` – 이 명령은 각 키의 지문을 포함하여 모든 키 페어에 대한 세부 정보를 나열합니다.
- `gpg --export -a user-name` – 이 명령은 키가 생성될 때 사용된 *user-name* 키의 퍼블릭 키 부분을 내보냅니다.

PGP 키 관리

PGP 키를 관리하려면 AWS Secrets Manager를 사용합니다.

Note

비밀 명칭에는 Transfer Family 서버 ID가 포함됩니다. 즉, PGP 키 정보를 AWS Secrets Manager에 저장하기에 앞서 먼저 서버를 식별하거나 생성했어야 합니다.

모든 사용자에게 대해 하나의 키와 패스프레이즈를 사용하려는 경우 PGP 키 블록 정보를 비밀 명칭 `aws/transfer/server-id/epgp-default` 아래에 저장할 수 있습니다. 여기서 *server-id*은

(는) Transfer Family 서버의 ID입니다. Transfer Family는가 워크플로를 실행하는 사용자와 **user-name** 일치하는 키가 없는 경우이 기본 키를 사용합니다.

특정 사용자에게 대한 키를 생성할 수 있습니다. 이 경우 보안 암호 이름의 형식은 `aws/transfer/server-id/user-name`, 여기서는 Transfer Family 서버의 워크플로를 실행하는 사용자와 **user-name** 일치합니다.

Note

사용자당 Transfer Family 서버당 최대 3개의 PGP 프라이빗 키를 저장할 수 있습니다.

복호화에 사용할 PGP 키를 구성하려면

1. 사용 중인 GPG 버전에 따라 다음 명령 중 하나를 실행하여 PGP 키 페어를 생성합니다.

- **GnuPG** 버전 2.3.0 이상을 사용하는 경우, 다음 명령을 실행합니다.

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

RSA를 선택하거나,를 선택한 경우 타원 곡선**Curve25519**에 대해 **NIST BrainPool** 또는를 선택할 **ECC**수 있습니다. `gpg --gen-key` 대신를 실행하는 경우 ECC Curve 25519 암호화 알고리즘을 사용하는 키 페어를 생성합니다.

- RSA가 기본 암호화 타입이므로 2.3.0 이전의 **GnuPG** 버전의 경우, 다음 명령을 사용할 수 있습니다.

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

Important

키 생성 프로세스 중에 패스프레이즈와 이메일 주소를 제공해야 합니다. 이러한 값을 기록해 두세요. 이 절차의 AWS Secrets Manager 뒷부분에 키의 세부 정보를 입력할 때 암호를 제공해야 합니다. 그리고 다음 단계에서 프라이빗 키를 내보내려면 동일한 이메일 주소를 제공해야 합니다.

2. 프라이빗 키를 내보내려면 다음 명령을 실행합니다. 이 명령을 사용하려면 *private.pgp*를 프라이빗 키 블록을 저장할 파일 명칭으로, *marymajor@example.com*를 키 쌍을 생성할 때 사용한 이메일 주소로 바꾸세요.

```
gpg --output private.pgp --armor --export-secret-key marymajor@example.com
```

3. AWS Secrets Manager 를 사용하여 PGP 키를 저장합니다.
 - a. 에 로그인 AWS Management Console 하고 <https://console.aws.amazon.com/secretsmanager/> AWS Secrets Manager 콘솔을 엽니다.
 - b. 왼쪽 탐색 창에서 암호를 선택합니다.
 - c. 암호 페이지에서 새 암호 저장을 선택합니다.
 - d. 암호 타입 선택 페이지의 암호 타입에 대해 다른 타입의 암호를 선택합니다.
 - e. 키/값 쌍 섹션에서 키/값 탭을 선택합니다.
 - 키 – **PGPPrivateKey**를 입력합니다.

 Note

PGPPrivateKey 문자열을 정확히 입력해야 합니다: 문자 앞이나 사이에 공백을 추가하지 마세요.

- 값 — 프라이빗 키의 텍스트를 값 필드에 붙여넣습니다. 이 절차의 앞부분에서 키를 내보낼 때 지정한 파일 (예: *private.pgp*) 에서 프라이빗 키의 텍스트를 찾을 수 있습니다. 키는 -----BEGIN PGP PRIVATE KEY BLOCK----- (으)로 시작하고 -----END PGP PRIVATE KEY BLOCK----- (으)로 끝납니다.

 Note

텍스트 블록에는 프라이빗 키만 포함되고 퍼블릭 키는 포함되지 않는지 확인하세요.

- f. 행 추가를 선택하고 키/값 쌍 섹션에서 키/값 탭을 선택합니다.
 - 키 – **PGPPassphrase**를 입력합니다.

Note

PGPPassphrase 문자열을 정확히 입력해야 합니다: 문자 앞이나 사이에 공백을 추가하지 마세요.

- 값 – PGP 키 쌍을 생성할 때 사용한 패스프레이즈를 입력합니다.

Choose secret type

Secret type [Info](#)

☐ Credentials for Amazon RDS database
 ☐ Credentials for Amazon DocumentDB database
 ☐ Credentials for Amazon Redshift cluster
 ☒ 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

최대 3세트의 키와 패스프레이즈를 추가할 수 있습니다. 두 번째 세트를 추가하려면 새 행 두 개를 추가하고 키에 **PGPPrivateKey2**와(과) **PGPPassphrase2**를 입력한 다음 다른 프라이빗 키와 패스프레이즈를 붙여넣습니다. 세 번째 세트를 추가하려면 키 값이 **PGPPrivateKey3** 및 **PGPPassphrase3**여야 합니다.

- g. 다음을 선택합니다.
- h. 암호 구성 페이지에 암호를 위한 명칭과 설명을 입력합니다.

- 기본 키, 즉 모든 Transfer Family 사용자가 사용할 수 있는 키를 만들려면 다음을 입력하세요 **aws/transfer/*server-id*/pgp-default**. *server-id*를 복호화 단계가 있는 워크플로가 포함된 서버의 ID로 바꾸세요.
- 특정 Transfer Family 사용자가 사용할 키를 만들려면 **aws/transfer/*server-id*/*user-name***를 입력하세요. *server-id*를 복호화 단계가 있는 워크플로가 포함된 서버의 ID로 바꾸고 *user-name*를 워크플로를 실행하는 사용자 이름으로 바꾸세요. *user-name*은(는) Transfer Family 서버가 사용하는 ID 제공자에 저장됩니다.
 - 다음을 선택하고 순환 구성 페이지의 기본값을 그대로 사용합니다. 그 다음 다음을 선택합니다.
 - 검토 페이지에서 저장을 선택하여 암호를 만들고 저장합니다.

다음 스크린샷은 특정 Transfer Family 서버의 사용자 **marymajor**의 세부 정보를 보여줍니다. 이 예에서는 세 개의 키와 해당 패스프레이즈를 보여줍니다.

The screenshot shows the AWS Secrets Manager console for a secret named `/aws/transfer/s-[redacted]/marymajor`. The secret is located in the `us-east-2` region. The secret value is displayed in a table with two columns: **Secret key** and **Secret value**. The secret value is a PGP private key block, which is a long string of text. The secret is also associated with a description: "Contains the PGP secret keys and corresponding passphrases to use for user marymajor on Transfer Family server s-[redacted]".

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

지원되는 PGP 클라이언트

다음 클라이언트는 Transfer Family에서 테스트되었으며 PGP 키를 생성하고 워크플로를 통해 해독하려는 파일을 암호화하는 데 사용할 수 있습니다.

- Gpg4win + Kleopatra.

Note

파일 서명/암호화를 선택할 때는 서명 주체 선택을 취소해야 합니다: 현재 암호화된 파일에 대한 서명은 지원하지 않습니다.

Sign / Encrypt Files

Prove authenticity (sign)

☐ Sign as: [Key] > (certified, created: 11/9/2022)

Encrypt

☒ Encrypt for me: [Key] > (certified, created: 11/9/2022)

☐ Encrypt for others: [Please enter a name or email address...]

☐ Encrypt with password. Anyone you share the password with can read the data.

Output

Output files/folder:

[C:/Users/.../txt.gpg]

☐ Encrypt / Sign each file separately.

Encrypt **Cancel**

암호화된 파일에 서명하고 복호화 워크플로를 사용하여 Transfer Family 서버에 업로드하려고 하면 다음 오류가 발생합니다.

Encrypted file with signed message unsupported

- 주요 GnuPG 버전: 2.4, 2.3, 2.2, 2.0, 1.4.

다른 PGP 클라이언트도 작동할 수 있지만 여기에 언급된 클라이언트만 Transfer Family로 테스트되었습니다.

에 대한 자격 증명 및 액세스 관리 AWS Transfer Family

AWS Identity and Access Management (IAM)는 관리자가 AWS 리소스에 대한 액세스를 안전하게 제어하는 데 도움이 되는 AWS 서비스입니다. IAM 관리자는 누가 AWS Transfer Family 리소스를 사용할 수 있는 인증(로그인) 및 권한(권한 있음)을 받을 수 있는지 제어합니다. IAM은 추가 비용 없이 사용할 수 있는 AWS 서비스입니다.

주제

- [대상](#)
- [ID를 통한 인증](#)
- [정책을 사용하여 액세스 관리](#)
- [AWS Transfer Family 에서 IAM을 사용하는 방법](#)
- [AWS Transfer Family 자격 증명 기반 정책 예제](#)
- [AWS Transfer Family 태그 기반 정책 예제](#)
- [AWS Transfer Family 자격 증명 및 액세스 문제 해결](#)
- [조직 거버넌스를 위한 IAM 조건 키](#)

대상

AWS Identity and Access Management (IAM)를 사용하는 방법은 역할에 따라 다릅니다.

- 서비스 사용자 - 기능에 액세스할 수 없는 경우 관리자에게 권한 요청([참조 AWS Transfer Family 자격 증명 및 액세스 문제 해결](#))
- 서비스 관리자 - 사용자 액세스 결정 및 권한 요청 제출([AWS Transfer Family 에서 IAM을 사용하는 방법](#) 참조)
- IAM 관리자 - 액세스를 관리하기 위한 정책 작성([AWS Transfer Family 자격 증명 기반 정책 예제](#) 참조)

ID를 통한 인증

인증은 자격 증명 자격 증명을 AWS 사용하여 로그인하는 방법입니다. AWS 계정 루트 사용자, IAM 사용자 또는 IAM 역할을 수임하여 인증되어야 합니다.

AWS IAM Identity Center (IAM Identity Center), Single Sign-On 인증 또는 Google/Facebook 자격 증명과 같은 자격 증명 소스의 자격 증명을 사용하여 페더레이션 자격 증명으로 로그인할 수 있습니다. 로그인하는 방법에 대한 자세한 내용은 AWS 로그인 사용 설명서의 [AWS 계정에 로그인하는 방법](#) 섹션을 참조하세요.

프로그래밍 방식 액세스를 위해서는 요청에 암호화 방식으로 서명할 수 있는 SDK 및 CLI를 AWS 제공합니다. 자세한 내용은 IAM 사용 설명서의 [API 요청용 AWS Signature Version 4](#) 섹션을 참조하세요.

AWS 계정 루트 사용자

를 생성할 때 모든 AWS 서비스 및 리소스에 대한 완전한 액세스 권한이 있는 AWS 계정 theroot 사용자라는 하나의 로그인 자격 증명으로 AWS 계정시작합니다. 일상적인 태스크에 루트 사용자를 사용하지 않을 것을 강력히 권장합니다. 루트 사용자 자격 증명이 필요한 작업은 IAM 사용 설명서의 [루트 사용자 자격 증명이 필요한 작업](#) 섹션을 참조하세요.

페더레이션 ID

가장 좋은 방법은 인간 사용자에게 자격 증명 공급자와의 페더레이션을 사용하여 임시 자격 증명을 AWS 서비스 사용하여 액세스하도록 요구하는 것입니다.

페더레이션 자격 증명은 엔터프라이즈 디렉터리, 웹 자격 증명 공급자 또는 자격 증명 소스의 자격 증명을 AWS 서비스 사용하여 Directory Service 에 액세스하는 사용자입니다. 페더레이션 ID는 임시 자격 증명을 제공하는 역할을 수임합니다.

중앙 집중식 액세스 관리를 위해 AWS IAM Identity Center를 추천합니다. 자세한 정보는 AWS IAM Identity Center 사용 설명서의 [What is IAM Identity Center?](#)를 참조하세요.

IAM 사용자 및 그룹

[IAM 사용자](#)는 단일 개인 또는 애플리케이션에 대한 특정 권한을 가진 ID입니다. 장기 자격 증명에 있는 IAM 사용자 대신 임시 자격 증명을 사용하는 것이 좋습니다. 자세한 내용은 IAM 사용 설명서의 [자격 증명 공급자와의 페더레이션을 사용하여 임시 자격 증명을 AWS 사용하여 액세스하도록 인간 사용자에게 요구하기](#)를 참조하세요.

[IAM 그룹](#)은 IAM 사용자 모음을 지정하고 대규모 사용자 집합에 대한 관리 권한을 더 쉽게 만듭니다. 자세한 내용은 IAM 사용 설명서의 [IAM 사용자 사용 사례](#) 섹션을 참조하세요.

IAM 역할

[IAM 역할](#)은 임시 자격 증명을 제공하는 특정 권한이 있는 자격 증명입니다. [사용자에서 IAM 역할\(콘솔\)로 전환하거나 또는 API 작업을 호출하여 역할을](#) 수임할 수 있습니다. AWS CLI AWS 자세한 내용은 IAM 사용 설명서의 [역할 수임 방법](#)을 참조하세요.

IAM 역할은 페더레이션 사용자 액세스, 임시 IAM 사용자 권한, 교차 계정 액세스, 교차 서비스 액세스 및 Amazon EC2에서 실행되는 애플리케이션에 유용합니다. 자세한 내용은 IAM 사용 설명서의 [교차 계정 리소스 액세스](#)를 참조하세요.

정책을 사용하여 액세스 관리

정책을 AWS 생성하고 자격 증명 또는 리소스에 연결하여 AWS 에서 액세스를 제어합니다. 정책은 자격 증명 또는 리소스와 연결될 때 권한을 정의합니다.는 보안 주체가 요청할 때 이러한 정책을 AWS 평가합니다. 대부분의 정책은 JSON 문서 AWS 로 저장됩니다. JSON 정책 문서에 대한 자세한 내용은 IAM 사용 설명서의 [JSON 정책 개요](#) 섹션을 참조하세요.

정책을 사용하여 관리자는 어떤 보안 주체가 어떤 리소스에 대해 어떤 조건에서 작업을 수행할 수 있는지 정의하여 누가 무엇을 액세스할 수 있는지 지정합니다.

기본적으로 사용자 및 역할에는 어떠한 권한도 없습니다. IAM 관리자는 IAM 정책을 생성하고 사용자가 수임할 수 있는 역할에 추가합니다. IAM 정책은 작업을 수행하기 위해 사용하는 방법과 관계없이 작업에 대한 권한을 정의합니다.

ID 기반 정책

ID 기반 정책은 ID(사용자, 사용자 그룹 또는 역할)에 연결하는 JSON 권한 정책 문서입니다. 이러한 정책은 자격 증명이 수행할 수 있는 작업, 대상 리소스 및 이에 관한 조건을 제어합니다. ID 기반 정책을 생성하는 방법을 알아보려면 IAM 사용 설명서에서 [고객 관리형 정책으로 사용자 지정 IAM 권한 정의](#)를 참조하세요.

ID 기반 정책은 인라인 정책(단일 ID에 직접 포함) 또는 관리형 정책(여러 ID에 연결된 독립 실행형 정책)일 수 있습니다. 관리형 정책 또는 인라인 정책을 선택하는 방법을 알아보려면 IAM 사용 설명서의 [관리형 정책 및 인라인 정책 중에서 선택](#) 섹션을 참조하세요.

리소스 기반 정책

리소스 기반 정책은 리소스에 연결하는 JSON 정책 설명서입니다. 예를 들어 IAM 역할 신뢰 정책 및 Amazon S3 버킷 정책이 있습니다. 리소스 기반 정책을 지원하는 서비스에서 서비스 관리자는 이러한 정책을 사용하여 특정 리소스에 대한 액세스를 통제할 수 있습니다. 리소스 기반 정책에서 [보안 주체를 지정](#)해야 합니다.

리소스 기반 정책은 해당 서비스에 있는 인라인 정책입니다. 리소스 기반 정책에서는 IAM의 AWS 관리형 정책을 사용할 수 없습니다.

액세스 제어 목록(ACL)

액세스 제어 목록(ACL)은 어떤 위탁자(계정 멤버, 사용자 또는 역할)가 리소스에 액세스할 수 있는 권한을 가지고 있는지를 제어합니다. ACL은 JSON 정책 문서 형식을 사용하지 않지만 리소스 기반 정책과 유사합니다.

Amazon S3 AWS WAF 및 Amazon VPC는 ACLs. ACL에 관한 자세한 내용은 Amazon Simple Storage Service 개발자 가이드의 [액세스 제어 목록\(ACL\) 개요](#)를 참조하세요.

기타 정책 타입

AWS 는 보다 일반적인 정책 유형에서 부여한 최대 권한을 설정할 수 있는 추가 정책 유형을 지원합니다.

- 권한 경계 - ID 기반 정책에서 IAM 엔터티에 부여할 수 있는 최대 권한을 설정합니다. 자세한 정보는 IAM 사용 설명서의 [IAM 엔터티의 권한 범위](#)를 참조하세요.
- 서비스 제어 정책(SCP) - AWS Organizations내 조직 또는 조직 단위에 대한 최대 권한을 지정합니다. 자세한 내용은 AWS Organizations 사용 설명서의 [서비스 제어 정책](#)을 참조하세요.
- 리소스 제어 정책(RCP) - 계정의 리소스에 사용할 수 있는 최대 권한을 설정합니다. 자세한 내용은 AWS Organizations 사용 설명서의 [리소스 제어 정책\(RCP\)](#)을 참조하세요.
- 세션 정책 - 역할 또는 페더레이션 사용자에게 대해 임시 세션을 프로그래밍 방식으로 생성할 때 파라미터로 전달하는 고급 정책입니다. 자세한 내용은 IAM 사용 설명서의 [세션 정책](#)을 참조하세요.

여러 정책 유형

여러 정책 유형이 요청에 적용되는 경우, 결과 권한은 이해하기가 더 복잡합니다. 에서 여러 정책 유형이 관련될 때 요청을 허용할지 여부를 AWS 결정하는 방법을 알아보려면 IAM 사용 설명서의 [정책 평가 로직](#)을 참조하세요.

AWS Transfer Family 에서 IAM을 사용하는 방법

AWS Identity and Access Management (IAM)을 사용하여 액세스를 관리하기 전에 어떤 IAM 기능을 사용할 수 있는지 이해해야 AWS Transfer Family합니다 AWS Transfer Family. AWS Transfer Family 및 기타 AWS 서비스에서 IAM을 사용하는 방법을 전체적으로 알아보려면 IAM 사용 설명서의 [AWS IAM으로 작업하는 서비스를](#) 참조하세요.

주제

- [AWS Transfer Family 자격 증명 기반 정책](#)
- [AWS Transfer Family 리소스 기반 정책](#)
- [AWS Transfer Family 태그 기반 권한 부여](#)
- [AWS Transfer Family IAM 역할](#)

AWS Transfer Family 자격 증명 기반 정책

IAM 보안 인증 기반 정책을 사용하면 허용되거나 거부되는 작업과 리소스 및 작업이 허용되거나 거부되는 조건을 지정할 수 있습니다. AWS Transfer Family 은(는) 특정 작업, 리소스 및 조건 키를 지원합니다. JSON 정책에서 사용하는 모든 요소에 대해 알아보려면 AWS Identity and Access Management 사용자 가이드의 [IAM JSON 정책 요소 참조](#)를 참조하세요.

작업

관리자는 AWS JSON 정책을 사용하여 누가 무엇에 액세스할 수 있는지 지정할 수 있습니다. 즉, 어떤 보안 주체가 어떤 리소스와 어떤 조건에서 작업을 수행할 수 있는지를 지정할 수 있습니다.

JSON 정책의 Action요소는 정책에서 액세스를 허용하거나 거부하는 데 사용할 수 있는 작업을 설명합니다. 연결된 작업을 수행할 수 있는 권한을 부여하기 위한 정책에 작업을 포함하세요.

의 정책 작업은 작업 앞에 접두사를 AWS Transfer Family 사용합니다. 예를 들어 Transfer Family CreateServer API 작업으로 서버를 생성할 수 있는 권한을 부여하려면 해당 정책에 `transfer:CreateServer` 작업을 포함합니다. 정책 명령문에는 Action 또는 NotAction 요소가 포함되어야 합니다. AWS Transfer Family 는 이 서비스로 수행할 수 있는 작업을 설명하는 고유한 작업 집합을 정의합니다.

명령문 하나에 여러 작업을 지정하려면 다음과 같이 쉼표로 구분합니다.

```
"Action": [
    "transfer:action1",
    "transfer:action2"
```

와일드카드(*)를 사용하여 여러 작업을 지정할 수 있습니다. 예를 들어, Describe라는 단어로 시작하는 모든 작업을 지정하려면 다음 작업을 포함합니다.

```
"Action": "transfer:Describe*"
```

AWS Transfer Family 작업 목록을 보려면 서비스 승인 참조의에서 [정의한 작업을 AWS Transfer Family](#) 참조하세요.

리소스

관리자는 AWS JSON 정책을 사용하여 누가 무엇에 액세스할 수 있는지 지정할 수 있습니다. 즉, 어떤 보안 주체가 어떤 리소스와 어떤 조건에서 작업을 수행할 수 있는지를 지정할 수 있습니다.

Resource JSON 정책 요소는 작업이 적용되는 하나 이상의 객체를 지정합니다. 모범 사례에 따라 [Amazon 리소스 이름\(ARN\)](#)을 사용하여 리소스를 지정합니다. 리소스 수준 권한을 지원하지 않는 작업의 경우, 와일드카드(*)를 사용하여 해당 문이 모든 리소스에 적용됨을 나타냅니다.

```
"Resource": ""
```

Transfer Family 서버 리소스에는 다음 ARN이 있습니다.

```
arn:aws:transfer:${Region}:${Account}:server/${ServerId}
```

예를 들어, 명령문에 s-01234567890abcdef Transfer Family 서버를 지정하려면 다음 ARN을 사용합니다.

```
"Resource": "arn:aws:transfer:us-east-1:123456789012:server/s-01234567890abcdef"
```

ARN 형식에 대한 자세한 설명은 서비스 승인 참조에서 [Amazon 리소스 이름\(ARN\)](#) 또는 IAM 사용자 가이드에서 [IAM ARN](#)을 참조하세요.

특정 계정에 속하는 모든 인스턴스를 지정하려면 와일드카드(*)를 사용합니다.

```
"Resource": "arn:aws:transfer:us-east-1:123456789012:server/*"
```

일부 AWS Transfer Family 작업은 IAM 정책에 사용되는 리소스와 같은 여러 리소스에서 수행됩니다. 이러한 경우, 와일드카드(*)를 사용해야 합니다.

```
"Resource": "arn:aws:transfer:*:123456789012:server/*"
```

Transfer Family 서버 및 사용자에게 대한 액세스를 허용하는 정책을 만드는 경우와 같이 둘 이상의 리소스 타입을 지정해야 하는 경우도 있습니다. 단일 명령문에서 여러 리소스를 지정하려면 ARN을 쉼표로 구분합니다.

```
"Resource": [
    "resource1",
    "resource2"
]
```

AWS Transfer Family 리소스 목록을 보려면 서비스 승인 참조의에서 [정의한 리소스 유형을 AWS Transfer Family](#) 참조하세요.

조건 키

관리자는 AWS JSON 정책을 사용하여 누가 무엇에 액세스할 수 있는지 지정할 수 있습니다. 즉, 어떤 보안 주체가 어떤 리소스와 어떤 조건에서 작업을 수행할 수 있는지를 지정할 수 있습니다.

Condition 요소는 정의된 기준에 따라 문이 실행되는 시기를 지정합니다. 같음(equals) 또는 미만(less than)과 같은 [조건 연산자](#)를 사용하여 정책의 조건을 요청의 값과 일치시키는 조건식을 생성할 수 있습니다. 모든 AWS 전역 조건 키를 보려면 IAM 사용 설명서의 [AWS 전역 조건 컨텍스트 키](#)를 참조하세요.

AWS Transfer Family 는 자체 조건 키 세트를 정의하고 일부 전역 조건 키 사용을 지원합니다. AWS Transfer Family 조건 키 목록을 보려면 서비스 승인 참조의에 [대한 조건 키를 AWS Transfer Family](#) 참조하세요.

예제

자격 AWS Transfer Family 증명 기반 정책의 예를 보려면 섹션을 참조하세요 [AWS Transfer Family 자격 증명 기반 정책 예제](#). VPC 엔드포인트별 IAM 정책은 섹션을 참조하세요 [Transfer Family 서버에 대한 VPC 엔드포인트 액세스 제한](#).

AWS Transfer Family 리소스 기반 정책

리소스 기반 정책은 지정된 보안 주체가 AWS Transfer Family 리소스에 대해 수행할 수 있는 작업과 조건을 지정하는 JSON 정책 문서입니다. Amazon S3은 Amazon S3 **##**에 대한 리소스 기반 권한 정책을 지원합니다. 리소스 기반 정책을 사용하여 리소스별로 다른 계정에 사용 권한을 부여할 수 있습니다. 리소스 기반 정책을 사용하여 AWS 서비스가 Amazon S3 **##**에 액세스하도록 허용할 수도 있습니다.

교차 계정 액세스를 활성화하려는 경우, 전체 계정이나 다른 계정의 IAM 엔터티를 [리소스 기반 정책의 위탁자](#)로 지정할 수 있습니다. 리소스 기반 정책에 교차 계정 위탁자를 추가하는 것은 트러스트 관계 설정의 절반밖에 되지 않는다는 것을 유념하세요. 보안 주체와 리소스가 서로 다른 AWS 계정에 있

는 경우 보안 주체 엔터티에 리소스에 액세스할 수 있는 권한도 부여해야 합니다. 엔터티에 자격 증명 기반 정책을 연결하여 권한을 부여합니다. 하지만 리소스 기반 정책이 동일 계정의 보안 주체에 액세스를 부여하는 경우 추가 보안 인증 기반 정책이 필요하지 않습니다. 자세한 내용은 AWS Identity and Access Management 사용자 가이드의 [IAM 역할과 리소스 기반 정책의 차이](#)를 참조하세요.

Amazon ECR 서비스는 리소스 기반 정책 중 한 가지 타입만 지원하는데, 이 정책 타입은 **##** 정책이라고 하며 **##**에 연결되어 있습니다. 이 정책은 객체에서 작업을 수행할 수 있는 보안 주체 엔터티(계정, 사용자, 역할 및 연동 사용자)를 정의합니다.

예제

AWS Transfer Family 리소스 기반 정책의 예를 보려면 섹션을 참조하세요 [AWS Transfer Family 태그 기반 정책 예제](#).

AWS Transfer Family 태그 기반 권한 부여

AWS Transfer Family 리소스에 태그를 연결하거나 요청에 태그를 전달할 수 있습니다 AWS Transfer Family. 태그에 근거하여 액세스를 제어하려면 `transfer:ResourceTag/key-name`, `aws:RequestTag/key-name` 또는 `aws:TagKeys` 조건 키를 사용하여 정책의 [조건 요소](#)에 태그 정보를 제공합니다. 태그를 사용하여 AWS Transfer Family 리소스에 대한 액세스를 제어하는 방법에 대한 자세한 내용은 섹션을 참조하세요 [AWS Transfer Family 태그 기반 정책 예제](#).

AWS Transfer Family IAM 역할

[IAM 역할](#)은 특정 권한이 있는 AWS 계정 내 엔터티입니다.

에서 임시 자격 증명 사용 AWS Transfer Family

임시 보안 인증을 사용하여 페더레이션을 통해 로그인하거나, IAM 역할을 맡거나, 교차 계정 역할을 맡을 수 있습니다. [AssumeRole](#) 또는 [GetFederationToken](#)과 같은 AWS STS API 작업을 호출하여 임시 보안 자격 증명을 얻습니다.

AWS Transfer Family 는 임시 자격 증명 사용을 지원합니다.

AWS Transfer Family 자격 증명 기반 정책 예제

기본적으로 IAM 사용자 및 역할은 AWS Transfer Family 리소스를 생성하거나 수정할 수 있는 권한이 없습니다. 또한 AWS Management Console AWS CLI 또는 AWS API를 사용하여 작업을 수행할 수 없습니다. IAM 관리자는 지정된 리소스에서 특정 API 작업을 수행할 수 있는 권한을 사용자와 역할에게

부여하는 IAM 정책을 생성해야 합니다. 그런 다음 관리자는 해당 권한이 필요한 IAM 사용자 또는 그룹에 이러한 정책을 연결해야 합니다.

이러한 예 JSON 정책 설명서를 사용하여 IAM 보안 인증 기반 정책을 생성하는 방법을 알아보려면 AWS Identity and Access Management 사용자 가이드의 [JSON 탭에서 정책 생성](#)을 참조하세요.

주제

- [정책 모범 사례](#)
- [AWS Transfer Family 콘솔 사용](#)
- [사용자가 자신이 권한을 볼 수 있도록 허용](#)

정책 모범 사례

자격 증명 기반 정책에 따라 계정에서 사용자가 AWS Transfer Family 리소스를 생성, 액세스 또는 삭제할 수 있는지 여부가 결정됩니다. 이 작업으로 인해 AWS 계정에 비용이 발생할 수 있습니다. ID 기반 정책을 생성하거나 편집할 때는 다음 지침과 권장 사항을 따르세요.

- AWS 관리형 정책을 시작하고 최소 권한으로 전환 - 사용자 및 워크로드에 권한 부여를 시작하려면 많은 일반적인 사용 사례에 대한 권한을 부여하는 AWS 관리형 정책을 사용합니다. 에서 사용할 수 있습니다 AWS 계정. 사용 사례에 맞는 AWS 고객 관리형 정책을 정의하여 권한을 추가로 줄이는 것이 좋습니다. 자세한 내용은 IAM 사용 설명서의 [AWS 관리형 정책](#) 또는 [AWS 직무에 대한 관리형 정책](#)을 참조하세요.
- 최소 권한 적용 - IAM 정책을 사용하여 권한을 설정하는 경우, 작업을 수행하는 데 필요한 권한만 부여합니다. 이렇게 하려면 최소 권한으로 알려진 특정 조건에서 특정 리소스에 대해 수행할 수 있는 작업을 정의합니다. IAM을 사용하여 권한을 적용하는 방법에 대한 자세한 정보는 IAM 사용 설명서에 있는 [IAM의 정책 및 권한](#)을 참조하세요.
- IAM 정책의 조건을 사용하여 액세스 추가 제한 - 정책에 조건을 추가하여 작업 및 리소스에 대한 액세스를 제한할 수 있습니다. 예를 들어, SSL을 사용하여 모든 요청을 전송해야 한다고 지정하는 정책 조건을 작성할 수 있습니다. AWS 서비스와 같은 특징을 통해 사용되는 경우 조건을 사용하여 서비스 작업에 대한 액세스 권한을 부여할 수도 있습니다 CloudFormation. 자세한 내용은 IAM 사용 설명서의 [IAM JSON 정책 요소: 조건](#)을 참조하세요.
- IAM Access Analyzer를 통해 IAM 정책을 확인하여 안전하고 기능적인 권한 보장 - IAM Access Analyzer에서는 IAM 정책 언어(JSON)와 모범 사례가 정책에서 준수되도록 새로운 및 기존 정책을 확인합니다. IAM Access Analyzer는 100개 이상의 정책 확인 항목과 실행 가능한 추천을 제공하여 안전하고 기능적인 정책을 작성하도록 돕습니다. 자세한 내용은 IAM 사용 설명서의 [IAM Access Analyzer에서 정책 검증](#)을 참조하세요.

- 다중 인증(MFA) 필요 -에서 IAM 사용자 또는 루트 사용자가 필요한 시나리오가 있는 경우 추가 보안을 위해 MFA를 AWS 계정칩니다. API 작업을 직접적으로 호출할 때 MFA가 필요하다면 정책에 MFA 조건을 추가합니다. 자세한 내용은 IAM 사용 설명서의 [MFA를 통한 보안 API 액세스](#)를 참조하세요.

IAM의 모범 사례에 대한 자세한 내용은 IAM 사용 설명서의 [IAM의 보안 모범 사례](#)를 참조하세요.

AWS Transfer Family 콘솔 사용

AWS Transfer Family 콘솔에 액세스하려면 최소 권한 집합이 있어야 합니다. 이러한 권한은 AWS 계정의 AWS Transfer Family 리소스에 대한 세부 정보를 나열하고 볼 수 있도록 허용해야 합니다. 최소 필수 권한보다 더 제한적인 보안 인증 기반 정책을 만들면 콘솔이 해당 정책에 연결된 개체(IAM 사용자 또는 역할)에 대해 의도대로 작동하지 않습니다. 자세한 내용은 AWS Identity and Access Management 사용자 가이드의 [사용자에게 권한 추가](#)를 참조하세요.

AWS CLI 또는 AWS API만 호출하는 사용자에게는 최소 콘솔 권한을 허용할 필요가 없습니다. 그 대신, 수행하려는 API 작업과 일치하는 작업에만 액세스할 수 있도록 합니다.

사용자가 자신이 권한을 볼 수 있도록 허용

이 예제는 IAM 사용자가 자신의 사용자 ID에 연결된 인라인 및 관리형 정책을 볼 수 있도록 허용하는 정책을 생성하는 방법을 보여줍니다. 이 정책에는 콘솔에서 또는 AWS CLI 또는 AWS API를 사용하여 프로그래밍 방식으로이 작업을 완료할 수 있는 권한이 포함됩니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ViewOwnUserInfo",
      "Effect": "Allow",
      "Action": [
        "iam:GetUserPolicy",
        "iam:ListGroupsWithUser",
        "iam:ListAttachedUserPolicies",
        "iam:ListUserPolicies",
        "iam:GetUser"
      ],
      "Resource": ["arn:aws:iam::*:user/${aws:username}"]
    },
    {
      "Sid": "NavigateInConsole",
      "Effect": "Allow",

```



```

    "Action": [
        "iam:GetGroupPolicy",
        "iam:GetPolicyVersion",
        "iam:GetPolicy",
        "iam:ListAttachedGroupPolicies",
        "iam:ListGroupPolicies",
        "iam:ListPolicyVersions",
        "iam:ListPolicies",
        "iam:ListUsers"
    ],
    "Resource": "*"
}
]
}

```

AWS Transfer Family 태그 기반 정책 예제

다음은 태그를 기반으로 AWS Transfer Family 리소스에 대한 액세스를 제어하는 방법의 예입니다.

태그를 사용하여 AWS Transfer Family 리소스에 대한 액세스 제어

IAM 정책의 조건은 AWS Transfer Family 리소스에 대한 권한을 지정하는 데 사용하는 구문의 일부입니다. AWS Transfer Family 리소스의 태그를 기반으로 리소스(예: 사용자, 서버, 역할 및 기타 엔터티)에 대한 액세스를 제어할 수 있습니다. 태그는 키/값 쌍입니다. 리소스 태그 지정에 대한 자세한 내용은 [AWS 리소스 태그 지정](#)을 참조하세요. AWS 일반 참조.

에서 AWS Transfer Family 리소스에는 태그가 있을 수 있으며 일부 작업에는 태그가 포함될 수 있습니다. IAM 정책을 생성하면 태그 조건 키를 사용하여 다음을 통제할 수 있습니다.

- AWS Transfer Family 리소스에 있는 태그를 기반으로 리소스에 대해 작업을 수행할 수 있는 사용자입니다.
- 어떤 태그가 작업의 요청에서 전달될 수 있는지 통제합니다.
- 요청에서 특정 키를 사용할 수 있는지 여부를 통제합니다.

태그 기반 액세스 통제를 사용하면 API 수준에서보다 더 세밀한 통제를 적용할 수 있습니다. 또한 리소스 기반 액세스 통제를 사용하는 것보다 더 동적인 통제를 적용할 수 있습니다. 요청에서 제공된 태그(요청 태그)를 기반으로 작업을 허용하거나 거부하는 IAM 정책을 생성할 수 있습니다. 작업 중인 리소스에 대한 태그(리소스 태그)를 기반으로 IAM 정책을 생성할 수 있습니다. 일반적으로 리소스 태그는 리소스에 이미 있는 태그용이고, 요청 태그는 리소스에 태그를 추가하거나 리소스에서 태그를 제거할 때 사용됩니다.

태그 조건 키의 전체 구문 및 의미는 IAM 사용자 가이드의 [AWS 리소스 태그를 사용한 액세스 통제](#)를 참조하세요. API Gateway로 IAM 정책을 지정하는 방법에 대한 자세한 설명은 API Gateway 개발자 가이드의 [IAM 권한으로 API에 대한 액세스 통제](#)를 참조하세요.

예 1: 리소스 태그 기반 작업 거부

태그를 기반으로 리소스에서 수행할 작업을 거부할 수 있습니다. 다음 예 정책은 사용자 또는 서버 리소스에 키 stage와(과) 값 prod이(가) 태그 지정된 경우 TagResource, UntagResource, StartServer, StopServer, DescribeServer, DescribeUser 작업을 거부합니다.

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

예 2: 리소스 태그 기반 작업 허용

태그를 기반으로 리소스에서 작업을 수행하도록 허용할 수 있습니다. 다음 예 정책은 사용자 또는 서버 리소스에 키 stage와(과) 값 prod이(가) 태그 지정된 경우 TagResource, UntagResource, StartServer, StopServer, DescribeServer, DescribeUser 작업을 거부합니다.

```
{
```

```

{
  "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"
        }
      }
    }
  ]
}

```

예 3: 요청 태그에 따른 사용자 또는 서버 생성 거부

다음의 정책 예에는 명령문이 2개입니다. 첫 번째 명령문은 태그의 코스트 센터 키에 값이 없는 경우 모든 리소스에 대한 CreateServer 작업을 거부합니다.

두 번째 명령문은 태그의 코스트 센터 키에 1, 2 또는 3을 제외한 다른 값이 포함된 경우 CreateServer 작업을 거부합니다.

Note

이 정책은 1, 2, 또는 3 라는 costcenter 키와 값을 포함하는 리소스를 만들거나 삭제할 수 있도록 허용합니다.

```

{
  "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"
            ]
        }
    }
}
]
}

```

AWS Transfer Family 자격 증명 및 액세스 문제 해결

다음 정보를 사용하여 및 IAM으로 작업할 때 발생할 수 있는 일반적인 문제를 진단 AWS Transfer Family 하고 수정할 수 있습니다.

주제

- [에서 작업을 수행할 권한이 없음 AWS Transfer Family](#)
- [iam:PassRole을 수행하도록 인증되지 않음](#)
- [내 AWS 계정 외부의 사람이 내 AWS Transfer Family 리소스에 액세스하도록 허용하고 싶습니다.](#)

에서 작업을 수행할 권한이 없음 AWS Transfer Family

에서 작업을 수행할 권한이 없다는 AWS Management Console 메시지가 표시되면 관리자에게 문의하여 도움을 받아야 합니다. 관리자는 로그인 보안 인증 정보를 제공한 사람입니다.

다음 예 오류는 mateojackson IAM 사용자가 콘솔을 사용하여 ##에 대한 세부 정보를 보려고 하지만 transfer:*GetWidget* 권한이 없는 경우에 발생합니다.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
transfer:GetWidget on resource: my-example-widget
```

이 경우, Mateo는 *my-example-widget* 작업을 사용하여 transfer::*GetWidget* 리소스에 액세스하도록 허용하는 정책을 업데이트하라고 관리자에게 요청합니다.

iam:PassRole을 수행하도록 인증되지 않음

iam:PassRole 작업을 수행할 수 있는 권한이 없다는 오류가 수신되면 AWS Transfer Family에 역할을 전달할 수 있도록 정책을 업데이트해야 합니다.

일부 AWS 서비스에서는 새 서비스 역할 또는 서비스 연결 역할을 생성하는 대신 기존 역할을 해당 서비스에 전달할 수 있습니다. 이렇게 하려면 역할을 서비스에 전달할 권한이 있어야 합니다.

다음 예 오류는 marymajor라는 IAM 사용자가 콘솔을 사용하여 AWS Transfer Family에서 작업을 수행하려고 하는 경우에 발생합니다. 하지만 작업을 수행하려면 서비스 역할이 부여한 권한이 서비스에 있어야 합니다. Mary는 서비스에 역할을 전달할 권한이 없습니다.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
iam:PassRole
```

이 경우, Mary가 iam:PassRole 작업을 수행할 수 있도록 Mary의 정책을 업데이트해야 합니다.

도움이 필요한 경우 AWS 관리자에게 문의하세요. 관리자는 로그인 자격 증명을 제공한 사람입니다.

다음 예 정책에는 역할을 AWS Transfer Family에 전달할 권한이 포함되어 있습니다.

123456789012를 AWS 계정 ID로 바꾸고 **MyTransferRole**를 실제 IAM 역할 이름으로 바꿉니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    { "Action": "iam:PassRole",
      "Resource": "arn:aws:iam::123456789012:role/MyTransferRole",
```

```

    "Effect": "Allow"
  }
]
}

```

내 AWS 계정 외부의 사람이 내 AWS Transfer Family 리소스에 액세스하도록 허용하고 싶습니다.

다른 계정의 사용자 또는 조직 외부의 사람이 리소스에 액세스할 때 사용할 수 있는 역할을 생성할 수 있습니다. 역할을 수임할 신뢰할 수 있는 사람을 지정할 수 있습니다. 리소스 기반 정책 또는 액세스 제어 목록(ACL)을 지원하는 서비스의 경우, 이러한 정책을 사용하여 다른 사람에게 리소스에 대한 액세스 권한을 부여할 수 있습니다.

자세한 내용은 다음을 참조하세요.

- 에서 이러한 기능을 AWS Transfer Family 지원하는지 여부를 알아보려면 섹션을 참조하세요 [AWS Transfer Family에서 IAM을 사용하는 방법](#).
- 소유 AWS 계정 한의 리소스에 대한 액세스 권한을 제공하는 방법을 알아보려면 [IAM 사용 설명서의 소유한 다른의 IAM 사용자에게 액세스 권한 제공을 참조 AWS 계정 하세요](#).
- 타사에 리소스에 대한 액세스 권한을 제공하는 방법을 알아보려면 IAM 사용 설명서의 [타사가 AWS 계정 소유한에 대한 액세스 권한 제공을 참조 AWS 계정참조하세요](#).
- ID 페더레이션을 통해 액세스 권한을 제공하는 방법을 알아보려면 IAM 사용 설명서의 [외부에서 인증된 사용자에게 액세스 권한 제공\(ID 페더레이션\)](#)을 참조하세요.
- 크로스 계정 액세스에 대한 역할과 리소스 기반 정책 사용의 차이점을 알아보려면 IAM 사용 설명서의 [IAM의 크로스 계정 리소스 액세스](#)를 참조하세요.

조직 거버넌스를 위한 IAM 조건 키

AWS Transfer Family 는 모든 IAM 정책에서 리소스 구성을 제한할 수 있는 IAM 조건 키를 제공합니다. 이러한 조건 키는 사용자 또는 역할에 연결된 자격 증명 기반 정책 또는 조직 거버넌스를 위한 서비스 제어 정책(SCPs)에 사용할 수 있습니다.

서비스 제어 정책은 전체 AWS 조직에 적용되는 IAM 정책으로, 여러 계정에 걸쳐 예방 가드레일을 제공합니다. SCPs에서 이러한 조건 키를 사용하면 조직 전체에서 보안 및 규정 준수 요구 사항을 적용하는 데 도움이 됩니다.

참고 항목

- [Transfer Family에 사용되는 작업, 리소스 및 조건 키](#)
- [서비스 제어 정책\(SCP\)](#)
- 서비스 제어 정책을 사용하여 예방 가드레일을 적용하는 방법을 설명하는 [이 동영상](#) 보기

사용 가능한 조건 키

AWS Transfer Family 는 IAM 정책에 사용할 수 있도록 다음과 같은 조건 키를 지원합니다.

`transfer:RequestServerEndpointType`

엔드포인트 유형(PUBLIC, VPC, VPC_ENDPOINT)에 따라 서버 생성 및 업데이트를 제한합니다. 일반적으로 퍼블릭 엔드포인트를 방지하는 데 사용됩니다.

`transfer:RequestServerProtocols`

지원되는 프로토콜(SFTP, FTPS, FTP, AS2)을 기반으로 서버 생성 및 업데이트를 제한합니다.

`transfer:RequestServerDomain`

도메인 유형(S3, EFS)에 따라 서버 생성을 제한합니다.

`transfer:RequestConnectorProtocol`

프로토콜(AS2, SFTP)을 기반으로 커넥터 생성을 제한합니다.

지원되는 작업

조건 키는 다음 AWS Transfer Family 작업에 적용할 수 있습니다.

- `CreateServer`: `RequestServerEndpointType`, `RequestServerProtocols` 및 `RequestServerDomain` 조건 키 지원
- `UpdateServer`: `RequestServerEndpointType` 및 `RequestServerProtocols` 조건 키 지원
- `CreateConnector`: `RequestConnectorProtocol` 조건 키 지원

SCP 정책 예제

다음 예제 SCP는 조직 전체에서 퍼블릭 AWS Transfer Family 서버 생성을 방지합니다.

```
{
```

```

"Version": "2012-10-17",
"Statement": [{
  "Sid": "DenyPublicTransferServers",
  "Effect": "Deny",
  "Action": ["transfer:CreateServer", "transfer:UpdateServer"],
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "transfer:RequestServerEndpointType": "PUBLIC"
    }
  }
}]
}

```

에 대한 규정 준수 검증 AWS Transfer Family

타사 감사자는 여러 규정 준수 프로그램의 AWS Transfer Family 일환으로의 보안 및 AWS 규정 준수를 평가합니다. 여기에는 SOC, PCI, HIPAA 등이 포함됩니다. 전체 목록은 [AWS 규정 준수 프로그램 제공 범위 내 서비스를 참조하세요](#).

특정 규정 준수 프로그램 범위의 AWS 서비스 목록은 [AWS 규정 준수 프로그램별 범위 내 서비스를 참조하세요](#). 일반 정보는 [AWS 규정 준수 프로그램](#)을 참조하세요.

를 사용하여 타사 감사 보고서를 다운로드할 수 있습니다 AWS Artifact. 자세한 내용은 [에서 보고서 다운로드를 참조하세요 AWS Artifact](#).

사용 시 규정 준수 책임은 데이터의 민감도, 회사의 규정 준수 목표 및 관련 법률과 규정에 따라 결정됩니다. AWS Transfer Family 는 규정 준수를 지원할 다음과 같은 리소스를 AWS 제공합니다.

- [보안 및 규정 준수 빠른 시작 가이드](#) -이 배포 가이드에서는 아키텍처 고려 사항에 대해 설명하고 보안 및 규정 준수 중심 기준 환경을 배포하기 위한 단계를 제공합니다 AWS.
- [HIPAA 보안 및 규정 준수를 위한 설계 백서](#) -이 백서에서는 기업이를 사용하여 HIPAA 준수 애플리케이션을 AWS 생성하는 방법을 설명합니다.
- [AWS 규정 준수 리소스](#) – 사용자의 업계와 위치에 해당할 수 있는 워크북 및 가이드 모음입니다.
- [AWS Config](#) -이 AWS 서비스는 리소스 구성이 내부 관행, 업계 지침 및 규정을 얼마나 잘 준수하는지 평가합니다.
- [AWS Security Hub CSPM](#) -이 AWS 서비스는 보안 업계 표준 및 모범 사례 준수를 확인하는 데 도움이 되는 내 보안 상태에 대한 포괄적인 보기를 제공합니다.

의 복원력 AWS Transfer Family

AWS 글로벌 인프라는 AWS 리전 및 가용 영역을 중심으로 구축됩니다. AWS 리전은 물리적으로 분리되고 격리된 여러 가용 영역을 제공하며, 이 가용 영역은 지연 시간이 짧고 처리량이 높으며 중복성이 높은 네트워킹과 연결됩니다. 가용 영역을 사용하면 중단 없이 가용 영역 간에 자동으로 장애 조치가 이루어지는 애플리케이션 및 데이터베이스를 설계하고 운영할 수 있습니다. 가용 영역은 기존의 단일 또는 복수 데이터 센터 인프라보다 가용성, 내결함성, 확장성이 뛰어납니다.

AWS Transfer Family 는 최대 3개의 가용 영역을 지원하며 연결 및 전송 요청을 위한 Auto Scaling 중복 플릿을 지원합니다.

모든 Transfer Family 엔드포인트의 경우:

- 가용 영역 수준 중복성이 서비스에 내장되어 있습니다.
- 각 AZ에는 중복 플릿이 있습니다.
- 이 중복성은 자동으로 제공됩니다.

Note

Virtual Private Cloud(VPC)의 엔드포인트의 경우 단일 서브넷을 제공할 수 있습니다. 그러나 가용 영역 중단 중 서비스 중단 위험을 줄이려면 VPC 내의 여러 가용 영역에 엔드포인트를 생성하는 것이 좋습니다.

참고 항목

- VPC에서 Transfer Family 서버를 생성하는 방법에 대한 자세한 내용은 섹션을 참조하세요 [Virtual Private Cloud\(VPC\)에 서버 생성](#).
- AWS 리전 및 가용 영역에 대한 자세한 내용은 [AWS 글로벌 인프라](#)를 참조하세요.
- 지연 시간 기반 라우팅을 사용하여 중복성을 높이고 네트워크 지연 시간을 최소화하는 방법에 대한 예는 블로그 게시물 [AWS Transfer Family 서버의 네트워크 지연 시간 최소화를 참조하세요](#).

VPC와 AWS Transfer Family APIs 간에 프라이빗 연결 생성

PrivateLink로 구동되는 인터페이스 VPC 엔드포인트를 생성하여 VPC와 AWS Transfer Family APIs 간에 프라이빗 연결을 설정할 수 있습니다. [AWS PrivateLink](#) 인터넷 게이트웨이, NAT 디바이스, VPN 연결 또는 AWS Direct Connect 연결을 사용하지 않고 VPC에 있는 것처럼 AWS Transfer Family APIs

에 액세스할 수 있습니다. VPC의 인스턴스는 AWS Transfer Family API와 통신하는 데 퍼블릭 IP 주소를 필요로 하지 않습니다.

인터페이스 엔드포인트에 대해 사용 설정하는 각 서브넷에서 엔드포인트 네트워크 인터페이스를 생성합니다. 자세한 내용은 [PrivateLink 가이드의 Access AWS services AWS PrivateLink through PrivateLink](#)를 참조하세요. AWS PrivateLink AWS Transfer Family APIs에 대한 인터페이스 VPC 엔드포인트를 설정하기 전에 AWS PrivateLink 가이드의 [고려 사항을](#) 검토하세요.

VPC 엔드포인트 정책을 사용하여 액세스 제어

기본적으로 엔드포인트를 통해 AWS Transfer Family APIs에 대한 전체 액세스가 허용됩니다. VPC 엔드포인트 정책을 사용하여 인터페이스 엔드포인트에 대한 액세스를 제어할 수 있습니다. AWS Transfer Family APIs. 이 정책은 다음 정보를 지정합니다.

- 작업을 수행할 수 있는 위탁자.
- 수행할 수 있는 작업.
- 작업을 수행할 수 있는 리소스.

자세한 내용은 Amazon VPC 사용 설명서의 [VPC 엔드포인트를 통해 서비스에 대한 액세스 제어](#)를 참조하세요.

다음은 AWS Transfer Family APIs. 엔드포인트에 연결되면 이 정책은 키 Environment 및 값으로 태그가 지정된 리소스를 제외한 모든 리소스의 모든 AWS Transfer Family APIs 작업에 대한 액세스 권한을 부여합니다 Test.

```
{
  "Statement": [{
    "Effect": "Deny",
    "Action": "transfer:StartFileTransfer",
    "Principal": "*",
    "Resource": "*",
    "Condition": {
      "StringEquals": {
        "aws:ResourceTag/Environment": "Test"
      }
    }
  }, {
    "Effect": "Allow",
    "Action": "transfer:*",
    "Principal": "*",
```

```

    "Resource": "*"
  }]
}
```

AWS Transfer Family APIs용 인터페이스 VPC 엔드포인트 생성

Amazon VPC 콘솔 또는 AWS 명령줄 인터페이스()를 사용하여 AWS Transfer Family APIs에 대한 VPC 엔드포인트를 생성할 수 있습니다AWS CLI. 자세한 내용은 AWS PrivateLink 가이드의 [VPC 엔드포인트 생성을](#) 참조하세요.

다음 서비스 이름 중 하나를 사용하여 AWS Transfer Family APIs용 VPC 엔드포인트를 생성합니다.

- `com.amazonaws.region.transfer`
- `com.amazonaws.region.transfer-fips` - FIPS(Federal Information Processing Standard) Publication 140-3 미국 정부 표준을 준수하는 인터페이스 VPC 엔드포인트를 생성합니다.

엔드포인트에 대해 프라이빗 DNS를 활성화하는 경우 리전의 기본 DNS 이름, 예를 들어를 사용하여 AWS Transfer Family APIs에 API를 요청할 수 있습니다`transfer.us-east-1.amazonaws.com`.

의 인프라 보안 AWS Transfer Family

관리형 서비스인 AWS 글로벌 네트워크 보안으로 보호 AWS Transfer Family 됩니다. AWS 보안 서비스 및가 인프라를 AWS 보호하는 방법에 대한 자세한 내용은 [AWS 클라우드 보안을](#) 참조하세요. 인프라 보안 모범 사례를 사용하여 환경을 설계하려면 보안 원칙 AWS Well-Architected Framework의 [인프라 보호를](#) 참조하세요 AWS .

AWS 에서 게시한 API 호출을 사용하여 네트워크를 AWS Transfer Family 통해 액세스합니다. 클라이언트는 다음을 지원해야 합니다.

- Transport Layer Security(TLS). TLS 1.2는 필수이며 TLS 1.3을 권장합니다.
- DHE(Ephemeral Diffie-Hellman) 또는 ECDHE(Elliptic Curve Ephemeral Diffie-Hellman)와 같은 완전 전송 보안(PFS)이 포함된 암호 제품군. Java 7 이상의 최신 시스템은 대부분 이러한 모드를 지원합니다.

NLBs 및 NATs 앞에 배치하지 마세요. AWS Transfer Family

Note

FTP 및 FTPS 프로토콜로 구성된 서버는 VPC로만 구성을 허용합니다. FTP/FTPS에 사용할 수 있는 퍼블릭 엔드포인트가 없습니다.

많은 고객이 트래픽을 AWS Transfer Family 서버로 라우팅하도록 Network Load Balancer(NLB)를 구성합니다. 일반적으로 VPC 내부와 인터넷 모두에서 서버에 액세스하거나 인터넷에서 FTP를 지원하는 방법을 AWS 제공하기 전에 서버를 생성했기 때문에이 작업을 수행합니다. 이 구성은 고객의 비용을 증가시킬 뿐만 아니라 단원에서 설명하는 다른 문제를 일으킬 수도 있습니다.

NAT 게이트웨이는 클라이언트가 기업 방화벽 뒤의 고객 프라이빗 네트워크에서 연결할 때 필수 구성 요소입니다. 그러나 많은 클라이언트가 동일한 NAT 게이트웨이 뒤에 있는 경우 성능 및 연결 제한에 영향을 미칠 수 있다는 점에 유의해야 합니다. 클라이언트에서 FTP 또는 FTPS 서버로의 통신 경로에 NLB 또는 NAT가 있는 경우는 NLB 또는 NAT의 IP 주소만 AWS Transfer Family 보기 때문에 서버가 클라이언트의 IP 주소를 정확하게 인식할 수 없습니다.

NLB 뒤에 있는 Transfer Family 서버의 구성을 사용하는 경우 NLB를 사용하는 대신 VPC 엔드포인트로 이동하고 탄력적 IP 주소를 사용하는 것이 좋습니다. NAT 게이트웨이를 사용할 때는 아래 설명된 연결 제한 사항에 유의하세요.

FTPS 프로토콜을 사용하는 경우이 구성은 서버에 액세스하는 사용자를 감사하는 기능을 줄일 뿐만 아니라 성능에도 영향을 미칠 수 있습니다.는 소스 IP 주소를 AWS Transfer Family 사용하여 데이터 영역 전체에서 연결을 샤딩합니다. FTPS의 경우 10,000개의 동시 연결이 있는 대신 통신 경로에 NLB 또는 NAT 게이트웨이가 있는 Transfer Family 서버는 300개의 동시 연결로만 제한됩니다.

AWS Transfer Family 서버 앞에서 Network Load Balancer를 사용하지 않는 것이 좋지만 FTP 또는 FTPS 구현에 클라이언트의 통신 경로에 NLB 또는 NAT가 필요한 경우 다음 권장 사항을 따르세요.

- NLB의 경우 포트 8192-8200 대신 상태 확인에 포트 21을 사용합니다.
- AWS Transfer Family 서버의 경우를 설정하여 TLS 세션 재개를 활성화합니다
다TlsSessionResumptionMode = ENFORCED.

Note

이는 향상된 보안을 제공하므로 권장되는 모드입니다.

- 클라이언트가 후속 연결에 TLS 세션 재개를 사용해야 합니다.
 - 일관된 암호화 파라미터를 보장하여 더 강력한 보안 보장을 제공합니다.
 - 잠재적 다운그레이드 공격을 방지하는 데 도움이 됩니다.
 - 성능을 최적화하면서 보안 표준 준수를 유지합니다.
- 가능하면 NLB를 사용하지 않고 마이그레이션하여 성능 및 연결 제한을 최대한 활용 AWS Transfer Family 하세요.

NLB 대안에 대한 추가 지침은 AWS Support를 통해 AWS Transfer Family 제품 관리 팀에 문의하세요. 보안 태세 개선에 대한 자세한 내용은 블로그 게시물 [AWS Transfer Family 서버 보안을 개선하기 위한 6가지 팁을 참조하세요](#).

VPC 연결 인프라 보안

VPC 송신 유형의 SFTP 커넥터는 네트워크 격리 및 프라이빗 연결을 통해 향상된 인프라 보안을 제공합니다.

네트워크 격리 이점

- 프라이빗 네트워크 트래픽: 프라이빗 SFTP 서버로의 모든 커넥터 트래픽은 퍼블릭 인터넷을 통과하지 않고 VPC 내에 유지됩니다.
- 제어된 송신: VPC를 통해 액세스하는 퍼블릭 엔드포인트의 경우 트래픽은 NAT 게이트웨이를 통해 라우팅되므로 송신 IP 주소 및 네트워크 정책을 제어할 수 있습니다.
- VPC 보안 제어: 기존 VPC 보안 그룹, 네트워크 ACLs 및 라우팅 테이블을 활용하여 커넥터 네트워크 액세스를 제어합니다.
- 하이브리드 연결: 추가 인터넷 노출 없이 설정된 VPN 또는 Direct Connect 연결을 통해 온프레미스 SFTP 서버에 액세스합니다.

Resource Gateway 보안 고려 사항

Resource Gateway는 VPC 간 리소스 액세스를 위한 보안 수신 지점을 제공합니다.

- 다중 AZ 배포: Resource Gateway에는고가용성 및 내결함성을 위해 두 개 이상의 가용 영역에 서브넷이 필요합니다.
- 보안 그룹 제어: 승인된 소스에서만 SFTP 포트(일반적으로 포트 22)에 대한 액세스를 제한하도록 보안 그룹을 구성합니다.

- 프라이빗 서브넷 배치: 프라이빗 SFTP 서버에 연결할 때 프라이빗 서브넷에 Resource Gateway를 배포하여 네트워크 격리를 유지합니다.
- 연결 제한: 각 Resource Gateway는 TCP 연결에 대해 350초 유휴 제한 시간으로 최대 350개의 동시 연결을 지원합니다.

웹 애플리케이션 방화벽 추가

AWS WAF 는 웹 애플리케이션 및 APIs으로부터 보호하는 데 도움이 되는 웹 애플리케이션 방화벽입니다. 사용자가 정의한 맞춤형 웹 보안 규칙 및 조건에 따라 웹 요청을 허용, 차단 또는 계산하는 규칙 집합(웹 액세스 통제 목록 또는 웹 ACL)을 구성할 수 있습니다. 자세한 내용은 [AWS WAF 를 사용하여 APIs 보호를 참조하세요](#).

를 추가하려면 AWS WAF

1. <https://console.aws.amazon.com/apigateway/>에서 Amazon API Gateway 콘솔을 엽니다.
2. API 탐색 창에서 맞춤 ID 제공자 템플릿을 선택합니다.
3. 단계를 선택합니다.
4. 단계(Stages) 창에서 단계 명칭을 선택합니다.
5. Stage Editor(단계 편집기) 창에서 설정 탭을 선택합니다.
6. 다음 중 하나를 수행하세요.
 - 웹 애플리케이션 방화벽(WAF)에서 이 단계와 연계할 지역 웹 ACL을 선택합니다.
 - 필요한 웹 ACL이 없는 경우 다음과 같이 웹 ACL을 생성해야 합니다:
 1. Create web ACL(웹 ACL 생성)을 선택합니다.
 2. AWS WAF 서비스 홈페이지에서 웹 ACL 생성을 선택합니다.
 3. 웹 ACL 세부 정보에서 명칭에 웹 ACL의 명칭을 입력합니다.
 4. 규칙에서 규칙 추가를 선택한 다음 나만의 규칙 및 규칙 그룹 추가를 선택합니다.
 5. 규칙 타입에서 IP 세트를 선택하여 특정 IP 주소 목록을 식별합니다.
 6. 규칙에는 규칙 명칭을 입력합니다.
 7. IP 세트의 경우 기존 IP 세트를 선택합니다. IP 집합을 생성하려면 [IP 세트 생성](#)을 참조하세요.
 8. 기원 주소로 사용할 IP 주소의 경우 헤더의 IP 주소를 선택합니다.
 9. 헤더 필드 명칭에 SourceIP를 입력합니다.
 10. 헤더 내 위치의 경우 첫 번째 IP 주소를 선택합니다.

11. 누락된 IP 주소를 위한 대처를 위해, 헤더에서 유효하지 않거나 누락된 IP 주소를 처리하려는 방법에 따라 일치 또는 불일치를 선택합니다.

12. 조치에서 IP 세트 조치를 선택합니다.

13. 어떤 규칙과도 일치하지 않는 요청에 대한 기본 웹 ACL 동작의 경우 허용 또는 차단 선택한 후 다음을 클릭합니다.

14. 4단계와 5단계에서 다음을 선택합니다.

15. 검토 및 생성에서 옵션을 검토한 다음 웹 ACL 생성을 선택합니다.

7. 변경 사항 저장(Save Changes)을 선택합니다.

8. 리소스를 선택합니다.

9. 작업 및 API 배포를 선택합니다.

AWS 웹 애플리케이션 방화벽으로 Transfer Family를 보호하는 방법에 대한 자세한 내용은 AWS 스토리지 블로그의 [AWS 애플리케이션 방화벽 및 Amazon API Gateway로 Transfer Family 보호](#)를 참조하세요.

교차 서비스 혼동된 대리자 방지

혼동된 대리자 문제는 작업을 수행할 권한이 없는 엔터티가 권한이 더 많은 엔터티에게 작업을 수행하도록 강요할 수 있는 보안 문제입니다. 에서 AWS 교차 서비스 가장은 혼동된 대리자 문제를 초래할 수 있습니다. 교차 서비스 가장은 한 서비스(직접 호출하는 서비스)가 다른 서비스(직접 호출되는 서비스)를 직접 호출할 때 발생할 수 있습니다. 호출하는 서비스는 다른 고객의 리소스에 대해 액세스 권한이 없는 방식으로 작동하게 권한을 사용하도록 조작될 수 있습니다. 이를 방지하기 위해 AWS에서는 계정의 리소스에 대한 액세스 권한이 부여된 서비스 위탁자를 사용하여 모든 서비스에 대한 데이터를 보호하는 데 도움이 되는 도구를 제공합니다. 이 문제에 대한 자세한 설명은 IAM 사용자 가이드의 [혼동되는 대리 문제](#)를 참조하세요.

리소스 정책에서 [aws:SourceArn](#) 및 [aws:SourceAccount](#) 전역 조건 컨텍스트 키를 사용하여 AWS Transfer Family가 리소스에 대해 갖는 권한을 제한하는 것이 좋습니다. 두 글로벌 조건 컨텍스트 키를 모두 사용하는 경우 `aws:SourceAccount` 값과 `aws:SourceArn` 값의 계정은 동일한 정책 명령문에서 사용할 경우 동일한 계정 ID를 사용해야 합니다.

혼동된 대리자 문제로부터 보호하는 가장 효과적인 방법은 허용하려는 리소스의 정확한 전체 Amazon 리소스 이름(ARN)을 사용하는 것입니다. 리소스의 전체 ARN을 모르거나 여러 리소스를 지정하는 경우, ARN의 알 수 없는 부분에 대해 와일드카드 문자(*)를 포함한 `aws:SourceArn` 글로벌 조건 컨텍스트 키를 사용합니다. 예를 들어 `arn:aws:transfer::region::account-id:server/*`입니다.

AWS Transfer Family는 다음과 같은 유형의 역할을 사용합니다.

- 사용자 역할 - 서비스 관리 사용자가 필요한 Transfer Family 리소스에 액세스할 수 있도록 허용합니다. AWS Transfer Family는 Transfer Family 사용자 ARN의 컨텍스트에서 이 역할을 수임합니다.
- 액세스 역할 - 전송 중인 Amazon S3 파일에만 액세스할 수 있습니다. 인바운드 AS2 전송의 경우 액세스 역할은 계약의 Amazon 리소스 이름(ARN)을 사용합니다. 아웃바운드 AS2 전송의 경우 액세스 역할은 커넥터의 ARN을 사용합니다.
- 간접 호출 역할 - Amazon API Gateway와 함께 서버의 사용자 지정 자격 증명 공급자로 사용합니다. Transfer Family는 Transfer Family 서버 ARN의 컨텍스트에서 이 역할을 담당합니다.
- 로깅 역할 - Amazon CloudWatch에 항목을 기록하는 데 사용됩니다. Transfer Family는 이 역할을 사용하여 File Transfer에 대한 정보와 함께 성공 및 실패 세부 정보를 기록합니다. Transfer Family는 Transfer Family 서버 ARN의 컨텍스트에서 이 역할을 담당합니다. 아웃바운드 AS2 전송의 경우, 로깅 역할은 커넥터 ARN을 사용합니다.
- 실행 역할 - Transfer Family 사용자가 전화를 걸어 워크플로를 시작할 수 있습니다. Transfer Family는 Transfer Family 워크플로 ARN의 컨텍스트에서 이 역할을 담당합니다.

자세한 내용은 IAM 사용자 가이드에서 [IAM의 정책 및 권한](#)을 참조하세요.

Note

다음 예에서는 자신의 정보로 각각의 **### ## ## ###**를 바꿉니다.

Note

이 예에서는 ArnLike 및 ArnEquals 모두를 사용합니다. 기능적으로 동일하므로 정책을 구성할 때 둘 중 하나를 사용할 수 있습니다. Transfer Family 설명서에서는 조건에 와일드 카드 문자가 포함된 경우에는 ArnLike를 사용하고, 정확한 일치 조건을 나타내기 위해서는 ArnEquals를 사용합니다.

AWS Transfer Family 사용자 역할 교차 서비스 혼동된 대리자 방지

다음 예 정책은 계정에 있는 모든 서버의 모든 사용자가 역할을 수임하도록 허용합니다.

```
{
```



```

"Version": "2012-10-17",
"Statement": [
  {
    "Sid": "",
    "Effect": "Allow",
    "Principal": {
      "Service": "transfer.amazonaws.com"
    },
    "Action": "sts:AssumeRole",
    "Condition": {
      "StringEquals": {
        "aws:SourceAccount": "123456789012"
      },
      "ArnLike": {
        "aws:SourceArn": "arn:aws:transfer:us-east-1:123456789012:user/*"
      }
    }
  }
]
}

```

다음 예 정책은 특정 서버의 모든 사용자가 역할을 수임하도록 허용합니다.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": "transfer.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "123456789012"
        },
        "ArnEquals": {
          "aws:SourceArn": "arn:aws:transfer:us-east-1:123456789012:user/server-id/*"
        }
      }
    }
  ]
}

```

```

    }
  }
]
}

```

다음 예 정책은 특정 서버의 특정 사용자가 역할을 수임하도록 허용합니다.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": "transfer.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "ArnLike": {
          "aws:SourceArn": "arn:aws:transfer:us-
east-1:123456789012:user/server-id/user-name"
        }
      }
    }
  ]
}

```

AWS Transfer Family 워크플로 역할 교차 서비스 혼동된 대리자 방지

다음 예 정책은 계정의 모든 워크플로가 역할을 수임하도록 허용합니다.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": "transfer.amazonaws.com"
      }
    }
  ]
}

```

```

    },
    "Action": "sts:AssumeRole",
    "Condition": {
      "StringEquals": {
        "aws:SourceAccount": "111122223333"
      },
      "ArnLike": {
        "aws:SourceArn": "arn:aws:transfer:us-west-2:111122223333:workflow/*"
      }
    }
  }
]
}

```

다음 예 정책은 특정 워크플로가 역할을 가정하도록 허용합니다.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": "transfer.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "ArnLike": {
          "aws:SourceArn": "arn:aws:transfer:us-west-2:111122223333:workflow/workflow-id"
        }
      }
    }
  ]
}

```

AWS Transfer Family 커넥터 역할 교차 서비스 혼동된 대리자 방지

다음 예제 정책은 계정의 모든 커넥터가 역할을 수임하도록 허용합니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": "transfer.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "123456789012"
        },
        "ArnLike": {
          "aws:SourceArn": "arn:aws:transfer:us-east-1:123456789012:connector/*"
        }
      }
    }
  ]
}
```

다음 예제 정책은 특정 커넥터가 역할을 수입하도록 허용합니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": "transfer.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "ArnLike": {
          "aws:SourceArn": "arn:aws:transfer:us-east-1:123456789012:connector/connector-id"
        }
      }
    }
  ]
}
```

```

    }
  ]
}

```

AWS Transfer Family 로깅 및 호출 역할 교차 서비스 혼동된 대리자 방지

Note

다음 예는 로깅 및 호출 역할 모두에 사용할 수 있습니다.

이 예제에서는 서버에 워크플로가 연결되어 있지 않은 경우 워크플로의 ARN 세부 정보를 제거할 수 있습니다.

다음 예제 로깅/호출 정책은 계정의 모든 서버(및 워크플로)가 역할을 수임하도록 허용합니다.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowAllServersWithWorkflowAttached",
      "Effect": "Allow",
      "Principal": {
        "Service": "transfer.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "111122223333"
        },
        "ArnLike": {
          "aws:SourceArn": [
            "arn:aws:transfer:us-west-2:111122223333:server/*",
            "arn:aws:transfer:us-west-2:111122223333:workflow/*"
          ]
        }
      }
    }
  ]
}

```

다음 예제 로깅/호출 정책은 특정 서버(및 워크플로)가 역할을 수임하도록 허용합니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowSpecificServerWithWorkflowAttached",
      "Effect": "Allow",
      "Principal": {
        "Service": "transfer.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "111122223333"
        },
        "ArnEquals": {
          "aws:SourceArn": [
            "arn:aws:transfer:us-west-2:111122223333:server/server-id",
            "arn:aws:transfer:us-west-2:111122223333:workflow/workflow-id"
          ]
        }
      }
    }
  ]
}
```

AWS AWS Transfer Family에 대한 관리형 정책

사용자, 그룹 및 역할에 권한을 추가하려면 직접 정책을 작성하는 것보다 AWS 관리형 정책을 사용하는 것이 더 쉽습니다. 팀에 필요한 권한만 제공하는 [고객 관리형 정책 AWS Identity and Access Management \(IAM\)을 생성하는](#) 데는 시간과 전문 지식이 필요합니다. 빠르게 시작하려면 AWS 관리형 정책을 사용할 수 있습니다. 이 정책은 일반적인 사용 사례를 다루며 사용자의 AWS 계정에서 사용할 수 있습니다. AWS 관리형 정책에 대한 자세한 정보는 [IAM 사용 설명서](#)에서 AWS 관리형 정책을 참조하세요. 모든 AWS 관리형 정책의 자세한 목록은 [AWS 관리형 정책 참조 가이드](#)를 참조하세요.

AWS 서비스는 AWS 관리형 정책을 유지 관리하고 업데이트합니다. AWS 관리형 정책에서는 권한을 변경할 수 없습니다. 서비스에서 때때로 추가 권한을 AWS 관리형 정책에 추가하여 새로운 기능을 지원합니다. 이 유형의 업데이트는 정책이 연결된 모든 ID(사용자, 그룹 및 역할)에 적용됩니다. 서비스는

새로운 기능이 시작되거나 새 작업을 사용할 수 있을 때 AWS 관리형 정책에 업데이트됩니다. 서비스는 AWS 관리형 정책에서 권한을 제거하지 않으므로 정책 업데이트로 인해 기존 권한이 손상되지 않습니다.

또한 여러 서비스에 걸쳐 있는 직무에 대한 관리형 정책을 AWS 지원합니다. 예를 들어 관리ReadOnlyAccess AWS 형 정책은 모든 AWS 서비스 및 리소스에 대한 읽기 전용 액세스를 제공합니다. 서비스가 새 기능을 시작하면 새 작업 및 리소스에 대한 읽기 전용 권한을 AWS 추가합니다. 직무 정책의 목록과 설명은 IAM 사용 설명서의 [직무에 관한AWS 관리형 정책](#)을 참조하세요.

AWS 관리형 정책: AWSTransferConsoleFullAccess

이 AWSTransferConsoleFullAccess 정책은 AWS 관리 콘솔을 통해 Transfer Family에 대한 전체 액세스 권한을 제공합니다. 자세한 내용은 [AWS Transfer Family의 서비스 연결 역할을 참조하세요](#).

AWS 관리형 정책: AWSTransferFullAccess

또한 AWSTransferFullAccess 정책은 Transfer Family 서비스에 대한 전체 액세스 권한을 제공합니다. 자세한 내용은 [AWS Transfer Family의 서비스 연결 역할을 참조하세요](#).

AWS 관리형 정책: AWSTransferLoggingAccess

이 AWSTransferLoggingAccess 정책은 AWS Transfer Family에 로그 스트림 및 그룹을 생성하고 로그 이벤트를 계정에 넣을 수 있는 모든 액세스 권한을 부여합니다. 자세한 내용은 [AWS Transfer Family의 서비스 연결 역할을 참조하세요](#).

AWS 관리형 정책: AWSTransferReadOnlyAccess

AWSTransferReadOnlyAccess 정책은 Transfer Family 서비스에 대한 읽기 전용 액세스를 제공합니다. 자세한 내용은 [AWS Transfer Family의 서비스 연결 역할을 참조하세요](#).

AWS AWS 관리형 정책에 대한 Transfer Family 업데이트

이 서비스가 이러한 변경 사항을 추적하기 시작한 이후부터 AWS Transfer Family의 AWS 관리형 정책 업데이트에 대한 세부 정보를 봅니다. 이 페이지의 변경 사항에 대한 자동 알림을 받아보려면 [에 대한 문서 기록 AWS Transfer Family](#) 페이지에서 RSS 피드를 구독하세요.

변경	설명	Date
설명서 업데이트	Transfer Family 관리 정책 각각에 대한 섹션이 추가되었습니다.	2022년 1월 27일
AWSTransferReadOnlyAccess - 기존 정책 업데이트	AWS Transfer Family는 정책을 읽을 수 있도록 허용하는 새 권한을 추가했습니다 AWS Managed Microsoft AD.	2021년 9월 30일
AWS Transfer Family에서 변경 사항 추적 시작	AWS Transfer Family가 AWS 관리형 정책에 대한 변경 사항 추적을 시작했습니다.	2021년 6월 15일

Transfer Family Terraform 모듈

[HashiCorp Terraform](#)은 HashiCorp 구성 언어(HCL)를 사용하여 개발된 오픈 소스 코드형 인프라(IaC) 엔진입니다. HashiCorp Terraform은 백엔드 인프라용 AWS Transfer Family와 함께 수백 개의 클라우드 서비스를 관리하고 클라우드 APIs를 제공합니다.

Terraform을 사용하여 관련 종속성 및 사용자 지정과 함께 AWS Transfer Family SFTP 서버 및 SFTP 커넥터를 안전하게 배포할 수 있습니다. AWS Transfer Family를 실행하는 데 필요한 리소스를 생성하기 위한 Terraform 코드가 포함된 리포지토리는 GitHub의 [Terraform Transfer Family 모듈](#) 소스 코드를 참조하세요.

Note

Terraform용 AWS Transfer Family 모듈은 커뮤니티에서 지원하는 작업입니다. 이는 AWS 서비스의 일부가 아닙니다. 최선의 지원은 AWS 스토리지 커뮤니티에서 제공합니다.

SFTP 서버

이 자동화는 사용자 지정 가능한 Terraform 모듈과 end-to-end 예제를 제공하여 SFTP 엔드포인트(PUBLIC 또는 VPC 엔드포인트 유형)를 생성하고, 로깅 및 모니터링을 위해 Amazon CloudWatch와 통합하고, 엔드포인트 액세스를 위해 사용자 ID를 관리하고, 파일이 저장된 Amazon S3 버킷에 액세스하기 위한 IAM 역할을 구성합니다. 모듈은 향상된 보안 및 키 교체 기능을 위해 사용자당 여러 SSH 퍼블릭 키(최대 50개의 키)를 지원합니다.

SFTP 커넥터

AWS Transfer Family Terraform 모듈은 이제 Amazon S3와 원격 SFTP 서버 간에 파일을 전송하는 SFTP 커넥터 배포를 지원합니다. SFTP 커넥터는 Amazon S3와 원격 SFTP 서버 간에 파일을 복사할 수 있는 완전 관리형 로우 코드 기능을 제공합니다.

이제 Terraform을 사용하여 단일 배포에서 SFTP 커넥터, 관련 종속성 및 사용자 지정을 프로그래밍 방식으로 프로비저닝할 수 있습니다. 또한 모듈은 일정 또는 이벤트 트리거를 기반으로 파일 전송 워크플로를 자동화하는 end-to-end 예제를 제공합니다. 배포에 Terraform을 사용하면 시간이 많이 걸리고 오류가 발생하기 쉬운 수동 구성이 필요하지 않으며 확장할 수 있는 빠르고 반복 가능하며 안전한 배포 옵션을 제공합니다.

AS2

AS2 Terraform 템플릿에 대한 지원을 표시하려면 [Transfer Family Terraform 템플릿 기능 요청에](#) 썸업 반응(#)을 추가합니다. 사용 사례를 설명하는 설명을 추가할 수도 있습니다.

B2B Data Interchange

AWS B2B Data Interchange 는 JSON 및 XML 데이터 형식과 주고받는 전자 데이터 교환(EDI) 문서의 변환, 검증 및 생성을 자동화합니다. B2B Data Interchange용 Terraform 템플릿에 대한 지원을 표시하려면 [기능 요청에](#) 썸업 반응(#)을 추가합니다. 사용 사례를 설명하는 설명을 추가할 수도 있습니다.

문제 해결 AWS Transfer Family

이 장에서는 사용 시 발생할 수 있는 일반적인 문제에 대한 문제 해결 정보를 제공합니다 AWS Transfer Family. 각 섹션에서는 문제에 대한 해결책을 빠르게 찾는 데 도움이 되는 특정 기능 영역에 중점을 둡니다.

- Transfer Family의 IAM 관련 문제는 [AWS Transfer Family 자격 증명 및 액세스 문제 해결](#)를 참조하세요.
- Transfer Family 웹 앱 관련 문제는 섹션을 참조하세요 [웹 앱 문제 해결](#).

주제

- [인증 문제 해결](#)
- [SFTP 커넥터 문제 해결](#)
- [SFTP 연결 및 전송 문제 해결](#)
- [사용자 지정 자격 증명 공급자 문제 해결](#)
- [워크플로 문제 해결](#)
- [EFS 문제 해결](#)
- [스토리지 및 암호화 문제 해결](#)
- [모니터링 및 알림 문제 해결](#)
- [리전 간 전송 문제 해결](#)
- [Terraform 배포 문제 해결](#)
- [웹 애플리케이션 방화벽 통합 문제 해결](#)
- [서비스 관리형 사용자 문제 해결](#)
- [AS2 문제 해결](#)

인증 문제 해결

이 섹션에서는 다음 인증 문제에 대해 가능한 해결 방법을 설명합니다.

주제

- [인증 실패—SSH/SFTP](#)
- [관리형 AD 불일치 영역 문제](#)

- [Active Directory 그룹 제한 초과](#)
- [기타 인증 문제](#)
- [Amazon API Gateway 문제 해결](#)
- [ID 제공자 테스트 문제 해결](#)
- [웹 앱에서 Amazon S3 버킷 복제](#)

인증 실패—SSH/SFTP

설명

SSH(Secure Shell) 파일 전송 프로토콜(SFTP)을 사용하여 서버에 연결하려고 하면 다음과 비슷한 메시지가 나타납니다:

```
Received disconnect from 3.130.115.105 port 22:2: Too many authentication failures
Authentication failed.
```

Note

API Gateway를 사용 중이고 이 오류가 발생하는 경우 [인증 오류가 너무 많음](#)를 참조하세요.

원인

사용자에 대한 RSA 키 쌍을 추가하지 않았으므로 대신 암호를 사용하여 인증해야 합니다.

솔루션

sftp 명령은 -o PubkeyAuthentication=no 옵션을 지정할 때 실행됩니다. 이 옵션을 선택하면 시스템에서 강제로 암호를 요청합니다. 예제:

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

관리형 AD 불일치 영역 문제

설명

사용자의 영역과 그룹 영역이 일치해야 합니다. 둘 다 기본 영역에 있거나 둘 다 신뢰할 수 있는 영역에 있어야 합니다.

원인

사용자와 그룹이 일치하지 않는 경우 Transfer Family에서 사용자를 인증할 수 없습니다. 사용자의 ID 공급자를 테스트하면 사용자 그룹에 연결된 액세스 권한을 찾을 수 없음이라는 오류 메시지가 표시됩니다.

솔루션

사용자 영역에서 그룹 영역(기본값 또는 신뢰할 수 있음)과 일치하는 그룹을 참조합니다.

Active Directory 그룹 제한 초과

설명

AWS Transfer Family 서버에 Active Directory 그룹을 더 추가하려고 하면 허용되는 최대 그룹 수에 도달했음을 나타내는 오류가 표시됩니다.

원인

AWS Transfer Family 의 기본 제한은 서버당 Active Directory 그룹 100개입니다.

솔루션

가능한 두 가지 해결 방법은 다음과 같습니다.

- Active Directory 그룹을 통합하여 필요한 총 수를 줄입니다.
- 사용 사례에 100개 이상의 그룹이 필요한 경우 사용자 지정 자격 증명 공급자를 [통한 Active Directory 인증 간소화에 설명된 대로 사용자 지정 자격 증명 공급자 AWS Transfer Family](#) 솔루션을 사용하는 것이 좋습니다.

기타 인증 문제

설명

인증 오류가 발생했지만 다른 문제 해결은 작동하지 않습니다.

원인

선행 또는 후행 슬래시(/)가 포함된 논리 디렉터리의 대상을 지정했을 수 있습니다.

솔루션

논리적 디렉터리 대상을 업데이트하여 슬래시로 시작하고 후행 슬래시를 포함하지 않도록 하세요. 예를 들어 /amzn-s3-demo-bucket/images는 허용되지만 amzn-s3-demo-bucket/images 및 /amzn-s3-demo-bucket/images/는 허용되지 않습니다.

Amazon API Gateway 문제 해결

이 섹션에서는 다음 API Gateway 문제에 대해 가능한 해결 방법을 설명합니다.

주제

- [인증 오류가 너무 많음](#)
- [연결 종료](#)

인증 오류가 너무 많음

설명

SSH(Secure Shell) 파일 전송 프로토콜(SFTP)을 사용하여 서버에 연결하려고 하면 다음 오류가 발생합니다.

```
Received disconnect from 3.15.127.197 port 22:2: Too many authentication failures
Authentication failed.
Couldn't read packet: Connection reset by peer
```

원인

잘못된 데이터베이스 사용자 암호를 입력했을 수 있습니다. 올바른 암호를 다시 입력해 보세요.

암호가 올바르면 유효하지 않은 Amazon 리소스 이름(ARN) 역할이 문제의 원인일 수 있습니다. 이것이 문제인지 확인하려면 서버의 ID 공급자를 테스트하세요. 다음과 비슷한 응답이 표시되는 경우 역할 ARN은 자리 표시자일 뿐이며 역할 ID 값이 모두 0인 것으로 표시됩니다:

```
{
  "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"
}

```

솔루션

자리 표시자 역할 ARN을 서버 액세스 권한이 있는 실제 역할로 교체하세요.

역할을 업데이트하려면

1. <https://console.aws.amazon.com/cloudformation> CloudFormation 콘솔을 엽니다.
2. 왼쪽 탐색 창에서 스택을 선택합니다.
3. 스택 목록에서 스택을 선택한 다음 파라미터 탭을 선택합니다.
4. 업데이트를 선택합니다. 업데이트 스택 페이지에서 현재 템플릿 사용을 선택하고 다음을 선택합니다.
5. UserRoleArn을 Transfer Family 서버에 액세스할 수 있는 충분한 권한이 있는 역할 ARN으로 바꾸세요.

Note

필요한 권한을 부여하려면 AmazonAPIGatewayAdministrator 및 AmazonS3FullAccess 관리형 정책을 역할에 추가할 수 있습니다.

6. 다음을 선택하고 다시 다음을 선택합니다. **##** 검토 페이지에서 이 IAM 리소스를 생성할 AWS CloudFormation 수 있음을 승인합니다를 선택한 다음 스택 업데이트를 선택합니다.

연결 종료

설명

SSH(Secure Shell) 파일 전송 프로토콜(SFTP)을 사용하여 서버에 연결하려고 하면 다음 오류가 발생합니다.

```
Connection closed
```

원인

이 문제의 한 가지 가능한 원인은 Amazon CloudWatch 로깅 역할이 Transfer Family와 신뢰 관계를 맺고 있지 않기 때문입니다.

솔루션

서버의 로깅 역할이 Transfer Family와 신뢰 관계를 맺고 있는지 확인하세요. 자세한 내용은 [신뢰 관계를 구축하기 위해](#) 단원을 참조하십시오.

ID 제공자 테스트 문제 해결

설명

콘솔 또는 TestIdentityProvider API 작업을 사용하여 자격 증명 공급자를 테스트하는 경우 Response 필드가 비어 있습니다. 예제:

```
{
  "Response": "{}",
  "StatusCode": 200,
  "Message": ""
}
```

원인

가장 가능성이 높은 원인은 잘못된 사용자 이름이나 암호로 인해 인증이 실패했기 때문입니다.

솔루션

사용자에 대한 올바른 자격 증명을 사용하고 있는지 확인하고 필요한 경우 사용자 이름 또는 암호를 업데이트하세요.

웹 앱에서 Amazon S3 버킷 복제

설명

동일한 Amazon S3 버킷이 Transfer Family 웹 앱 인터페이스에 여러 번 나타납니다.

원인

이는 사용자가 동일한 Amazon S3 버킷에 대한 권한이 있는 여러 Active Directory 그룹에 속할 때 발생합니다. 웹 앱에는 동일한 버킷 위치에 대한 중복 권한 부여를 포함하여 사용자의 UID 또는 GID와 관련된 모든 최상위 권한 부여가 나열됩니다.

솔루션

중복 목록을 방지하려면 각 사용자에게 Amazon S3 위치당 하나의 권한만 부여하도록 권한 부여를 통합합니다. Amazon S3 Access Grants 구성을 검토하고 서로 다른 Active Directory 그룹에서 동일한 버킷에 대한 중복 권한 부여를 제거합니다.

SFTP 커넥터 문제 해결

이 섹션에서는 SFTP 커넥터 관련 문제에 대해 가능한 해결 방법을 설명합니다.

주제

- [SFTP 커넥터의 신뢰할 수 있는 호스트 키 추가 문제 해결](#)
- [키 협상 실패](#)
- [SFTP 커넥터 제한](#)
- [SFTP 커넥터 성능 최적화](#)
- [VPC 연결 문제 해결](#)
- [기타 SFTP 커넥터 문제](#)

SFTP 커넥터의 신뢰할 수 있는 호스트 키 추가 문제 해결

설명

SFTP 커넥터를 만들거나 편집하고 신뢰할 수 있는 호스트 키를 추가하면 다음 오류 메시지가 나타납니다: Failed to edit connector details (Invalid host key format.)

원인

올바른 퍼블릭 키를 붙여넣는 경우 키 comment 부분을 포함했다는 문제가 발생할 수 있습니다.는 현재 키의 주석 부분을 수락하지 AWS Transfer Family 않습니다.

솔루션

키를 텍스트 필드에 붙여넣을 때 키의 설명 부분을 삭제하세요. 예를 들어, 키는 다음과 비슷할 것입니다:

```
ssh-rsa AAAA...== marymajor@dev-dsk-marymajor-1d-c1234567.us-east-1.amazon.com
```

== 문자 뒤에 오는 텍스트를 제거하고, 키에서 == 문자 앞까지 오는 부분만 붙여넣습니다.

```
ssh-rsa AAAA...==
```

키 협상 실패

설명

키 교환 협상이 실패하는 오류가 발생합니다. 예제:

```
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]
```

원인

이 오류는 서버에서 지원하는 호스트 키 알고리즘과 커넥터에서 지원하는 호스트 키 알고리즘이 겹치지 않기 때문입니다.

솔루션

원격 서버가 오류 메시지에 나열된 클라이언트 호스트 키 알고리즘 중 하나 이상을 지원하는지 확인하세요. 지원되는 키 알고리즘 목록은 [AWS Transfer Family SFTP 커넥터에 대한 보안 정책](#) 섹션을 참조하세요.

SFTP 커넥터 제한

설명

파일 전송에 SFTP 커넥터를 사용할 때 다음과 같은 오류가 발생합니다.

```
{"type":"ExecutionThrottled","details":{},"connectorId":"c-1234567890abcdef0"}
```

또는 대용량 작업 중에 파일 전송이 지연되거나 간헐적으로 실패하는 것을 확인할 수 있습니다.

원인

SFTP 커넥터에는 동시 파일 전송 및 API 작업 수를 제한하는 서비스 할당량이 있습니다. 이러한 제한을 초과하면 서비스를 보호하고 모든 고객에게 공정한 사용을 보장하기 위해 제한이 발생합니다.

솔루션

SFTP 커넥터 문제를 해결하려면 다음 솔루션을 시도하세요.

1. 애플리케이션에서 지수 백오프 및 재시도 로직을 구현합니다. 예를 들어 실패한 작업을 자동으로 재시도하고 시도 사이의 대기 시간을 늘리는 함수를 생성합니다.
2. 애플리케이션에서 속도 제한을 구현합니다.
 - 동시 전송 수를 제한합니다.
 - 전송 배치 간에 지연을 추가합니다.
3. 서비스 할당량을 기준으로 사용량을 모니터링합니다.
 - CloudWatch 지표를 사용하여 API 사용량을 추적합니다.
 - 할당량 한도에 가까워지면 알림을 받도록 경보를 설정합니다.
4. SFTP 커넥터 확장 옵션은 섹션을 참조하세요 [SFTP 커넥터 크기 조정](#).
5. 제한이 지속되고 비즈니스 운영에 영향을 미치는 경우 Service Quotas 콘솔을 통해 할당량 증가를 요청합니다.

SFTP 커넥터 성능 최적화

설명

SFTP 커넥터 전송 속도가 예상보다 느리거나 성능이 일관되지 않습니다.

원인

SFTP 커넥터 성능은 네트워크 조건, 파일 크기, 원격 서버 구성, 동시 전송 제한 등 다양한 요인의 영향을 받을 수 있습니다.

솔루션

SFTP 커넥터 성능을 최적화하려면:

- 최적의 성능을 위해 원격 SFTP 서버를 구성합니다.
 - 세션당 최대 세션 및 전송 증가
 - 지연 시간이 긴 연결을 위한 TCP 창 크기 최적화
 - 양쪽 끝에서 지원되는 경우 압축 사용
- 원격 SFTP 서버와 가까운 리전에 Transfer Family 커넥터를 배치하여 네트워크 최적화를 고려합니다.
- 모니터링 전략을 구현하여 성능 병목 현상을 식별합니다.

- 네트워크 처리량 및 지연 시간 모니터링
- 느린 전송의 패턴에 대한 로그 분석

VPC 연결 문제 해결

이 섹션에서는 VPC_LATTICE 지원 SFTP 커넥터의 일반적인 문제에 대한 해결 방법을 설명합니다.

커넥터가 PENDING 상태에서 멈춤

설명

VPC_LATTICE 지원 SFTP 커넥터는 장기간(10분 이상) PENDING 상태를 유지합니다.

원인

이는 DNS 해결 지연, Resource Gateway 구성 문제 또는 VPC Lattice 서비스 네트워크 연결 문제로 인해 발생할 수 있습니다.

솔루션

문제의 근본 원인을 분석하는 데 도움이 필요하면 AWS Support 에 문의하세요. [AWS](#) 다음 해결 방법을 시도할 수도 있습니다.

1. Resource Gateway가 ACTIVE 상태인지 확인합니다.

```
aws vpc-lattice get-resource-gateway --resource-gateway-  
identifier rgw-1234567890abcdef0
```

2. 리소스 구성이 올바르게 구성되고 활성 상태인지 확인합니다.

```
aws vpc-lattice get-resource-configuration --resource-configuration-  
identifier rcfg-1234567890abcdef0
```

3. Resource Gateway에 VPC Lattice를 지원하는 두 개 이상의 가용 영역에 서브넷이 있는지 확인합니다.
4. 문제가 지속되면 동일한 구성으로 커넥터를 삭제하고 다시 생성합니다.

ERRORED 상태의 커넥터

설명

VPC_LATTICE 지원 SFTP 커넥터는 오류 세부 정보와 함께 ERRORED 상태를 표시합니다.

원인

일반적인 원인으로는 잘못된 리소스 구성 ARN, VPC 서브넷의 IP 주소 부족 또는 리전 간 리소스 공유 시도 등이 있습니다.

솔루션

1. 를 사용하여 오류 세부 정보를 확인합니다describe-connector.

```
aws transfer describe-connector --connector-id c-1234567890abcdef0
```

2. 리소스 구성 ARN이 커넥터와 동일한 리전에 있는지 확인합니다.
3. VPC 서브넷에 Resource Gateway에 사용할 수 있는 IP 주소가 충분한지 확인합니다.
4. VPC에서 리소스 구성 대상(IP 주소 또는 DNS 이름)에 연결할 수 있는지 확인합니다.

퍼블릭 IP 주소 지원되지 않음 오류

설명

퍼블릭 IP 주소로 리소스 구성을 생성하려고 할 때 오류가 발생합니다. ValidationException: IP address x.x.x.x is not in allowed ranges.

원인

퍼블릭 엔드포인트에 대한 리소스 구성은 IP 주소가 아닌 DNS 이름을 사용해야 합니다.

솔루션

리소스 구성을 생성할 때 IP 주소 대신 SFTP 서버의 퍼블릭 DNS 이름을 사용합니다.

```
aws vpc-lattice create-resource-configuration \
  --name my-public-server-config \
  --resource-gateway-identifier rgw-1234567890abcdef0 \
  --resource-configuration-definition dnsResource={domainName="my.sftp.server.com"} \
  --port-ranges 22
```

가용 영역이 지원되지 않는 오류

설명

Resource Gateway를 생성할 때 오류가 발생합니다. Subnet subnet-xxx is not valid because it is not in a supported Availability Zone.

원인

모든 가용 영역이 VPC Lattice 교차 VPC 리소스 액세스를 지원하는 것은 아닙니다. 오류 메시지에는 해당 리전에서 지원되는 AZs 나열됩니다.

솔루션

1. 오류 메시지에 나열된 지원되는 가용 영역에 서브넷을 생성합니다.
2. 지원되는 AZs.
3. 지원되는 서로 다른 AZs에 서브넷이 두 개 이상 있는지 확인합니다.

VPC_LATTICE 커넥터를 사용한 연결 제한 시간

설명

VPC 커넥터를 통한 파일 전송이 시간 초과되거나 간헐적으로 실패합니다.

원인

VPC Lattice에는 연결 제한(리소스당 350개의 연결) 및 유희 제한 시간(TCP의 경우 350초)이 있습니다.

솔루션

1. 동시 연결을 모니터링하여 리소스당 연결 한도 350개 이내로 유지합니다.
2. 애플리케이션에서 연결 풀링 및 재사용을 구현합니다.
3. SFTP 클라이언트 애플리케이션에서 적절한 제한 시간 값을 구성합니다(350초 미만).
4. 동일한 대상에 대해 여러 리소스 구성을 생성하여 로드를 분산하는 것이 좋습니다.

기타 SFTP 커넥터 문제

설명

를 실행하면 오류가 발생StartFileTransfer하지만 문제의 원인을 알 수 없으며, API 호출 후에는 커넥터 ID만 반환됩니다.

원인

이 오류에는 여러 원인이 있을 수 있습니다. 문제를 해결하려면 커넥터를 테스트하고 CloudWatch 로그를 검색하는 것이 좋습니다.

솔루션

- 커넥터 테스트: 섹션을 참조하세요 [SFTP 커넥터를 테스트합니다.](#).. 테스트가 실패하면 시스템은 테스트 실패 이유를 기반으로 오류 메시지를 제공합니다. 이 섹션에서는 콘솔에서 또는 [TestConnection](#) API 명령을 사용하여 커넥터를 테스트하는 방법을 설명합니다.
- 커넥터에 대한 CloudWatch 로그 보기: 섹션을 참조하세요 [SFTP 커넥터의 로그 항목 예](#). 이 주제에서는 SFTP 커넥터 로그 항목에 대한 예제와 적절한 로그를 찾는 데 도움이 되는 명명 규칙을 제공합니다.

SFTP 연결 및 전송 문제 해결

이 섹션에서는 SFTP 연결 및 파일 전송 문제에 대해 가능한 해결 방법을 설명합니다.

주제

- [SFTP 연결 문제 해결](#)
- [SFTP 클라이언트 문제 해결](#)
- [파일 업로드 문제 해결](#)
- [VPC 송신 유형 SFTP 커넥터 문제 해결](#)

SFTP 연결 문제 해결

설명

SFTP 클라이언트가 연결을 시작할 수 없습니다. 이 문제는 지속적으로 또는 간헐적으로 발생할 수 있습니다. 예를 들어 SFTP 클라이언트 디버그 로그에 다음과 같은 이벤트 시퀀스가 표시될 수 있습니다.

```
sftp -vvv username@1.1.1.1
.....
debug1: Local version string .....
kex_exchange_identification: read: Connection reset by peer
Connection reset by 1.1.1.1 port 22
Connection closed.
```

원인

3방향 핸드셰이크라고도 하는 0바이트 TCP ACK(데이터가 없는 ACK)가 삭제되거나 지연되는 엣지 케이스가 있습니다.

솔루션

해결 방법으로 Transfer Family는 다른 구성을 사용하여이 문제를 해결하지만 이전 클라이언트와의 호환성 문제를 일으킬 수 있는 솔루션을 제공합니다. 따라서이 솔루션은 포트 2223에서만 사용할 수 있습니다.

VPC([Virtual Private Cloud\(VPC\)에 서버 생성](#))에서 Transfer Family 서버를 생성하는 절차에서 보안 그룹을 지정할 때 포트 2223을 사용하도록 SSH 트래픽을 구성합니다.

SFTP 클라이언트 문제 해결

SFTP 클라이언트 측 메시지는에 설명되어 있습니다[SFTP 메시지](#). SFTP 클라이언트 문제를 해결하는 가장 좋은 방법은 SFTP 클라이언트 로그를 확인하고 필요한 경우 네트워크 관리자에게 문의하는 것입니다.

파일 업로드 문제 해결

이 섹션에서는 다음 파일 업로드 문제에 대해 가능한 해결 방법을 설명합니다.

주제

- [Amazon S3 파일 업로드 오류 문제 해결](#)
- [읽을 수 없는 파일 명칭 문제 해결](#)

Amazon S3 파일 업로드 오류 문제 해결

설명

Transfer Family를 사용하여 Amazon S3 스토리지에 파일을 업로드하려고 하면 다음과 같은 오류 메시지가 나타납니다: AWS Transfer는 S3 객체에 대한 임의 액세스 쓰기를 지원하지 않음.

원인

Amazon S3를 서버 스토리지로 사용하는 경우 Transfer Family는 단일 전송에 대한 복수 연결을 지원하지 않습니다.

솔루션

Transfer Family 서버가 Amazon S3를 스토리지로 사용하는 경우 단일 전송에 복수 연결을 사용한다고 언급하는 클라이언트 소프트웨어 옵션을 비활성화하세요.

읽을 수 없는 파일 명칭 문제 해결

설명

업로드한 파일 중 일부에 손상된 파일 명칭이 있습니다. FTP 및 SFTP 전송 시 옴라우트, 악센트 부호가 있는 글자 또는 중국어나 아랍어와 같은 특정 스크립트 등 파일 명칭의 특정 문자가 제대로 표시되지 않는 문제가 발생하는 경우가 있습니다.

원인

FTP 및 SFTP 프로토콜을 사용하면 클라이언트가 파일 명칭의 문자 인코딩을 협상할 수 있지만 Amazon S3와 Amazon EFS는 그렇지 않습니다. 대신 UTF-8 문자 인코딩이 필요합니다. 따라서 특정 문자가 제대로 렌더링되지 않습니다.

솔루션

이 문제를 해결하려면 클라이언트 응용 프로그램에서 파일 명칭 문자 인코딩을 검토하고 해당 인코딩이 UTF-8로 설정되어 있는지 확인하세요.

VPC 송신 유형 SFTP 커넥터 문제 해결

VPC 송신 유형 SFTP 커넥터에 문제가 있는 경우 다음을 확인하세요.

커넥터 상태가 PENDING임

설명

VPC 송신 유형 커넥터는 생성 후 몇 분 동안 PENDING 상태로 유지되며 TestConnection은 "Connector not available"를 반환합니다.

원인

VPC 커넥터의 DNS 확인은 생성 후 완료하는 데 몇 분 정도 걸릴 수 있습니다.

솔루션

파일 전송을 시도하기 전에 커넥터 상태가 ACTIVE가 될 때까지 기다립니다. 이는 VPC 송신 유형 커넥터의 일반적인 동작입니다.

연결 시간 초과

설명

SFTP 서버에 연결을 시도할 때 VPC 송신 유형 커넥터 시간이 초과됩니다.

원인

보안 그룹은 포트 22에서 Resource Gateway 서브넷과 대상 SFTP 서버 간의 트래픽을 허용하지 않을 수 있습니다.

솔루션

보안 그룹이 Resource Gateway 서브넷과 대상 SFTP 서버 간의 포트 22 트래픽을 허용하는지 확인합니다.

리소스 구성 오류

설명

리소스 구성 문제로 인해 VPC 송신 유형 커넥터가 연결되지 않습니다.

원인

리소스 구성이 잘못된 IP 주소 또는 DNS 이름을 가리키거나 Resource Gateway가 SFTP 서버(프라이빗 엔드포인트용)와 동일한 VPC에 있지 않을 수 있습니다.

솔루션

리소스 구성이 올바른 IP 주소 또는 DNS 이름을 가리키고 Resource Gateway가 SFTP 서버(프라이빗 엔드포인트용)와 동일한 VPC에 있는지 확인합니다. 자세한 내용은 Amazon VPC Lattice 사용 설명서의 [Resource configurations](#) 섹션을 참조하시기 바랍니다.

퍼블릭 엔드포인트 문제

설명

VPC 송신 유형 커넥터는 퍼블릭 SFTP 엔드포인트에 연결할 수 없습니다.

원인

퍼블릭 엔드포인트의 경우 리소스 구성에 DNS 이름(IP 주소 아님)을 사용해야 하며 VPC에 아웃바운드 인터넷 액세스를 위한 NAT 게이트웨이가 있어야 합니다.

솔루션

리소스 구성에서 IP 주소가 아닌 DNS 이름을 사용하고 있는지 확인합니다. VPC에 아웃바운드 인터넷 액세스를 위한 NAT 게이트웨이가 있는지 확인합니다.

가용 영역 문제

설명

가용 영역 제한으로 인해 Resource Gateway를 생성할 수 없습니다.

원인

Resource Gateway에는 최소 2개의 가용 영역에 서브넷이 필요하며 모든 AZs 지원하는 것은 아닙니다.

솔루션

리전에서 VPC Lattice에 대해 지원되는 가용 영역을 확인하고 지원되는 AZs가 2개 이상 있는지 확인합니다.

사용자 지정 자격 증명 공급자 문제 해결

이 섹션에서는 Transfer Family의 사용자 지정 자격 증명 공급자와 관련된 문제에 대해 가능한 해결 방법을 설명합니다.

주제

- [API Gateway 통합 오류 문제 해결](#)
- [Lambda 함수 제한 시간 문제 해결](#)
- [일관된 Lambda 제한 시간 문제 해결](#)
- [KeyError 예외 문제 해결](#)

API Gateway 통합 오류 문제 해결

설명

사용자는 Transfer Family 서버로 인증할 수 없으며 자격 증명 공급자를 테스트할 때 다음과 같은 오류가 표시됩니다.

```
{
  "Response": "",
  "StatusCode": 500,
  "Message": "Internal server error"
}
```

원인

API Gateway 통합 오류는 다음과 같은 이유로 발생할 수 있습니다.

- 잘못된 API Gateway 구성
- Lambda 함수 오류가 제대로 처리되지 않음
- API Gateway와 Lambda 간의 권한 문제
- Lambda 함수의 잘못된 응답

솔루션

API Gateway 통합 오류를 해결하려면:

1. 자세한 오류 정보는 Lambda 함수 로그를 확인하세요.
 - CloudWatch 콘솔에서 로그 그룹 > /aws/lambda/your-function-name으로 이동합니다.
 - 근본 원인을 나타내는 오류 메시지 또는 스택 추적 찾기
2. Lambda 함수가 올바른 형식의 응답을 반환하는지 확인합니다.

```
{
  "Role": "arn:aws:iam::123456789012:role/TransferUserRole",
  "HomeDirectory": "/mybucket/home/username"
}
```

3. API Gateway에 대한 자세한 CloudWatch 로깅 활성화:
 - API Gateway 콘솔에서 API를 선택하고 스테이지를 선택합니다.
 - 스테이지를 선택하고 로그/추적에서 CloudWatch Logs를 활성화합니다.
 - 로그 수준을 ERROR 또는 INFO로 설정
4. API Gateway 엔드포인트를 직접 테스트합니다.

```
curl -X POST https://your-api-id.execute-api.region.amazonaws.com/prod/servers/your-server-id/users/username/config \
```

```
-H "Content-Type: application/json" \
-d '{"Password": "password"}'
```

5. API Gateway와 Lambda 간의 권한을 확인합니다.

- API Gateway에 Lambda 함수를 호출할 수 있는 권한이 있는지 확인합니다.
- Lambda 함수의 실행 역할에 필요한 권한이 있는지 확인합니다.

Lambda 함수 제한 시간 문제 해결

설명

사용자가 사용자 지정 자격 증명 공급자를 사용하여 Transfer Family 서버로 인증하려고 하면 지연 시간이 길어지고 인증이 실패합니다. Lambda 로그에 제한 시간 오류가 표시됩니다.

원인

사용자 지정 자격 증명 공급자에 사용되는 Lambda 함수의 기본 제한 시간은 3초입니다. 인증 로직이 이 제한 시간보다 오래 걸리는 경우(예: 외부 데이터베이스를 쿼리하거나 타사 ID 제공업체에 API를 호출하는 경우) 함수가 제한 시간을 초과하고 인증이 실패합니다.

솔루션

Lambda 제한 시간 문제를 해결하려면:

1. Lambda 함수 제한 시간을 늘립니다.
 - Lambda 콘솔에서 함수로 이동하여 구성 탭을 선택합니다.
 - 일반 구성에서 편집을 클릭합니다.
 - 제한 시간 값 증가(인증 함수의 경우 최대 15초 권장)
2. Lambda 함수 코드를 최적화합니다.
 - 데이터베이스 쿼리에 연결 풀링 사용
 - 자주 액세스하는 데이터에 대한 캐싱 구현
 - 인증 중 외부 API 호출 최소화
3. Lambda 프로비저닝된 동시성을 사용하여 콜드 스타트를 제거하는 것이 좋습니다.

```
aws lambda put-provisioned-concurrency-config \
  --function-name my-authentication-function \
  --qualifier prod \
  --provisioned-concurrent-executions 5
```

4. CloudWatch 지표를 사용하여 Lambda 성능 모니터링 및 지속 시간 임계값에 대한 경보 설정

일관된 Lambda 제한 시간 문제 해결

설명

Lambda 함수를 인증에 사용할 때 사용자에게 일관된 제한 시간이 발생합니다.

원인

Lambda는 인증에 사용되는 해당 AWS 서비스(예: DynamoDB, Secrets Manager 또는 기타 자격 증명 공급자)에 연결할 수 없습니다.

솔루션

서브넷이 AWS 서비스에 도달할 수 있는지 확인합니다. 또는 인터넷 자격 증명 공급자(예: Okta)에 연결하는 경우 Lambda 함수의 서브넷이 NAT 게이트웨이를 통해 인터넷에 연결할 수 있는지 확인합니다.

KeyError 예외 문제 해결

설명

Transfer Family 로그 항목에 'KeyError' 예외가 표시됩니다.

원인

가장 가능성이 높은 원인은 사용자 또는 identity_provider 레코드의 형식이 잘못되었거나 필수 필드가 누락되었기 때문입니다.

솔루션

로그 그룹에 ERRORS 있는 `/aws/transfer/your-server-id` 로그에서 단서를 검토합니다.

워크플로 문제 해결

이 섹션에서는 관리형 워크플로 관련 문제에 대해 가능한 해결 방법을 설명합니다.

주제

- [관리형 워크플로 문제 해결](#)
- [워크플로 복호화 문제 해결](#)

관리형 워크플로 문제 해결

이 섹션에서는 다음 워크플로 문제에 대해 가능한 해결 방법을 설명합니다.

주제

- [Amazon CloudWatch를 사용하여 워크플로 관련 오류를 해결합니다.](#)
- [워크플로 복사 오류 문제 해결](#)

Amazon CloudWatch를 사용하여 워크플로 관련 오류를 해결합니다.

설명

워크플로에 문제가 있는 경우 Amazon CloudWatch를 사용하여 원인을 조사할 수 있습니다.

원인

원인은 여러 가지가 있을 수 있습니다. Amazon CloudWatch Logs를 사용하여 조사하세요.

솔루션

Transfer Family는 워크플로 실행 상태를 CloudWatch Logs에 내보냅니다. CloudWatch Logs에는 다음과 같은 타입의 워크플로 오류가 표시될 수 있습니다:

- "type": "StepErrored"
- "type": "ExecutionErrored"
- "type": "ExecutionThrottled"
- "Service failure on starting workflow"

다양한 필터 및 패턴 구문을 사용하여 워크플로의 실행 로그를 필터링할 수 있습니다. 예를 들어, CloudWatch 로그에 로그 필터를 생성하여 ExecutionErrored 메시지가 포함된 워크플로 실행 로그를 캡처할 수 있습니다. 자세한 내용은 Amazon CloudWatch 사용자 가이드에서 [신청을 통한 로그 데이터의 실시간 처리](#) 및 [필터 및 패턴 구문](#)을 참조하세요.

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"}
```

여기서 StepErrored는 워크플로 내의 한 단계에서 오류가 발생했음을 나타냅니다. 단일 워크플로에서 여러 단계를 구성할 수 있습니다. 이 오류는 오류가 발생한 단계를 알려주고 오류 메시지를 제공합니다. 이 특정 예에서는 파일에 태그를 지정하도록 단계를 구성했지만 Amazon EFS 파일 시스템에서 파일에 태그를 지정하는 것은 지원되지 않으므로 단계에서 오류가 발생했습니다.

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"}}
```

워크플로에서 어떤 단계도 실행할 수 없는 경우 ExecutionErrored 메시지가 생성됩니다. 예를 들어, 특정 워크플로에서 단일 단계를 구성했는데 해당 단계를 실행할 수 없는 경우 전체 워크플로가 실패합니다.

Executionthrottled

시스템에서 지원할 수 있는 속도보다 빠른 속도로 워크플로가 트리거되면 실행이 제한됩니다. 이 로그 메시지는 워크플로의 실행 속도를 늦춰야 함을 나타냅니다. 워크플로 실행 속도를 스케일 다운할 수 없는 경우 [연락처 AWS](#) AWS Support 로 문의하세요.

워크플로 시작 시 서비스 실패

서버에서 워크플로를 제거하고 새 워크플로로 바꾸거나 워크플로의 실행 역할에 영향을 미치는 서버 구성을 업데이트할 때마다 새 워크플로를 실행하기 전에 약 10분을 기다려야 합니다. Transfer Family 서버는 워크플로 세부 정보를 캐시하며 서버가 캐시를 새로 고치는 데 10분이 걸립니다.

또한 활성 SFTP 세션에서 로그아웃한 다음 10분 대기 시간이 지난 후 다시 로그인해야 변경 사항을 확인할 수 있습니다.

워크플로 복사 오류 문제 해결

설명

업로드된 파일을 복사하는 단계가 포함된 워크플로를 실행하는 경우 다음 오류가 발생할 수 있습니다:

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

원인

소스 파일이 대상 버킷과 다른에 있는 Amazon S3 버킷 AWS 리전 에 있습니다.

솔루션

복사 단계가 포함된 워크플로를 실행하는 경우, 소스 버킷과 목적지 버킷이 동일한 AWS 리전안에 있는지 확인하세요.

워크플로 복호화 문제 해결

이 섹션에서는 암호화된 워크플로와 관련된 다음 문제에 대해 가능한 해결 방법을 설명합니다.

주제

- [익명 수신자 암호화 문제 해결](#)
- [서명된 암호화 파일의 오류 문제 해결](#)
- [FIPS 알고리즘의 오류 문제 해결](#)

익명 수신자 암호화 문제 해결

설명

복호화 워크플로는 암호화된 특정 파일을 처리할 때 실패하지만 다른 파일과 함께 작동합니다.

원인

수신자를 지정하지 않고 파일을 암호화했을 수 있으므로(익명 암호화) 워크플로에서 복호화에 사용할 키를 결정하기가 어렵습니다.

솔루션

-r 파라미터를 사용하여 항상 비익명 수신자로 파일을 암호화합니다. 예제:

```
gpg -e -r user@example.com --openpgp file.txt
```

파일이 특정 수신자로 암호화되었는지 익명으로 암호화되었는지 확인하려면 --list-packets 명령을 사용합니다.

```
gpg --list-packets file.txt.gpg
```

이 명령은 GPG로 암호화된 파일의 패킷 구조를 복호화하지 않고 표시합니다. 다음과 같은 수신자 정보가 포함된 출력을 찾습니다.

```
:pubkey enc packet: version 3, algo 1, keyid 1A2B3C4D5E6F7G8H
```

키 ID 정보가 표시되면 파일이 특정 수신자에 대해 암호화되었습니다. 이 정보가 누락된 경우 파일이 익명으로 암호화되어 Transfer Family 워크플로에서 복호화 실패가 발생할 수 있습니다.

이 명령을 사용하여 다음을 확인할 수도 있습니다.

- 사용된 암호화 알고리즘
- 압축 방법(있는 경우)
- 생성 타임스탬프

GPG 설치에 지원되는 암호 알고리즘 목록을 보려면 다음을 사용합니다.

```
gpg --version
```

이 정보는 특히 FIPS 규정 준수가 필요한 경우 Transfer Family 워크플로와 호환되는 암호화 알고리즘을 사용하는 데 도움이 될 수 있습니다.

서명된 암호화 파일의 오류 문제 해결

설명

복호화 워크플로가 실패하고 다음 오류가 발생합니다.

```
"Encrypted file with signed message unsupported"
```

원인

Transfer Family는 현재 암호화된 파일에 대한 서명을 지원하지 않습니다.

솔루션

PGP 클라이언트에서 암호화된 파일에 서명할 수 있는 옵션이 있는 경우 Transfer Family는 현재 암호화된 파일에 대한 서명을 지원하지 않으므로 선택을 취소해야 합니다.

FIPS 알고리즘의 오류 문제 해결

설명

복호화 워크플로가 실패하고 로그 메시지가 다음과 비슷합니다:

```
{
  "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"
  }
}
```

원인

Transfer Family 서버에는 FIPS 모드가 활성화되어 있으며 관련 복호화 워크플로 단계가 있습니다. Transfer Family 서버에 업로드하기 전에 파일을 암호화하는 경우 암호화 클라이언트는 FIPS에서 승인

하지 않은 대칭 암호화 알고리즘을 사용하는 암호화된 파일을 생성할 수 있습니다. 이러한 시나리오에서는 워크플로에서 파일을 해독할 수 없습니다. 다음 예에서 GnuPG 버전 2.4.0은 OCB (비 FIPS 블록 암호 모드)를 사용하여 파일을 암호화합니다: 이로 인해 워크플로가 실패합니다.

솔루션

파일을 암호화하는 데 사용한 GPG 키를 편집한 다음 다시 암호화해야 합니다. 다음 절차는 취해야 하는 단계를 설명합니다.

PGP 키를 편집하려면

1. `gpg --list-keys`를 실행하여 편집해야 하는 키를 식별합니다.

그러면 키 목록이 반환됩니다. 각 키에는 다음과 유사한 세부 정보가 있습니다.

```
pub   ed25519 2022-07-07 [SC]
       wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
uid           [ultimate] Mary Major <marymajor@example.com>
sub    cv25519 2022-07-07 [E]
```

2. 편집할 키를 식별합니다. 이전 단계에 표시된 예에서 ID는 `wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY`입니다.
3. `gpg --edit-key wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY`를 실행합니다.

시스템은 GnuPG 프로그램 및 지정된 키에 대한 세부 정보로 응답합니다.

4. `gpg>` 프롬프트에서 `showpref`를 입력합니다. 다음과 같은 세부 정보가 반환됩니다:

```
[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
```

참고로, 키에 저장된 선호 알고리즘이 나열되어 있습니다.

5. OCB를 제외한 모든 알고리즘을 유지하도록 키를 편집하려고 합니다. `setpref` 명령을 실행하여 유지할 모든 알고리즘을 지정합니다:

```
gpg> setpref AES256, AES192, AES, 3DES, SHA512, SHA384, SHA256, SHA224, SHA1, ZLIB,
BZIP2, ZIP, Uncompressed
```

그러면 다음 세부 사항이 반환됩니다:

```
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. y를 입력하여 업데이트한 다음, 변경을 확인하라는 메시지가 표시되면 암호를 입력합니다.
7. 변경 사항을 저장합니다.

```
gpg> save
```

복호화 워크플로를 다시 실행하기 전에 편집한 키를 사용하여 파일을 다시 암호화해야 합니다.

EFS 문제 해결

이 섹션에서는 Amazon EFS 스토리지 관련 문제에 대해 가능한 해결 방법을 설명합니다.

주제

- [Amazon SES 문제 해결](#)

Amazon SES 문제 해결

이 섹션에서는 다음 Amazon EFS 문제에 대해 가능한 해결 방법을 설명합니다.

주제

- [Amazon EFS 서비스 관리 사용자 문제 해결](#)
- [누락된 POSIX 프로파일 문제 해결](#)
- [Amazon EFS로 논리적 디렉터리 문제 해결](#)

Amazon EFS 서비스 관리 사용자 문제 해결

설명

sftp 명령을 실행하면 프롬프트가 나타나지 않고 대신 다음 메시지가 표시됩니다:

```
Couldn't canonicalize: Permission denied
Need cwd
```

원인

AWS Identity and Access Management (IAM) 사용자의 역할에 Amazon Elastic File System(Amazon EFS)에 액세스할 수 있는 권한이 없습니다.

솔루션

사용자 역할에 대한 정책 권한을 늘리세요. 와 같은 AWS 관리형 정책을 추가할 수 있습니다AmazonElasticFileSystemClientFullAccess.

누락된 POSIX 프로파일 문제 해결

설명

서버에 Amazon EFS 스토리지를 사용하고 사용자 지정 자격 증명 공급자를 사용하는 경우 AWS Lambda 함수에 POSIX 프로파일을 제공해야 합니다.

원인

한 가지 가능한 원인은 AWS Lambda지원 Amazon API Gateway 메서드를 생성하기 위해 제공하는 템플릿에 현재 POSIX 정보가 포함되어 있지 않기 때문입니다.

POSIX 정보를 제공한 경우 POSIX 정보를 제공하는 데 사용한 형식이 Transfer Family에서 올바르게 파싱되지 않을 수 있습니다.

솔루션

Transfer Family에 PosixProfile 파라미터에 대한 JSON 요소를 제공하고 있는지 확인하세요.

예를 들어, Python을 사용하는 경우 PosixProfile 파라미터를 파싱하는 위치에 다음 줄을 추가할 수 있습니다.

```
if PosixProfile:
    response_data["PosixProfile"] = json.loads(PosixProfile)
```

또는 JavaScript에서 다음 줄을 추가할 수 있습니다. 여기서 *uid-value* 및 *gid-value*는 각각 사용자 ID(UID)와 GID(그룹 ID)를 나타내는 0 이상의 정수입니다.

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

이 코드 예는 PosixProfile 파라미터를 문자열이 아닌 JSON 객체로 Transfer Family에 전송합니다.

또한 내에는 다음과 같이 PosixProfile 파라미터를 저장 AWS Secrets Manager해야 합니다. *your-uid*와(과) *your-gid*를 GID 및 UID의 실제 값으로 바꾸세요.

```
{"Uid": your-uid, "Gid": your-gid, "SecondaryGids": []}
```

Amazon EFS로 논리적 디렉터리 문제 해결

설명

사용자의 홈 디렉터리가 존재하지 않고 사용자가 `ls` 명령을 실행하면 시스템은 다음과 같이 응답합니다:

```
sftp> ls
remote readdir ("/"): No such file or directory
```

원인

Transfer Family 서버에서 Amazon EFS를 사용하는 경우, 사용자가 논리적 홈 디렉터리에서 작업할 수 있으려면 먼저 읽기 및 쓰기 권한이 있는 사용자의 홈 디렉터리를 생성해야 합니다. 사용자는 논리적 홈 디렉터리 상의 `mkdir`에 대한 권한이 없기 때문에 이 디렉터를 직접 만들 수는 없습니다.

솔루션

상위 디렉터리에 대한 관리 액세스 권한이 있는 사용자는 사용자의 논리적 홈 디렉터를 생성해야 합니다.

스토리지 및 암호화 문제 해결

이 섹션에서는 스토리지 및 암호화 관련 문제에 대해 가능한 해결 방법을 설명합니다.

주제

- [암호화된 Amazon S3 버킷에 대한 정책 문제 해결](#)

- [ResourceNotFound 예외 문제 해결](#)

암호화된 Amazon S3 버킷에 대한 정책 문제 해결

설명

Transfer Family 서버의 스토리지로 사용하고 있는 암호화된 Amazon S3 버킷이 있습니다. 서버에 파일을 업로드하려고 하면 오류 Couldn't close file: Permission denied가 발생합니다.

그리고 서버 로그를 보면 다음과 같은 오류가 표시됩니다:

```
ERROR Message="Access denied" Operation=CLOSE Path=/bucket/user/test.txt BytesIn=13
ERROR Message="Access denied"
```

원인

IAM 사용자에게 대한 정책에는 암호화된 버킷에 액세스할 권한이 없습니다.

솔루션

요구되는 AWS Key Management Service (AWS KMS) 권한을 부여하려면 정책에 추가 권한을 지정해야 합니다. 자세한 내용은 [데이터 보호 및 암호화](#)를 참조하세요.

ResourceNotFound 예외 문제 해결

설명

리소스를 찾을 수 없는 오류가 발생합니다. 예를 들어 UpdateServer를 실행하면 다음과 같은 오류가 발생할 수 있습니다:

```
An error occurred (ResourceNotFoundException) when calling the UpdateServer operation:
Unknown server
```

원인

ResourceNotFoundException 메시지를 받는 데에는 여러 가지 이유가 있습니다. 대부분의 경우 API 명령에 지정한 리소스가 존재하지 않습니다. 기존 리소스를 지정한 경우 가장 가능성이 높은 원인은 기본 지역이 리소스의 지역과 다르기 때문입니다. 예를 들어 기본 지역이 us-east-1이고 Transfer Family 서버가 us-east-2에 있는 경우 알 수 없는 리소스 예외가 발생합니다.

기본 지역 설정에 대한 자세한 설명은 [aws configure를 통한 빠른 구성](#)을 참조하세요.

솔루션

API 명령에 지역 파라미터를 추가하여 특정 리소스를 찾을 위치를 명시적으로 지정합니다.

```
aws transfer -describe-server --server-id server-id --region us-east-2
```

모니터링 및 알림 문제 해결

이 섹션에서는 누락되거나 불완전한 CloudWatch 지표 및 누락된 EventBridge 이벤트를 AWS Transfer Family 포함하여 모니터링 및 알림 문제 해결에 대한 정보를 제공합니다.

주제

- [누락되거나 불완전한 CloudWatch 지표 문제 해결](#)
- [누락된 EventBridge 이벤트 문제 해결](#)

누락되거나 불완전한 CloudWatch 지표 문제 해결

설명

Transfer Family 서버의 CloudWatch 지표가 누락되었거나 불완전하거나 예상대로 업데이트되지 않았습니다.

원인

누락되거나 불완전한 지표는 다음과 같은 여러 요인으로 인해 발생할 수 있습니다.

- 구성 문제 로깅
- 예상 기간 내에 지표를 생성하지 않는 낮은 활동 수준
- 잘못된 차원 또는 시간 범위의 지표 보기

솔루션

CloudWatch 지표가 누락되거나 불완전한 문제를 해결하려면:

1. Transfer Family 서버에 로깅이 올바르게 구성되어 있는지 확인합니다.

- Transfer Family 콘솔의 서버 세부 정보 > 추가 세부 정보 > 로깅 역할에서 로깅이 활성화되어 있는지 확인합니다.

- 로깅 역할에는 필요한 권한과 신뢰 관계가 있습니다.
2. CloudWatch 콘솔에서 지표를 볼 때:
 - 서버 수준 지표에 ServerId와 같은 올바른 차원 사용
 - 활동 기간을 포함하도록 시간 범위 조정
 - 올바른에 있는지 확인 AWS 리전
 3. Transfer Family 서버에서 테스트 활동을 생성하여 지표가 생성되고 있는지 확인합니다.

누락된 EventBridge 이벤트 문제 해결

설명

Transfer Family 이벤트를 캡처하도록 Amazon EventBridge 규칙을 구성했지만 이벤트가 대상 대상으로 전달되지 않거나 예상 작업을 트리거하고 있습니다.

원인

EventBridge 이벤트 누락의 원인은 다음과 같습니다.

- 잘못 구성된 이벤트 패턴
- 이벤트 대상의 권한 문제
- 서비스 제한 또는 제한
- 서버 구성으로 인해 이벤트가 생성되지 않음

솔루션

누락된 EventBridge 이벤트 문제를 해결하려면:

1. Transfer Family 이벤트와 일치하도록 이벤트 패턴의 형식이 올바른지 확인합니다.

```
{
  "source": ["aws.transfer"],
  "detail-type": ["Transfer State Change"],
  "detail": {
    "serverId": ["s-1234567890abcdef0"]
  }
}
```

2. 이벤트 대상에 필요한 권한이 있는지 확인합니다.

- Lambda 대상의 경우 Lambda 함수의 리소스 정책에서 EventBridge가 호출하도록 허용하는지 확인합니다.
- SQS 대상의 경우 대기열 정책에서 EventBridge가 메시지를 전송하도록 허용하는지 확인합니다.
- SNS 대상의 경우 주제 정책에서 EventBridge가 게시하도록 허용하는지 확인합니다.

3. 샘플 이벤트를 생성하여 규칙을 테스트합니다.

- EventBridge 콘솔을 사용하여 패턴과 일치하는 테스트 이벤트 생성
- Transfer Family 서버에서 이벤트를 생성해야 하는 작업 수행

4. EventBridge 규칙 지표를 활성화하여 규칙 호출 및 실패를 모니터링합니다.

```
aws events put-rule --name "TransferStateChangeRule" --event-pattern '{...}' --state
ENABLED --metrics-enabled
```

5. CloudWatch Logs에서 이벤트 전송 실패와 관련된 오류 메시지를 확인합니다.

리전 간 전송 문제 해결

이 섹션에서는에서 파일 전송과 관련된 문제에 대해 가능한 해결 방법을 설명합니다 AWS 리전.

주제

- [리전 간 전송 권한 문제 해결](#)
- [리전 간 전송 성능 문제 해결](#)

리전 간 전송 권한 문제 해결

설명

Transfer Family 워크플로를 사용하여 서로 다른 리전의 Amazon S3 버킷 간에 파일을 전송하려고 하면 다음과 같은 오류가 발생합니다.

```
{
  "type": "StepErrored",
  "details": {
    "errorType": "BAD_REQUEST",
    "errorMessage": "Access Denied (Service: Amazon S3; Status Code: 403; Error Code:
AccessDenied)",
    "stepType": "COPY",
```

```

    "stepName": "cross_region_copy"
  }
}

```

원인

리전 간 전송에는 소스 버킷과 대상 버킷 모두에 대한 특정 IAM 권한이 필요합니다. Transfer Family 서버 또는 워크플로에서 사용하는 IAM 역할에 다른 리전의 버킷에 액세스할 수 있는 충분한 권한이 없을 수 있습니다.

솔루션

리전 간 전송 권한 문제를 해결하려면:

1. IAM 역할에 소스 버킷과 대상 버킷 모두에 대한 권한이 있는지 확인합니다.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetObject",
        "s3:GetObjectVersion"
      ],
      "Resource": "arn:aws:s3:::source-bucket-name/*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "s3:PutObject",
        "s3:PutObjectAcl"
      ],
      "Resource": "arn:aws:s3:::destination-bucket-name/*"
    }
  ]
}

```

2. KMS 암호화를 사용하는 경우 소스 및 대상 KMS 키 모두에 대한 권한을 추가합니다.

```

{
  "Effect": "Allow",
  "Action": [

```

```

    "kms:Decrypt",
    "kms:GenerateDataKey"
  ],
  "Resource": [
    "arn:aws:kms:source-region:account-id:key/source-key-id",
    "arn:aws:kms:destination-region:account-id:key/destination-key-id"
  ]
}

```

3. 두 리전의 버킷 정책이 Transfer Family 서버의 IAM 역할에서 액세스를 허용하는지 확인합니다.
4. 교차 계정 전송의 경우 적절한 교차 계정 권한이 구성되어 있는지 확인합니다.

리전 간 전송 성능 문제 해결

설명

리전 간 전송은 예상보다 크게 느리거나 대용량 파일 전송 중에 시간 초과됩니다.

원인

리전 간 전송은 기본적으로 지연 시간이 길어지며 네트워크 조건, 파일 크기 및 서비스 제한의 영향을 받을 수 있습니다. 대용량 파일이나 대용량의 작은 파일은 성능이 저하될 수 있습니다.

솔루션

리전 간 전송 성능을 개선하려면:

- 대용량 파일의 경우 Amazon S3 Transfer Acceleration을 사용하는 것이 좋습니다.

```

aws s3 cp --source-region us-east-1 --region us-west-2 \
  s3://source-bucket/large-file.zip s3://destination-bucket/large-file.zip \
  --acl bucket-owner-full-control --s3-accelerate

```

- 여러 개의 작은 파일의 경우 전송하기 전에 일괄 처리합니다.
 - 압축을 사용하여 여러 파일 결합
 - 대규모 전송에 Amazon S3 배치 작업 사용
- 대규모 전송에 적절한 제한 시간 설정과 함께 Transfer Family SFTP 커넥터 사용 고려
- 반복 전송의 경우 임시 전송 대신 Amazon S3 교차 리전 복제(CRR)를 사용하여 데이터를 복제하는 것이 좋습니다.
- Amazon CloudWatch 지표를 사용하여 전송 성능을 모니터링하여 병목 현상 식별

Terraform 배포 문제 해결

이 섹션에서는 Terraform을 사용하여 Transfer Family 리소스를 배포하는 것과 관련된 문제에 대해 가능한 해결 방법을 설명합니다. Transfer Family용 Terraform 모듈에 대한 일반 정보는 섹션을 참조하세요 [Transfer Family Terraform 모듈](#).

주제

- [Terraform 리소스 생성 실패 문제 해결](#)
- [Terraform 상태 관리 문제 해결](#)

Terraform 리소스 생성 실패 문제 해결

설명

Terraform을 사용하여 Transfer Family 리소스를 생성하려고 하면 다음과 같은 오류가 발생합니다.

```
Error: error creating Transfer Server: InvalidRequestException: The request is not valid.
Error: error creating Transfer User: InvalidRequestException: Unable to create the user because the server endpoint type is incompatible with the home directory type.
```

원인

이러한 오류는 일반적으로 호환되지 않는 구성 파라미터 또는 Terraform 구성의 종속성 누락으로 인해 발생합니다. 일반적인 사용 사례는 다음과 같습니다.

- 호환되지 않는 엔드포인트 유형 및 스토리지 구성
- 필수 IAM 역할 또는 정책 누락
- 잘못된 보안 정책 사양
- VPC 엔드포인트 구성 문제

솔루션

Terraform 배포 문제를 해결하려면:

- Terraform 구성이 호환되는 파라미터 조합을 사용하는지 확인합니다.
 - 퍼블릭 엔드포인트의 경우 스토리지에 Amazon S3를 사용하고 있는지 확인합니다.

- VPC 엔드포인트의 경우 적절한 VPC 및 보안 그룹 구성을 확인합니다.
- depends_on 속성과 함께 명시적 종속성을 사용하여 리소스가 올바른 순서로 생성되도록 합니다.
- 모든 IAM 역할에 필요한 신뢰 관계 및 권한이 있는지 확인합니다.
- 최신 버전의 Terraform용 AWS 공급자를 사용하여 모든 Transfer Family 기능과의 호환성을 보장합니다.
- 복잡한 배포 및 간단한 사용 사례의 경우 <https://github.com/aws-ia/terraform-aws-transfer-family> GitHub에서 제공되는 공식 Transfer Family Terraform 모듈을 사용하는 것이 좋습니다. 이 모듈은 간단하고 복잡한 고객 사용 사례를 다루는 광범위한 예제를 제공하고, AWS 모범 사례를 따르며, 코드 형 인프라(IaC) 구성에 도움이 필요한 고객을 위해 배포를 간소화할 수 있습니다.

Terraform 상태 관리 문제 해결

설명

Terraform 외부에서 Transfer Family 리소스를 변경한 후(콘솔 또는를 통해 AWS CLI) terraform plan 또는를 실행할 때 상태 드리프트 또는 오류가 발생합니다terraform apply.

원인

Terraform은 관리하는 리소스를 추적하는 상태 파일을 유지합니다. Terraform 외부에서 변경하면 상태 파일이 실제 리소스와 동기화되지 않아 후속 Terraform 작업 중에 오류 또는 예기치 않은 동작이 발생합니다.

솔루션

Transfer Family 리소스와 관련된 Terraform 상태 관리 문제를 해결하려면

1. terraform import를 사용하여 기존 리소스를 Terraform 관리로 가져옵니다.

```
terraform import <transfer_family_server.example> s-<server-id>
terraform import <transfer_family_server.example> s-<server-id>/username
```

2. terraform refresh를 사용하여 상태 파일을 현재 실제 인프라로 업데이트합니다.
3. 가져올 수 없거나 복잡한 상태 문제가 있는 리소스의 경우 terraform state rm를 사용하여 상태 파일에서 리소스를 제거한 다음 Terraform으로 다시 생성하는 것이 좋습니다.
4. 미래 상태 드리프트를 방지하기 위해 Terraform을 통해서만 Transfer Family 리소스를 관리하는 정책을 구현합니다.

5. 잠금과 함께 원격 상태 스토리지를 사용하여 팀에서 작업할 때 동시 수정 방지

웹 애플리케이션 방화벽 통합 문제 해결

이 섹션에서는 Transfer Family AWS WAF 와의 통합과 관련된 문제에 대해 가능한 해결 방법을 설명합니다.

주제

- [WAF 차단 합법적인 트래픽 문제 해결](#)
- [사용자 지정 자격 증명 공급자와의 WAF 통합 문제 해결](#)

WAF 차단 합법적인 트래픽 문제 해결

설명

Transfer Family 엔드포인트 AWS WAF 로를 구성한 후 합법적인 사용자는 연결할 수 없거나 간헐적인 연결 실패가 발생합니다. 로그에 HTTP 403(금지됨) 응답이 표시될 수 있습니다.

원인

AWS WAF 규칙이 너무 제한적이거나 잘못 구성되어 합법적인 트래픽을 차단하는 오탐지가 발생할 수 있습니다. 일반적인 사용 사례는 다음과 같습니다.

- 실수로 회사 네트워크 또는 VPNs 차단하는 IP 기반 규칙
- 임계값이 정상 트래픽 패턴에 비해 너무 낮은 속도 기반 규칙
- 사용 사례에 지나치게 공격적인 관리형 규칙 그룹

솔루션

거짓 긍정 문제를 해결하려면:

1. AWS WAF 로깅을 활성화하여 블록을 트리거하는 규칙을 식별합니다. 지침은 [AWS WAF 웹 ACL 트래픽 로깅을 참조하세요](#).
2. 로그를 검토하여 차단된 요청의 패턴을 식별합니다.
3. 다음과 같이 규칙을 조정합니다.
 - 허용 목록에 IP 주소 또는 범위 추가

- 속도 기반 규칙의 속도 제한 증가
- 차단 없이 모니터링하도록 차단 모드 대신 특정 규칙을 개수 모드로 설정
- 규칙 그룹 제외를 사용하여 특정 규칙에 대한 예외 생성

4. 완전히 배포하기 전에 합법적인 트래픽의 대표 샘플을 사용하여 업데이트된 구성을 테스트합니다.

사용자 지정 자격 증명 공급자와의 WAF 통합 문제 해결

설명

사용자 지정 자격 증명 공급자를 사용하는 Transfer Family 서버 AWS WAF 로를 구성한 후 인증이 실패하거나 사용자에게 간헐적인 인증 문제가 발생합니다.

원인

API Gateway에서 사용자 지정 자격 증명 공급자를 사용하는 경우 AWS WAF 규칙은 Transfer Family와 자격 증명 공급자 간의 API 호출을 방해할 수 있습니다. 이는 규칙 세트를 기반으로 API 트래픽을 검사하고 잠재적으로 차단하기 때문에 AWS WAF 발생할 수 있습니다.

솔루션

AWS WAF 및 사용자 지정 자격 증명 공급자 관련 문제를 해결하려면:

- AWS WAF 구성에 사용자 지정 자격 증명 공급자가 사용하는 API Gateway 엔드포인트에 대한 예외가 포함되어 있는지 확인합니다.
- Transfer Family 서비스 보안 주체(`transfer.amazonaws.com`)를 AWS WAF 규칙의 허용 목록에 추가합니다.
- 관리형 규칙 그룹을 사용하는 경우 API 인증 흐름에 영향을 미칠 수 있는 규칙을 검토하고 이러한 특정 규칙을 비활성화하는 것이 좋습니다.
- `TestIdentityProvider` API 작업을 사용하여 ID 제공업체를 직접 테스트하여 AWS WAF 간섭 없이 올바르게 작동하는지 확인합니다.

서비스 관리형 사용자 문제 해결

이 섹션에서는 서비스 관리형 사용자의 문제에 대해 가능한 해결 방법을 설명합니다.

주제

- [서비스 관리 사용자 문제 해결](#)

서비스 관리 사용자 문제 해결

이 섹션에서는 다음 문제에 대한 가능한 해결 방법을 설명합니다.

주제

- [퍼블릭 키 본문이 너무 긴 문제 해결](#)
- [문제 해결에서 SSH 퍼블릭 키를 추가하지 못했습니다.](#)

퍼블릭 키 본문이 너무 긴 문제 해결

설명

서비스 관리 사용자를 생성하려고 하면 다음 오류가 발생합니다:

```
Failed to create user (1 validation error detected:  
'sshPublicKeyBody' failed to satisfy constraint: Member must have length less than or  
equal to 2048)
```

원인

퍼블릭 키 본문에 대한 PGP 키를 입력 중일 수 있으며 서비스 관리형 사용자에게 대한 PGP 키 AWS Transfer Family 는 지원하지 않습니다.

솔루션

PGP 키가 RSA 기반인 경우, 귀하는 그것을 PEM 형식으로 변환할 수 있습니다. 예를 들어 Ubuntu는 <https://manpages.ubuntu.com/manpages/jammy/en/man1/openpgp2ssh.1.html> 변환 도구를 제공합니다.

문제 해결에서 SSH 퍼블릭 키를 추가하지 못했습니다.

설명

서비스 관리 사용자에게 대한 퍼블릭 키를 추가하려고 하면 다음 오류가 발생합니다:

```
Failed to add SSH public key (Unsupported or invalid SSH public key format)
```

원인

SSH2-formatted의 퍼블릭 키를 가져오려고 할 수 있으며 서비스 관리 사용자를 위한 SSH2-formatted 퍼블릭 키 AWS Transfer Family 는 지원하지 않습니다.

솔루션

키를 OpenSSH 형식으로 변환해야 합니다. 이 과정은 [SSH2 키를 SSH 퍼블릭 키 형식으로 변환](#)에 설명되어 있습니다.

AS2 문제 해결

이 섹션에서는 AS2 전송 관련 문제에 대해 가능한 해결 방법을 설명합니다.

주제

- [AS2 문제 해결](#)
- [AS2 인증서 문제](#)
- [AS2 MDN 수신 문제](#)
- [인증서 만료 모니터링 문제](#)

AS2 문제 해결

[AS2 오류 코드](#) 적용성 설명 2(AS2) 지원 서버에 대한 , 메시지 및 문제 해결 팁은이 가이드의 AS2 장애 코드 섹션에 설명되어 있습니다.

AS2 인증서 문제

설명

AS2 전송에 인증서 관련 오류가 발생했습니다.

원인

일반적인 원인에는 만료된 인증서, 잘못된 인증서 형식 또는 일치하지 않는 인증서 체인이 포함됩니다.

솔루션

다음 솔루션을 사용해 보세요.

- 인증서가 만료되지 않았는지 확인
- 인증서가 올바른 형식인지 확인(용 PEM AWS Transfer Family)

- 인증서 체인이 완전하고 유효한지 확인
- 서명 및 암호화 인증서가 거래 파트너 간에 일치하는지 확인
- 중단 방지를 위해 만료 전 인증서 교체

AS2 MDN 수신 문제

설명

AS2 전송에 대해 예상되는 메시지 처리 알림(MDNs 수신되지 않습니다).

원인

네트워크 연결 문제, 잘못된 엔드포인트 구성 또는 MDN 형식 불일치로 인해 MDN 문제가 발생할 수 있습니다.

솔루션

다음 솔루션을 고려하세요.

- MDN URL이 올바르게 구성되고 액세스 가능한지 확인
- AS2 서버와 MDN 엔드포인트 간의 네트워크 연결 확인
- 두 거래 파트너가 동일한 MDN 유형(동기식 또는 비동기식)으로 구성되어 있는지 확인합니다.
- MDN 처리와 관련된 오류가 있는지 AS2 로그 검토
- 동기 MDNs 사용하는 경우 제한 시간이 적절하게 설정되었는지 확인합니다.

인증서 만료 모니터링 문제

이 섹션에서는 인증서 만료 모니터링 및 DaysUntilExpiry 지표와 관련된 일반적인 문제에 대한 솔루션을 제공합니다.

DaysUntilExpiry 지표가 표시되지 않음

문제: 인증서를 가져온 후 Amazon CloudWatch에 DaysUntilExpiry 지표가 표시되지 않습니다.

해결 방법:

- 인증서를 가져온 후 최대 24시간 동안 기다립니다. Transfer Family가 계정에 지표를 내보내는 데 최대 하루가 걸릴 수 있습니다.

- CloudWatch의 AWS/Transfer 네임스페이스에서 올바른 AWS 리전을 살펴보고 있는지 확인합니다.

인증서 만료 경보가 트리거되지 않음

문제: 인증서 만료에 대한 CloudWatch 경보가 예상대로 트리거되지 않습니다.

해결 방법:

- 경보가 통계와 Maximum 1일의 기간으로 구성되어 있는지 확인합니다.
- 임계값 비교가 원하는 일수Less than or equal to로 설정되어 있는지 확인합니다.
- 경보 구성에서가 선택Treat missing data as good (not breaching)되어 있는지 확인합니다.
- 경보 차원이 인증서의 CertificateId 및 설명(제공된 경우)과 일치하는지 확인합니다.
- 경보 작업(SNS 주제, 이메일 알림)이 제대로 구성되고 활성화 상태인지 확인합니다.

AWS Transfer Family API 참조

Transfer Family에 대한 전체 API 참조 가이드는 [AWS Transfer Family API 참조](#)에서 확인할 수 있습니다.

AWS Transfer Family 는 다음 프로토콜을 통해 Amazon Simple Storage Service(Amazon S3) 스토리지 안팎으로 파일을 전송하는 데 사용할 수 있는 보안 전송 서비스입니다.

- Secure Shell (SSH) File Transfer 프로토콜 (SFTP)
- File Transfer 프로토콜 보안(FTPS)
- File Transfer 프로토콜(FTP)
- 적응성 보고서 2(AS2)

서버, 사용자, 역할은 모두 Amazon 리소스 이름(ARN)으로 식별합니다. ARN을 사용하여 개체에 태그, 키 값 쌍을 할당할 수 있습니다. 태그는 이러한 개체(entity)를 그룹화하고 검색하는 데 사용할 수 있는 메타데이터입니다. 태그를 유용하게 사용할 수 있는 대표적인 분야는 회계입니다.

다음 규칙은 AWS Transfer Family ID 형식으로 관찰됩니다.

- s-01234567890abcdef 양식에서 얻은 ServerId 값
- key-01234567890abcdef 양식에서 얻은 SshPublicKeyId 값

Amazon 리소스 이름(ARN) 양식은 다음과 같은 형식을 취합니다.

- 사용자의 경우, ARN은 `arn:aws:transfer:region:account-id:server/server-id` 형식을 취합니다.

서버 ARN의 예: `arn:aws:transfer:us-east-1:123456789012:server/s-01234567890abcdef`

- 사용자의 경우, ARN은 `arn:aws:transfer:region:account-id:user/server-id/username` 형식을 취합니다.

예를 들면, `arn:aws:transfer:us-east-1:123456789012:user/s-01234567890abcdef/user1`입니다.

사용 중인 DNS 항목(엔드포인트)은 다음과 같습니다.

- `transfer.region.amazonaws.com` 형식을 취하는 API 엔드포인트
- `server-id.server.transfer.region.amazonaws.com` 형식을 취하는 서버 엔드포인트

에 대한이 API 인터페이스 참조에는 관리하는 데 사용할 수 있는 프로그래밍 인터페이스에 대한 설명서가 AWS Transfer Family 포함되어 있습니다 AWS Transfer Family. 참조 구조는 다음과 같습니다.

- 작업의 영문자순 목록은 [Actions](#)를 참조하세요.
- 데이터 형식의 알파벳순 목록은 [Types](#)을 참조하세요.
- 공통 쿼리 파라미터 목록은 [공통 파라미터](#)를 참조하세요.
- 오류 코드에 대한 설명은 [공통 오류](#)를 참조하세요.

Tip

명령을 실제로 실행하는 대신 모든 API 직접 호출에 `--generate-cli-skeleton` 파라미터를 사용하여 파라미터 템플릿을 생성하고 표시할 수 있습니다. 그런 다음 생성된 템플릿을 사용하여 사용자 정의하고 이후 명령의 입력으로 사용할 수 있습니다. 자세한 내용은 [파라미터 스켈레톤 파일 생성 및 사용](#)을 참조하세요.

API 요청 만들기

콘솔을 사용하는 것 외에도 AWS Transfer Family API를 사용하여 서버를 프로그래밍 방식으로 구성하고 관리할 수 있습니다. 이 섹션에서는 AWS Transfer Family 작업, 인증을 위한 요청 서명 및 오류 처리에 대해 설명합니다. Transfer Family에서 사용할 수 있는 지역 및 엔드포인트에 대한 자세한 설명은 AWS 일반 참조의 [AWS Transfer Family 엔드포인트 및 할당량](#)을 참조하세요.

Note

Transfer Family를 사용하여 애플리케이션을 개발할 때 AWS SDKs를 사용할 수도 있습니다. Java, .NET 및 PHP용 AWS SDK는 프로그래밍 작업을 간소화하며 기본 Transfer Family API를 포함합니다. SDK 라이브러리 다운로드에 대한 정보는 [샘플 코드 라이브러리](#) 섹션을 참조하세요.

Transfer Family 필수 요청 헤더

이 섹션에서는 AWS Transfer Family에 대한 모든 POST 요청과 함께 전송해야 하는 필수 헤더에 대해 설명합니다. 호출하려는 작업을 포함하는 요청에 대한 핵심 정보, 요청 날짜 및 요청 전송자의 권한을 부여함을 나타내는 정보를 식별할 HTTP 헤더를 포함해야 합니다. 헤더는 대소문자를 구별하고 헤더의 순서는 중요하지 않습니다.

다음은 [ListServers](#) 작업에서 사용하는 헤더의 예입니다.

```
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}
```

다음은 Transfer Family에 대한 POST 요청과 함께 포함해야 하는 헤더입니다. 아래에 표시된 "x-amz"로 시작하는 헤더는 고유합니다 AWS. 나머지 헤더는 HTTP 트랜잭션에 사용되는 공통 헤더입니다.

Transfer Family 요청 입력 및 서명

모든 요청 입력은 요청 본문의 JSON 페이로드의 일부로 전송되어야 합니다. 모든 요청 필드가 옵션(예: ListServers)인 작업의 경우 요청 본문에 {}와 같이 빈 JSON 객체를 제공해야 합니다. Transfer Family 페이로드 요청/응답의 구조는 기존 API 참조(예: [DescribeServer](#))에 문서화되어 있습니다.

Transfer Family는 AWS 서명 버전 4를 사용한 인증을 지원합니다. 자세한 내용은 [AWS API 요청 서명](#)을 참조하세요.

오류 응답

오류가 있는 경우, 응답 헤더 정보에는 다음 내용이 포함됩니다.

- 콘텐츠 타입: application/x-amz-json-1.1
- 적절한 4xx 또는 5xx HTTP 상태 코드

오류 응답의 본문에는 발생한 오류에 대한 정보가 포함됩니다. 다음 샘플 오류 응답은 모든 오류 응답에 공통된 응답 요소의 출력 구문을 나타냅니다.

```
{
  "__type": "String",
  "Message": "String", <!-- Message is lowercase in some instances -->
  "Resource": "String",
  "ResourceType": "String",
  "RetryAfterSeconds": "String"
}
```

다음 표는 이전 구문에 표시된 JSON 오류 응답 필드를 설명합니다.

__타입

Transfer Family API 호출의 예외 중 하나입니다.

타입: 문자열

메시지 또는 메시지

작업 오류 코드 메시지 중 하나입니다.

Note

일부 예외에서 message가 사용되고 다른 예외에서 Message가 사용됩니다. 인터페이스의 코드를 확인하여 적절한 대/소문자를 확인할 수 있습니다. 또는 각 옵션을 테스트하여 어떤 옵션이 효과가 있는지 확인할 수 있습니다.

타입: 문자열

리소스

오류가 호출된 리소스입니다. 예컨대, 이미 존재하는 사용자를 만들려고 하면 Resource가 기존 사용자의 사용자 이름이 됩니다.

타입: 문자열

ResourceType

오류가 호출되는 리소스 타입입니다. 예컨대, 이미 존재하는 사용자를 만들려고 하면 ResourceType이 User가 됩니다.

타입: 문자열

RetryAfterSeconds

명령을 재시도하기 전에 대기하는 시간(초).

타입: 문자열

오류 응답 예

DescribeServer API를 호출하고 존재하지 않는 서버를 지정하면 다음 JSON 본문이 반환됩니다.

```
{
  "__type": "ResourceNotFoundException",
  "Message": "Unknown server",
  "Resource": "s-11112222333344444",
  "ResourceType": "Server"
}
```

API 실행 시 병목 현상이 발생하는 경우 다음 JSON 본문이 반환됩니다.

```
{
  "__type": "ThrottlingException",
  "RetryAfterSeconds": "1"
}
```

CreateServer API를 사용하고 Transfer Family 서버를 생성할 수 있는 충분한 권한이 없는 경우 다음 JSON 본문이 반환됩니다.

```
{
  "__type": "AccessDeniedException",
  "Message": "You do not have sufficient access to perform this action."
}
```

CreateUser API를 사용하고 이미 존재하는 사용자를 지정하면 다음 JSON 본문이 반환됩니다.

```
{
```

```

    "__type": "ResourceExistsException",
    "Message": "User already exists",
    "Resource": "Alejandro-Rosalez",
    "ResourceType": "User"
  }

```

사용 가능한 라이브러리

AWS 는 명령줄 도구 및 쿼리 API 대신 언어별 APIs를 사용하여 애플리케이션을 빌드하려는 소프트웨어 개발자를 위해 라이브러리, 샘플 코드, 자습서 및 기타 리소스를 제공합니다. 이러한 라이브러리는 보다 쉽게 시작하도록 요청 인증, 요청 재시도 및 오류 처리 같은 기본 기능(API에는 포함되지 않음)을 제공합니다. [빌드할 도구 AWS](#) 참조

모든 언어의 라이브러리와 샘플 코드는 [샘플 코드 및 라이브러리](#)를 참조하세요.

자격 증명 공급자

AWS Transfer Family 는 사용자를 인증하고 관리하기 위해 여러 자격 증명 공급자 유형을 지원합니다. 각 서버는 하나의 인증 방법만 사용할 수 있으며, 서버 생성 시이 방법을 선택해야 합니다.

서비스 관리형

SERVICE_MANAGED 인증 방법을 사용하면 사용자 자격 증명이 내부에 저장되고 관리됩니다 AWS Transfer Family. 사용자는 서버의 사용자 이름과 연결된 SSH 퍼블릭 키를 사용하여 인증됩니다.

각 사용자는 서비스에 저장된 SSH 퍼블릭 키를 하나 이상 가질 수 있습니다. 클라이언트가 파일 작업을 요청하면 저장된 퍼블릭 키에 대해 인증된 사용자 이름과 SSH 프라이빗 키를 제공합니다.

Directory Service

AWS_DIRECTORY_SERVICE 인증 방법을 사용하면 Directory Service for Microsoft Active Directory(AWS Directory Service for Microsoft Active Directory)와 통합할 수 있습니다.

이 옵션을 사용하면 기존 Active Directory 그룹을 통해 사용자 인증 및 액세스를 관리할 수 있습니다. 사용자는 Active Directory 자격 증명을 사용하여 인증할 수 있습니다.

기본 제한은 서버당 100개의 Active Directory 그룹이며, 서비스 제한 증가를 통해 최대 150개의 그룹으로 늘릴 수 있습니다.

Lambda

AWS_LAMBDA 인증 방법을 사용하면 사용하여 사용자 지정 자격 증명 공급자에 연결할 수 있습니다 AWS Lambda.

이 옵션은 기존 자격 증명 관리 시스템과 통합할 수 있는 유연성을 제공합니다. Lambda 함수는 사용자를 인증하고 적절한 액세스 정책을 반환하는 역할을 합니다.

사용자 지정(API Gateway)

API_GATEWAY 인증 방법(콘솔에 사용자 지정으로 표시됨)을 사용하면 사용자 인증과 액세스 제어를 모두 제공하는 사용자 지정 인증 방법을 사용할 수 있습니다.

이 메서드는 Amazon API Gateway에 의존하여, 자격 증명 공급자가 제공한 API 직접 호출을 이용해 사용자 요청을 확인합니다. 이 사용자 지정 메서드를 이용해 디렉터리 서비스, 데이터베이스 이름과 암호 쌍 또는 기타 메커니즘에 대해 사용자를 인증할 수도 있습니다.

모든 인증 방법에 대해 사용자에게 Amazon S3 버킷 또는 Amazon Elastic File System 파일 시스템에 대한 액세스를 정의하는 정책이 할당됩니다. 서버는 AssumeRole 작업과 함께 IAM 역할을 통해 사용자로부터 신뢰 관계를 상속하므로 사용자를 대신하여 파일 작업을 수행할 수 있습니다.

이름 지정 규칙

AWS Transfer Family 는 리소스 식별자 및 Amazon 리소스 이름(ARNs)에 표준화된 형식을 사용합니다. API로 작업할 때는 이러한 규칙을 이해하는 것이 AWS Transfer Family 중요합니다.

ID 형식

다음 규칙은 AWS Transfer Family ID 형식으로 관찰됩니다.

서버 IDs

s-01234567890abcdef 양식에서 얻은 ServerId 값

SSH 퍼블릭 키 IDs

key-01234567890abcdef 양식에서 얻은 SshPublicKeyId 값

커넥터 IDs

c-01234567890abcdef 양식에서 얻은 ConnectorId 값

워크플로 IDs

w-01234567890abcdef 양식에서 얻은 WorkflowId 값

프로필 IDs

p-01234567890abcdef 양식에서 얻은 ProfileId 값

WebApp IDs

webapp-01234567890abcdef 양식에서 얻은 WebAppId 값

ARN 형식

Amazon 리소스 이름(ARN) 양식은 다음과 같은 형식을 취합니다.

서버 ARNs

사용자의 경우, ARN은 `arn:aws:transfer:region:account-id:server/server-id` 형식을 취합니다.

예시: `arn:aws:transfer:us-east-1:123456789012:server/s-01234567890abcdef`.

사용자 ARNs

사용자의 경우, ARN은 `arn:aws:transfer:region:account-id:user/server-id/username` 형식을 취합니다.

예시: `arn:aws:transfer:us-east-1:123456789012:user/s-01234567890abcdef/user1`.

커넥터 ARNs

커넥터의 경우 ARNs 취합니다 `arn:aws:transfer:region:account-id:connector/connector-id`.

예시: `arn:aws:transfer:us-east-1:123456789012:connector/c-01234567890abcdef`.

워크플로 ARNs

워크플로의 경우 ARNs 취합니다 `arn:aws:transfer:region:account-id:workflow/workflow-id`.

예시: `arn:aws:transfer:us-east-1:123456789012:workflow/w-01234567890abcdef`.

WebApp ARNs

웹 애플리케이션의 경우 ARNs 취합니다 `arn:aws:transfer:region:account-id:webapp/webapp-id`.

예시: `arn:aws:transfer:us-east-1:123456789012:webapp/webapp-01234567890abcdef`.

ARN을 사용하여 개체에 태그, 키 값 쌍을 할당할 수 있습니다. 태그는 이러한 개체(entity)를 그룹화하고 검색하는 데 사용할 수 있는 메타데이터입니다. 태그를 유용하게 사용할 수 있는 대표적인 분야는 회계입니다.

DNS 및 엔드포인트

AWS Transfer Family 는 API 엔드포인트와 서버 엔드포인트 모두에 대해 표준화된 DNS 이름 지정 규칙을 사용합니다. 이러한 엔드포인트를 이해하는 것은 클라이언트를 구성하고 API를 호출하는 데 필수적입니다.

API 엔드포인트

API 엔드포인트는 API 호출을 통해 AWS Transfer Family 리소스를 관리하는 데 사용됩니다. 이러한 엔드포인트의 형식은 다음과 같습니다.

표준 API 엔드포인트

표준 API 엔드포인트는 형식을 취합니다 `transfer.region.amazonaws.com`.

예시: `transfer.us-east-1.amazonaws.com`

듀얼 스택 API 엔드포인트

AWS Transfer Family 는 IPv4 또는 IPv6 요청을 사용하여 액세스할 수 있는 듀얼 스택 API 엔드포인트를 제공합니다.

- `https://transfer.region-code.api.aws`
- `https://transfer-fips.region-code.api.aws`

서버 엔드포인트

서버 엔드포인트는 파일 전송 클라이언트가 AWS Transfer Family 서버에 연결하는 데 사용됩니다. 이러한 엔드포인트의 형식은 다음과 같습니다.

표준 서버 엔드포인트

표준 서버 엔드포인트는 형식을 취합니다 `server-id.server.transfer.region.amazonaws.com`.

예시: `s-01234567890abcdef.server.transfer.us-east-1.amazonaws.com`

사용자 지정 호스트 이름

AWS Transfer Family 서버에 대한 사용자 지정 호스트 이름을 구성할 수도 있습니다. 사용자 지정 호스트 이름을 사용하여 사용자에게 보다 사용자 친화적이거나 브랜드가 지정된 환경을 제공할 수 있습니다.

사용자 지정 호스트 이름을 사용하려면 다음을 수행해야 합니다.

1. 도메인 이름 소유
2. 유효한 인증서 제공
3. AWS Transfer Family 서버를 가리키도록 DNS 레코드 구성

AWS 리전별 AWS Transfer Family 엔드포인트의 전체 목록은 [AWS Transfer Family 의 엔드포인트 및 할당량을 참조하세요](#) AWS 일반 참조.

에 대한 문서 기록 AWS Transfer Family

다음 표에서는이 릴리스에 대한 설명서를 설명합니다 AWS Transfer Family.

- API 버전: transfer-2018-11-05
- 최종 설명서 업데이트: 2025년 11월 19일

변경	설명	Date
Transfer Family 웹 앱에 대한 VPC 엔드포인트 지원	AWS Transfer Family 웹 앱은 이제 VPC 호스팅 엔드포인트를 지원하므로 퍼블릭 인터넷을 통과하지 않고도 Amazon S3와의 프라이빗 데이터 전송을 위해 가상 프라이빗 클라우드 내에서 웹 앱 엔드포인트를 호스팅할 수 있습니다. 자세한 내용은 VPC에서 Transfer Family 웹 앱 생성 을 참조하세요.	2025년 11월 19일
IPv6 지원	AWS Transfer Family 는 이제 SFTP 퍼블릭 엔드포인트용 듀얼 스택(IPv4 및 IPv6) 엔드포인트, SFTP/FTPS/FTP/AS2용 VPC 내부 엔드포인트, SFTP 및 AS2 커넥터, API 엔드포인트를 지원합니다. 자세한 내용은 Transfer Family 서버에 대한 IPv6 지원 을 참조하세요.	2025년 6월 30일
SFTP 커넥터 사용자 경험 개선 사항	• 커넥터에 대한 동시 연결 설정을 셀프 서비스하는 기능 • 연결을 인증하는 데 사용되는 OpenSSH 형식의 SSH 프라이빗 키를 제공하는 기능	2025년 4월 9일

변경	설명	Date
	SFTP 커넥터를 사용하여 원격 서버의 퍼블릭 호스트 키를 검색할 수 있는 기능 자세한 내용은 서비스 관리형 송신을 사용하여 SFTP 커넥터 생성을 참조하세요.	
SFTP 커넥터가 원격 SFTP 서버에서 파일을 삭제, 이름 변경 및 이동할 수 있는 기능	이제 원격 서버에서 복사한 후 소스 파일을 삭제하거나 이름을 바꾸거나 아카이브 위치로 이동하여 원격 SFTP 서버에 저장된 파일을 구성할 수 있습니다. 자세한 내용은 원격 SFTP 서버에서 파일 또는 디렉터리 이동 또는 이름 바꾸기을 참조하세요.	2025년 4월 7일
AWS Transfer Family 웹 앱 지원	AWS Transfer Family 웹 앱은 고객이 웹 브라우저를 통해 Amazon S3의 데이터에 액세스하기 위한 간단한 인터페이스를 생성하는 데 사용할 수 있는 새로운 리소스입니다. 자세한 내용은 Transfer Family 웹 앱을 참조하세요.	2024년 12월 1일
API 참조를 별도의 가이드로 이동	고객 경험을 개선하기 위해 이제 API 참조가 사용 설명서와 별도로 게시됩니다. 별도의 API 참조는 AWS Transfer Family API 시작을 참조하세요.	2024년 7월 31일

변경	설명	Date
SFTP 커넥터가 원격 파일 및 디렉터리를 나열하는 기능	Transfer Family는 고객이 SFTP 커넥터를 사용하여 원격 SFTP 서버에 저장된 파일을 나열할 수 있는 기능을 추가했습니다. 자세한 내용은 원격 디렉터리의 콘텐츠 나열 섹션을 참조하세요.	2024년 4월 23일
AS2 메시지 교환과 함께 거래 파트너의 자체 서명된 TLS 인증서를 사용할 수 있는 기능	AWS Transfer Family 는 HTTPS를 통해 서버에 적용성 문 2(AS2) 메시지를 보내기 위해 거래 파트너의 자체 서명된 공개 TLS 인증서를 가져오고 사용하는 옵션을 추가했습니다.	2024년 4월 12일
SFTP 커넥터에 대한 보안 정책 추가	AWS Transfer Family 에 SFTP 커넥터에 사용할 보안 정책이 추가되었습니다. 자세한 내용은 AWS Transfer Family SFTP 커넥터에 대한 보안 정책 을 참조하세요.	2024년 4월 5일
Amazon EventBridge 통합	AWS Transfer Family 이제는 모든 파일 전송 작업에 대한 이벤트를 Amazon EventBridge에 자동으로 게시합니다. 자세한 내용은 를 사용하여 Transfer Family 이벤트 관리 Amazon EventBridge 을 참조하세요.	2024년 2월 8일

변경	설명	Date
새 보안 정책 추가	AWS Transfer Family 에 새로운 FIPS 및 비 FIPS 보안 정책이 추가되었습니다. 또한 서버에 할당된 기본 보안 정책은 항상 최신 보안 정책입니다. 자세한 내용은 AWS Transfer Family 서버에 대한 보안 정책 을 참조하세요.	2024년 2월 5일
SFTP 커넥터 및 AS2의 고정 IP 주소 지원	Transfer Family는 이제 SFTP 커넥터 및 AS2에 대한 고정 IP 주소를 제공합니다. 이렇게 하면 IP 허용 목록 제어로 보호되는 원격 SFTP 서버와의 연결이 활성화됩니다. AS2의 경우 AS2 서버의 비동기 MDN 응답에 정적 IP 주소를 도입합니다.	2024년 1월 16일
사용 설명서가 최신 버전에서 더 가깝게 맞게 재구성되었습니다 AWS Transfer Family.	Transfer Family는 가이드가 시작된 이후 여러 기능을 추가했으므로 가이드를 재구성해야 합니다.	2024년 1월 3일

변경	설명	Date
논리적 디렉터리 매핑 개선 사항 Amazon S3 목록 성능 최적화	Transfer Family는 이제 최대 2.1MB의 논리적 디렉터리 매핑을 지원합니다. 이제 사용자 매핑이 파일에 대한 것인지 여부를 선언할 수도 있습니다. 자세한 내용은 논리적 디렉터리 사용 규칙 단원을 참조하십시오. Amazon S3를 스토리지로 사용하는 서버를 생성하거나 업데이트할 때 이제 S3 디렉터리 (또는 폴더)를 나열하는 성능을 최적화할 수 있습니다. 자세한 내용은 SFTP, FTPS 또는 FTP 서버 엔드포인트 구성 단원을 참조하십시오.	2023년 11월 17일
Virtual Private Cloud(VPC) 엔드포인트가 있는 SFTP 서버의 대체 포트	이제 VPC 엔드포인트가 있는 SFTP Transfer Family 서버에 대해 표준이 아닌 대체 포트를 활성화할 수 있습니다. 자세한 내용은 Virtual Private Cloud(VPC)에 서버 생성 단원을 참조하십시오.	2023년 11월 17일
SFTP 커넥터 지원	SFTP 커넥터는의 기능을 확장 AWS Transfer Family 하여 클라우드 및 온프레미스 모두에서 원격 서버와 통신합니다. 자세한 내용은 SFTP 커넥터 사용 단원을 참조하십시오.	2023년 7월 25일

변경	설명	Date
AS2 기본 인증 지원	Transfer Family는 이제 적용성 보고서 2(AS2) 프로토콜을 사용하는 서버에 대해 기본 인증 사용을 지원합니다. 자세한 내용은 AS2 커넥터의 기본 인증 을 참조하세요.	2023년 6월 30일
정형 JSON 로깅 지원	Transfer Family는 이제 구조화된 JSON 로그를 Amazon CloudWatch로 전송하고, 로그 스트림을 사용자 지정 로그 그룹으로 그룹화하고, 프로토콜 간에 일반적인 로그 쿼리를 수행할 수 있도록 지원합니다. 자세한 내용은 AWS Transfer Family 서버에 대한 Amazon CloudWatch 로깅 단원을 참조하십시오.	2023년 6월 24일
다양한 인증 방법 지원	Transfer Family는 암호, 퍼블릭/프라이빗 키 쌍 또는 둘 다를 사용한 인증을 지원합니다. 이는 SFTP 프로토콜을 사용하는 서버와 사용자 지정 자격 증명 공급자에 사용할 수 있습니다. 자세한 내용은 SFTP 지원 서버 생성 을 참조하세요.	2023년 5월 17일

변경	설명	Date
Transfer Family가 워크플로를 통해 처리하는 파일에 대한 Pretty Good Privacy(PGP) 암호 해독 지원	Transfer Family는 Pretty Good Privacy(PGP) 복호화를 기본적으로 지원합니다. SFTP, FTPS 또는 FTP를 통해 Amazon Simple Storage Service(S3) 또는 Amazon Elastic File System(Amazon EFS)에 업로드된 파일에서 PGP 복호화를 사용할 수 있습니다. 자세한 내용은 PGP 키 생성 및 워크플로에서 PGP 복호화 사용 섹션을 참조하세요.	2022년 12월 21일
Transfer Family를 통한 적응성 보고서 2(AS2) 파일 전송 프로토콜의 완전관리형 지원	AS2 프로토콜을 사용하여 AWS 환경 내부 또는 외부에 있는 거래 파트너와 정보를 주고받는 서버를 생성할 수 있습니다. 자세한 내용은 AS2 구성 단원을 참조하십시오.	2022년 7월 25일
서버 생성 시 디스플레이 배너 지원	서버를 생성할 때 사용자 지정 메시지를 추가할 수 있습니다. 사전 인증 메시지(모든 프로토콜)와 사후 인증 메시지(FTP 및 FTPS 서버용)를 표시할 수 있습니다. 자세한 내용은 SFTP 지원 서버 생성 , FTPS 지원 서버 생성 또는 FTP 지원 서버 생성 을 참조하세요.	2022년 2월 17일

변경	설명	Date
자격 증명 공급자 AWS Lambda 로서에 대한 지원	이제 Transfer Family 서버 AWS Lambda 와 함께를 사용하여 사용자 지정 자격 증명 공급자에 연결할 수 있습니다. 이전에는 사용자 지정 자격 증명 공급자를 통합하려면 Amazon API Gateway URL을 제공해야 했습니다. 자세한 내용은 AWS Lambda 를 사용하여 자격 증명 공급자 통합 를 참조하세요.	2021년 11월 16일
관리형 File Transfer 워크플로 지원	관리형 File Transfer 워크플로는 현재 수동으로 수행하는 일반적인 작업에 대한 업로드 후 처리 추상화를 제공합니다. 자세한 내용은 AWS Transfer Family 관리형 워크플로 를 참조하세요.	2021년 9월 2일
에 대한 지원 AWS Directory Service for Microsoft Active Directory	이제 서비스 관리형 및 사용자 지정 자격 증명 공급자 외에도 AWS Directory Service for Microsoft Active Directory 를 사용하여 인증 및 권한 부여를 위한 사용자 액세스를 관리할 수 있습니다. 자세한 내용은 Microsoft Active Directory 용 AWS Directory Service 사용 단원을 참조하십시오.	2021년 5월 24일

변경	설명	Date
신규 AWS 리전	AWS Transfer Family 이제 아프리카(케이프타운) 리전에서 사용할 수 있습니다. Transfer Family 엔드포인트에 대한 자세한 내용은 AWS 일반 참조의 AWS Transfer Family 엔드포인트 및 할당량 을 참조하세요.	2021년 2월 24일
신규 AWS 리전	AWS Transfer Family 이제 아시아 태평양(홍콩) 및 중동(바레인) 리전에서 사용할 수 있습니다. Transfer Family 엔드포인트에 대한 자세한 내용은 AWS 일반 참조의 AWS Transfer Family 엔드포인트 및 할당량 을 참조하세요.	2021년 2월 17일
데이터 스토어로 Amazon EFS 지원	Transfer Family는 이제 Amazon Elastic File System(Amazon EFS)에서 송수신하는 File Transfer를 지원합니다. Amazon EFS는 단순하고 확장 가능한 완전 관리형 탄력적 NFS 파일 시스템입니다. 자세한 내용은 Amazon EFS 파일 시스템 구성 를 참조하세요.	2021년 1월 6일

변경	설명	Date
에 대한 지원 AWS WAF	Transfer Family는 이제 공격으로부터 웹 애플리케이션 및 API 작업을 보호하는 데 도움이 되는 웹 애플리케이션 방화벽 AWS WAF을 지원합니다. 자세한 내용은 웹 애플리케이션 방화벽 추가 단원을 참조하십시오.	2020년 11월 24일
Virtual Private Cloud(VPC)의 여러 보안 그룹 지원	이제 VPC의 서버에 여러 보안 그룹을 연결할 수 있습니다. 자세한 내용은 Virtual Private Cloud(VPC)에 서버 생성 를 참조하세요.	2020년 10월 15일
신규 AWS 리전	이제 AWS GovCloud (US) 리전에서 Transfer Family를 사용할 수 있습니다. AWS GovCloud (US) 리전의 Transfer Family 엔드포인트에 대한 자세한 내용은 AWS Transfer Family 엔드포인트 및 할당량을 참조하세요 AWS 일반 참조. AWS GovCloud (US) 리전에서 Transfer Family를 사용하는 방법에 대한 자세한 내용은 AWS GovCloud (US) 사용 설명서 AWS Transfer Family 의 섹션을 참조하세요.	2020년 9월 30일

변경	설명	Date
이제 지원되는 암호화 알고리즘이 포함된 보안 정책을 서버에 연결할 수 있습니다.	이제 지원되는 암호화 알고리즘이 포함된 보안 정책을 서버에 연결할 수 있습니다. 자세한 내용은 AWS Transfer Family 서버에 대한 보안 정책 를 참조하세요.	2020년 8월 12일
FIPS(Federal Information Processing Standard) 엔드포인트 지원	이제 북미 AWS 리전에서 FIPS 지원 엔드포인트를 사용할 수 있습니다. 사용할 수 있는 리전은 AWS 일반 참조의 AWS Transfer Family 엔드포인트 및 할당량 을 참조하세요. SFTP 지원 서버 엔드포인트에 FIPS를 활성화하려면 SFTP 지원 서버 생성 을 참조하세요. FTPS 지원 서버 엔드포인트에 FIPS를 활성화하려면 FTPS 지원 서버 생성 을 참조하세요. FTP 지원 서버 엔드포인트에 FIPS를 활성화하려면 FTP 지원 서버 생성 을 참조하세요.	2020년 8월 12일
사용자 이름 문자 길이 증가 및 추가 허용 문자	이제 사용자 이름에 @ 기호(@)와 마침표(.)를 포함할 수 있으며 최대 길이는 100자일 수 있습니다. 사용자를 추가하려면 서버 엔드포인트에 대한 사용자 관리 를 참조하세요.	2020년 8월 12일

변경	설명	Date
자동 Amazon CloudWatch 로깅 AWS Identity and Access Management (IAM) 역할 생성 지원	Transfer Family는 이제 최종 사용자 활동을 볼 수 있는 CloudWatch 로깅 IAM 역할의 자동 생성을 지원합니다. 자세한 내용은 SFTP 지원 서버 생성 , FTPS 지원 서버 생성 또는 FTP 지원 서버 생성 를 참조하세요.	2020년 7월 30일
AWS Transfer Family 는 이제 소스 IP를 권한 부여 요소로 지원합니다.	Transfer Family는 최종 사용자의 소스 IP 주소를 인증 요소로 사용할 수 있는 지원을 추가하여 SFTP(보안 파일 전송 프로토콜), FTPS(SSL을 통한 파일 전송 프로토콜) 또는 FTP(파일 전송 프로토콜)를 통한 액세스를 승인할 때 추가 보안 계층을 적용할 수 있습니다. 자세한 내용은 사용자 지정 자격 증명 공급자와 작업 단원을 참조하십시오.	2020년 6월 9일
AWS SFTP용 전송은 이제이며 FTP AWS Transfer Family 및 FTPS에 대한 지원을 추가합니다.	이제 사용자의 파일 전송에 FTPS(파일 전송 프로토콜 보안)와 FTP(파일 전송 프로토콜)라는 두 가지 추가 프로토콜을 사용할 수 있습니다. 사용자는 기존 SFTP(Secure File Transfer Protocol) 지원 외에도에서 FTP over SSL(FTPS) 및 일반 텍스트 FTP 기반 워크플로를 이동, 실행 AWS, 보호 및 통합할 수 있습니다.	2020년 4월 23일

변경	설명	Date
Virtual Private Cloud(VPC) 보안 그룹 및 탄력적 IP 주소 지원	이제 보안 그룹을 사용하여 수신 IP 주소에 대한 허용 목록을 생성하여 서버에 추가 보안 계층을 제공할 수 있습니다. 엘라스틱 IP 주소를 서버의 엔드포인트와 연결할 수도 있습니다. 이렇게 하면 방화벽을 사용하는 사용자가 해당 엔드포인트에 대한 액세스를 허용할 수 있습니다. 자세한 내용은 Virtual Private Cloud(VPC)에 서버 생성 을 참조하세요.	2020년 1월 10일
VPC에서의 작업 지원	이제 VPC에 서버를 만들 수 있습니다. 퍼블릭 인터넷을 거치지 않고 서버를 사용하여 클라이언트를 통해 Amazon S3 버킷과 데이터를 주고 받을 수 있습니다. 자세한 내용은 Virtual Private Cloud(VPC)에 서버 생성 을 참조하세요.	2019년 3월 27일
의 첫 번째 버전이 AWS Transfer Family 릴리스되었습니다.	이 최초 릴리스는 방향 설정을 포함하며, 시작 방법을 설명하고, 클라이언트 구성과 사용자 구성 및 모니터링 활동 관련 정보를 제공합니다.	2018년 11월 25일

기계 번역으로 제공되는 번역입니다. 제공된 번역과 원본 영어의 내용이 상충하는 경우에는 영어 버전이 우선합니다.